ROYAL SOCIETY OPEN SCIENCE

rsos.royalsocietypublishing.org

Brief report

Cite this article: Moore CJ, Chua AJK, Berry CPL, Gair JR. 2016 Fast methods for training Gaussian processes on large datasets. *R. Soc. open sci.* **3**: 160125.
http://dx.doi.org/10.1098/rsos.160125

Received: 23 February 2016
Accepted: 7 April 2016

Subject Category:
Mathematics

Subject Areas:
statistics

Keywords:
Gaussian processes, regression, data analysis, inference

Author for correspondence:
C. J. Moore
e-mail: cjm96@ast.cam.ac.uk

Fast methods for training Gaussian processes on large datasets

C. J. Moore[1], A. J. K. Chua[1], C. P. L. Berry[2] and J. R. Gair[3]

[1]Institute of Astronomy, Madingley Road, Cambridge CB3 0HA, UK
[2]School of Physics and Astronomy, University of Birmingham, Birmingham B15 2TT, UK
[3]School of Mathematics, University of Edinburgh and Biomathematics and Statistics Scotland, James Clerk Maxwell Building, Peter Guthrie Tait Road, Edinburgh EH9 3FD, UK

AJKC, 0000-0001-5242-8269

Gaussian process regression (GPR) is a non-parametric Bayesian technique for interpolating or fitting data. The main barrier to further uptake of this powerful tool rests in the computational costs associated with the matrices which arise when dealing with large datasets. Here, we derive some simple results which we have found useful for speeding up the learning stage in the GPR algorithm, and especially for performing Bayesian model comparison between different covariance functions. We apply our techniques to both synthetic and real data and quantify the speed-up relative to using nested sampling to numerically evaluate model evidences.

1. Introduction

A wide range of commonly occurring inference problems can be fruitfully tackled using Bayesian methods. A particularly common inference problem is that of regression; determining the relationship of a control variable x to an output variable y given a set of measurements of $\{y_i\}$ at points $\{x_i\}$. The solution requires a model $y = f(x)$, which allows us to predict the value of y at an untested value of x. From a Bayesian standpoint, this can be achieved using Gaussian processes (GPs): a GP is a collection of random variables, of which any finite subset have a joint Gaussian probability distribution [1].

Gaussian process regression (GPR) is a powerful mathematical technique for performing non-parametric regression in a Bayesian framework [1–5]. The key assumption underpinning the method is that the observed dataset being interpolated is a realization of a GP with a particular covariance function. This assumption presents us with a challenge: how do we choose the covariance function which gives the best interpolant?

THE ROYAL SOCIETY PUBLISHING

The process of choosing the covariance function is known as *learning*, or *training* of the GP. In this training process, it is necessary to compute the inverse of the covariance matrix (the matrix formed by evaluating the covariance function pairwise between all n observed points). The time taken to evaluate the inverse of the covariance matrix scales as $\mathcal{O}(n^3)$ [1], where n is the number of points being interpolated; this has typically restricted the application of GPR to smaller problems ($n \lesssim 10^5$), although work has been done on extending its applicability to larger datasets [6–9].

In this paper, we present two techniques that speed up the training stage of the GPR algorithm. The first aims to reduce the dimensionality of the problem, and hence speed up the learning of the hyperparameters for a single covariance function. This does not change the fact that the cost of this process is $\mathcal{O}(n^3)$; instead it simply reduces the constant in this scaling. The second aims to enable fast Bayesian model comparison between different covariance functions while also incorporating the benefits of the first technique.

We consider maximizing the hyperlikelihood: the conditional probability of the data given a particular set of hyperparameters used to specify the covariance function.[1] We provide an expression for the Hessian matrix of the hyperlikelihood surface and show how this can be used as a valuable tool for comparing the performance of two different covariance functions. We also present modified expressions for the hyperlikelihood, its gradient and its Hessian matrix, which have all been analytically maximized and marginalized over a single-scale hyperparameter. This analytic maximization or marginalization reduces the dimensionality of the subsequent optimization problem and hence further speeds up the training and comparison of GPs.

These techniques are useful when attempting to rapidly fit large, irregularly sampled datasets with a variety of covariance function models. The authors have previously made use of these techniques in exploring the correlation structure of the differences between complicated waveform models in the field of gravitational-wave astronomy [10,11]; this was done so that the effect of different models on the parameter inferences could be marginalized over. There, the behaviour of the data was largely unknown *a priori* and it was necessary to quantitatively compare a wide range of different covariance functions. Work in this area with larger datasets is ongoing.

In §2, we review the GPR method and discuss methods of efficiently determining a covariance function. In §2.1, we present our expression for the Hessian of the hyperlikelihood along with a discussion of how it can be used for model comparison, and in §2.2, we show how the training of the GP can be accelerated by analytically maximizing or marginalizing the hyperlikelihood over a single-scale parameter. In §3, we apply these methods to both synthetic and real datasets, and compare the computational cost to that of a full numerical evaluation of the Bayesian model evidences. Finally, a brief discussion and concluding remarks are given in §4.

2. Gaussian process regression and training

The technique of GPR is a method for interpolating (or extrapolating) the data contained in a *training set* $\mathcal{D} = \{\boldsymbol{x}, \boldsymbol{y}\}$. The vector $\boldsymbol{x} = \{x_i \mid i = 1, 2, \ldots, n\}$ is called the input vector and the output vector is given by $y_i = f(x_i)$ for some unknown function f. The method works by assuming that the data have been drawn from an underlying GP $f(x) \sim \mathcal{GP}(\mu(x), k(x, x'))$ with specified mean $\mu(x)$ (usually assumed to be zero) and covariance function $k(x, x')$.

There is freedom in specifying the covariance function; common choices, such as the squared exponential and Matérn function, include a number m of free hyperparameters $\boldsymbol{\theta} = \{\theta_i \mid i = 1, 2, \ldots, m\}$ that control the properties of the GP, i.e. $k(x, x') = k(x, x'; \boldsymbol{\theta})$.

The predictive power of the method comes from computing the conditional probability of the function taking a given value at some new $(n + 1)$th input point x_*, given the observed values in $\mathcal{D}$ and the values of the hyperparameters $\boldsymbol{\theta}$. This predictive probability distribution $P(y(x_*) \mid \mathcal{D}, \boldsymbol{\theta})$ for the function at the new point is a Gaussian with mean $\bar{y}(x_*)$ and variance $[\sigma_y(x_*)]^2$ [1],

$$\bar{y}(x_*) = \boldsymbol{k}_*^{\mathrm{T}} \mathbf{K}^{-1} \boldsymbol{y}, \quad [\sigma_y(x_*)]^2 = k_{**} - \boldsymbol{k}_*^{\mathrm{T}} \mathbf{K}^{-1} \boldsymbol{k}_*, \tag{2.1}$$

and

$$y(x_*) \mid \{\mathcal{D}, \boldsymbol{\theta}\} \sim \mathcal{N}(\bar{y}(x_*), \sigma_y(x_*)), \tag{2.2}$$

[1]This is often referred to as the marginal likelihood. In order to avoid confusion with distributions over the model parameters, we prefer to consistently use the 'hyper' prefix to denote probability distributions connected to the inference of the hyperparameters.

where we have defined the scalar, vector and matrix shorthand

$$k_{**} \equiv k(x_*, x_*), \quad [\boldsymbol{k}_*]_i \equiv k(x_*, x_i) \quad \text{and} \quad [\mathbf{K}]_{ij} \equiv k(x_i, x_j). \tag{2.3}$$

Since the posterior distribution for (2.2) relies upon the form of the covariance, GPR cannot be used to make definite predictions until we have fixed a method for dealing with the unknown hyperparameters $\boldsymbol{\theta}$.

Ideally, we place a prior probability distribution on $\boldsymbol{\theta}$ and make predictions by evaluating the integral

$$P(y(x_*) \mid \mathcal{D}) = \int \mathrm{d}\boldsymbol{\theta} P(y(x_*) \mid \mathcal{D}, \boldsymbol{\theta}) P(\boldsymbol{\theta} \mid \mathcal{D}) = \int \mathrm{d}\boldsymbol{\theta} P(y(x_*) \mid \mathcal{D}, \boldsymbol{\theta}) P(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta}) P(\boldsymbol{\theta}), \tag{2.4}$$

where we have used Bayes' theorem to obtain the second equality. We have introduced the hyperlikelihood given by

$$\ln P(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta}) = -\tfrac{1}{2}[\boldsymbol{y}^{\mathrm{T}} \mathbf{K}^{-1} \boldsymbol{y} + \ln(\det \mathbf{K}) + n \ln(2\pi)], \tag{2.5}$$

which encodes the probability that the observed (training) data were drawn from a GP with covariance function k. The integral (2.4) is almost always analytically intractable and prohibitively expensive to evaluate numerically. A common approximate approach is to use the most probable values of the hyperparameters $\hat{\boldsymbol{\theta}}$, which maximize $P(\boldsymbol{\theta} \mid \mathcal{D})$ [12–14].

Assuming the prior distribution is sufficiently flat (or uninformative) over the region of interest, this is equivalent to maximizing the hyperlikelihood $P(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta})$. Under this approximation, the predictive distribution becomes

$$P(y(x_*) \mid \mathcal{D}) \simeq P(y(x_*) \mid \mathcal{D}, \hat{\boldsymbol{\theta}}), \tag{2.6}$$

which is simply the Gaussian in (2.2) with mean and variance evaluated at $\hat{\boldsymbol{\theta}}$. Implementing the above procedure requires numerically maximizing the hyperlikelihood in (2.5). This can be computationally expensive; in §2.1 and §2.2, we present methods for reducing the cost of maximizing the hyperlikelihood.

2.1. Using the gradient and Hessian

The maximization process may be accelerated if the gradient of the hyperlikelihood is known and a gradient-based algorithm, such as a conjugate gradient method [13,15], can be used. The gradient of the logarithm of the hyperlikelihood is given by [1]

$$\partial_{\boldsymbol{\theta}} \ln P(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta}) = \tfrac{1}{2} \boldsymbol{y}^{\mathrm{T}} \mathbf{K}^{-1} \cdot \partial_{\boldsymbol{\theta}} \mathbf{K} \cdot \mathbf{K}^{-1} \boldsymbol{y} - \tfrac{1}{2} \mathrm{Tr}(\mathbf{K}^{-1} \cdot \partial_{\boldsymbol{\theta}} \mathbf{K}). \tag{2.7}$$

This can be shown by differentiating (2.5) and making use of the standard results

$$\partial \mathbf{K}^{-1} = -\mathbf{K}^{-1} \cdot \partial \mathbf{K} \cdot \mathbf{K}^{-1}, \quad \partial(\det \mathbf{K}) = (\det \mathbf{K}) \mathrm{Tr}(\mathbf{K}^{-1} \cdot \partial \mathbf{K}). \tag{2.8}$$

The gradient in (2.7) is useful because the rate-determining step in computing the hyperlikelihood is computing the inverse matrix $\mathbf{K}^{-1}$ (usually achieved through a Cholesky decomposition in practice), which is an $\mathcal{O}(n^3)$ operation. All other steps in (2.5) scale as $\mathcal{O}(n^2)$ or less.[2] Once the inverse has been calculated, the gradient in (2.7) may also be evaluated in $\mathcal{O}(n^2)$; so in evaluating the hyperlikelihood for a large training set we can also get the gradient for negligible extra cost.

The procedure outlined above can be performed for multiple covariance functions, each yielding a different GP interpolant. It is, therefore, necessary to have a method of comparing the performance of different interpolants to decide which to use. One way to achieve this is to evaluate the (hyperprior-weighted) volume under the hyperlikelihood surface, the hyperevidence, and use this as a figure of merit for the performance. Evaluating this integral is prohibitive, so an approximation is to calculate the Hessian matrix of the $\ln P(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta})$ surface at the peak (the position and value of which have already been found) and to analytically integrate the resulting Gaussian. This procedure assumes flat (or slowly varying) hyperpriors in the vicinity of the peak, but this has already been assumed in going from (2.4) to (2.6). Differentiating the gradient in (2.7), again making use of the results in (2.8), and evaluating the

[2]As described in a footnote in [1], the matrix–matrix products in (2.7) should not be evaluated directly, as this is an $\mathcal{O}(n^3)$ operation. Rather, the first term should be evaluated in terms of matrix–vector products, and, in the second term, only the diagonal elements that contribute to the trace need to be calculated; these are both $\mathcal{O}(n^2)$ operations.

derivatives at the position of peak hyperlikelihood, $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$, gives the Hessian,

$$\partial_{\boldsymbol{\theta}}\partial_{\boldsymbol{\theta}'} \ln P(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta})|_{\hat{\boldsymbol{\theta}}} = -\tfrac{1}{2}\boldsymbol{y}^{\mathrm{T}}[2\mathbf{K}^{-1} \cdot \partial_{\boldsymbol{\theta}}\mathbf{K} \cdot \mathbf{K}^{-1}\partial_{\boldsymbol{\theta}'}\mathbf{K} \cdot \mathbf{K}^{-1} - \mathbf{K}^{-1} \cdot \partial_{\boldsymbol{\theta}}\partial_{\boldsymbol{\theta}'}\mathbf{K} \cdot \mathbf{K}^{-1}]\boldsymbol{y} + \tfrac{1}{2}\mathrm{Tr}(\mathbf{K}^{-1} \cdot \partial_{\boldsymbol{\theta}}\mathbf{K} \cdot \mathbf{K}^{-1} \cdot \partial_{\boldsymbol{\theta}'}\mathbf{K} - \mathbf{K}^{-1} \cdot \partial_{\boldsymbol{\theta}}\partial_{\boldsymbol{\theta}'}\mathbf{K}) = -\mathbf{H}. \tag{2.9}$$

This expression has the same advantages as the expression for the gradient; as the inverse of the covariance matrix has already been computed, the Hessian may be evaluated at negligible extra cost. The hyperlikelihood surface may therefore be approximated by the Gaussian [12,16]

$$\ln P(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta}) \approx \ln P(\boldsymbol{y} \mid \boldsymbol{x}, \hat{\boldsymbol{\theta}}) - \tfrac{1}{2}\Delta\boldsymbol{\theta}^{\mathrm{T}} \cdot \mathbf{H} \cdot \Delta\boldsymbol{\theta}. \tag{2.10}$$

We seek the hyperevidence, which is given by the following integral of the hyperposterior, where we have specified a prior $\Pi(\boldsymbol{\theta})$ on the hyperparameters;

$$\mathcal{Z}(\mathcal{D}) = \int \mathrm{d}\boldsymbol{\theta}\, \Pi(\boldsymbol{\theta}) P(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta}). \tag{2.11}$$

Assuming the hyperposterior is a sufficiently well-peaked distribution, with peak at position $\boldsymbol{\theta} = \tilde{\boldsymbol{\theta}}$, the hyperevidence may be written using the Laplace approximation [2] as

$$\mathcal{Z}(\mathcal{D}) \approx \Pi(\tilde{\boldsymbol{\theta}}) P(\boldsymbol{y} \mid \boldsymbol{x}, \tilde{\boldsymbol{\theta}}) \sqrt{\frac{(2\pi)^m}{\det(\boldsymbol{H} + \boldsymbol{H}_{\Pi})}}. \tag{2.12}$$

It is always possible to change the hyperparametrization so that the prior is flat in which case the hyperposterior is proportional to the hyperlikelihood.[3] If such a hyperparametrization has been chosen then $\Pi(\tilde{\boldsymbol{\theta}}) = 1/V$ (where V is the hyperprior volume, or range of integration), $\boldsymbol{H}_{\Pi} = 0$ and $\tilde{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}}$; therefore

$$\mathcal{Z}(\mathcal{D}) \approx \frac{P(\boldsymbol{y} \mid \boldsymbol{x}, \hat{\boldsymbol{\theta}})}{V} \sqrt{\frac{(2\pi)^m}{\det \mathbf{H}}}. \tag{2.13}$$

This expression is now invariant under further changes to the hyperparameter specification which preserve the property that the prior is constant. We use hyperparametrizations with flat hyperpriors as this choice uniquely specifies the approximation in equation (2.13); although there remains the possibility that another hyperparametrization exists in which the posterior is better approximated as a Gaussian.

For two covariance functions, k_1 and k_2, the odds ratio may be defined as the ratio of the value of (2.13) evaluated with k_1 to the value evaluated using k_2, and this may be used to discriminate among competing models. The hyperprior volume V in (2.13) acts as an Occam factor, penalizing models with greater complexity [2]. Once suitable prior volumes have been fixed, the Hessian approximation to the hyperevidence is a computationally inexpensive means of comparing covariance functions.

The Hessian may also be used to provide error estimates for the hyperparameters; from (2.10) it can be seen that the inverse of the Hessian is the covariance matrix of the maximum hyperlikelihood estimator of the hyperparameters.

2.2. Partial analytic maximization

In general, covariance functions can be arbitrarily complicated, with large numbers of hyperparameters. Inevitably, simple covariance functions are the most prevalent in the literature. If there are a small number of hyperparameters, then even reducing the number of hyperparameters by one can have a great impact on the length of time taken to maximize the hyperlikelihood. In this section, we show how the hyperlikelihood for any covariance function, regardless of complexity, can be analytically maximized over an overall scale parameter, thereby reducing the number of remaining hyperparameters. We also generalize the expressions for the gradient and the Hessian found in §2.1 to this case.

Consider the following transformation of the covariance, $k(\boldsymbol{x}_i, \boldsymbol{x}_j) \to \sigma_f^2 k(\boldsymbol{x}_i, \boldsymbol{x}_j)$; substituting this into the expression for the hyperlikelihood gives,

$$\ln P(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta}) = -\frac{1}{2\sigma_f^2}\boldsymbol{y}^{\mathrm{T}}\mathbf{K}^{-1}\boldsymbol{y} - \frac{1}{2}\ln(\det \mathbf{K}) - \frac{n}{2}\ln(2\pi\sigma_f^2). \tag{2.14}$$

This function always has a unique maximum with respect to variations in σ_f^2 at the position

$$\hat{\sigma}_f^2 = \frac{1}{n}\boldsymbol{y}^{\mathrm{T}}\mathbf{K}^{-1}\boldsymbol{y}; \tag{2.15}$$

[3]For example, if the original prior on the parameters $\boldsymbol{\theta}$ is $p(\boldsymbol{\theta})$, we can define $\theta_i'(\boldsymbol{\theta}) = \int_{-\infty}^{\theta_i} p(\theta_i \mid \theta_{i+1}, \ldots, \theta_m)\, \mathrm{d}\theta_i$ and then $p(\boldsymbol{\theta}')$ is a constant.

at this point, the hyperlikelihood takes the value

$$\ln P_{\max}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\vartheta}) = -\frac{n}{2}\ln(2\pi e\hat{\sigma}_f^2) - \frac{1}{2}\ln(\det \mathbf{K}). \tag{2.16}$$

Equation (2.16) is to be considered as a function of the remaining $m-1$ hyperparameters $\boldsymbol{\vartheta} = \{\boldsymbol{\theta} \setminus \sigma_f\}$. The peak evidence may now be found more easily by numerically maximizing $\ln P_{\max}$ in (2.16) with respect to the remaining parameters $\boldsymbol{\vartheta}$. If a gradient-based algorithm is used, it is advantageous to have an analogous expression to (2.7) to give inexpensive derivatives. This can be found by differentiating (2.16) with respect to $\boldsymbol{\vartheta}$, making use of the results in (2.8),

$$\partial_{\boldsymbol{\vartheta}} \ln P_{\max}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\vartheta}) = \frac{1}{2\hat{\sigma}_f^2}\boldsymbol{y}^{\mathrm{T}}\mathbf{K}^{-1}\cdot\partial_{\boldsymbol{\vartheta}}\mathbf{K}\cdot\mathbf{K}^{-1}\boldsymbol{y} - \frac{1}{2}\mathrm{Tr}(\mathbf{K}^{-1}\cdot\partial_{\boldsymbol{\vartheta}}\mathbf{K}). \tag{2.17}$$

These are not the same as the derivatives in (2.7).

As well as maximizing, we can also consider marginalizing over σ_f [16]. As we are marginalizing over a scale parameter we use the (improper) Jeffreys prior $P(\sigma_f) = c/\sigma_f$ [17]. The result is equal to the maximized form, up to a multiplicative constant,

$$P_{\mathrm{marg}}(\boldsymbol{y}\,|\,\boldsymbol{x},\boldsymbol{\theta}) = \int_0^\infty \mathrm{d}\sigma_f \frac{c}{\sigma_f} P(\boldsymbol{y}\,|\,\boldsymbol{x},\boldsymbol{\theta}) = \frac{c}{2}\left(\frac{2e}{n}\right)^{n/2}\Gamma\left(\frac{n}{2}\right)P_{\max}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\theta}). \tag{2.18}$$

As before, once the peak hyperlikelihood has been found, the Hessian at the peak position can aid in model comparison. In this case, the Hessian should be calculated using the second derivatives of $\ln P_{\mathrm{marg}}$. However, we may instead differentiate $\ln P_{\max}$, as this differs only by a constant which will cancel when using the Hessian to compare two models. Differentiating (2.17) with respect to $\boldsymbol{\vartheta}'$,[4]

$$\begin{aligned}\partial_{\boldsymbol{\vartheta}}\partial_{\boldsymbol{\vartheta}'}\ln P_{\mathrm{marg}}|_{\hat{\boldsymbol{\vartheta}}} \propto{}& \frac{1}{2n\hat{\sigma}_f^4}\boldsymbol{y}^{\mathrm{T}}\mathbf{K}^{-1}\cdot\partial_{\boldsymbol{\vartheta}}\mathbf{K}\cdot\mathbf{K}^{-1}\boldsymbol{y}\times\boldsymbol{y}^{\mathrm{T}}\mathbf{K}^{-1}\cdot\partial_{\boldsymbol{\vartheta}'}\mathbf{K}\cdot\mathbf{K}^{-1}\boldsymbol{y}\\ &- \frac{1}{2\hat{\sigma}_f^2}\boldsymbol{y}^{\mathrm{T}}[2\mathbf{K}^{-1}\cdot\partial_{\boldsymbol{\vartheta}}\mathbf{K}\cdot\mathbf{K}^{-1}\cdot\partial_{\boldsymbol{\vartheta}'}\mathbf{K}\cdot\mathbf{K}^{-1} - \mathbf{K}^{-1}\cdot\partial_{\boldsymbol{\vartheta}}\partial_{\boldsymbol{\vartheta}'}\mathbf{K}\cdot\mathbf{K}^{-1}]\boldsymbol{y}\\ &+ \frac{1}{2}\mathrm{Tr}(\mathbf{K}^{-1}\cdot\partial_{\boldsymbol{\vartheta}}\mathbf{K}\cdot\mathbf{K}^{-1}\cdot\partial_{\boldsymbol{\vartheta}'}\mathbf{K} - \mathbf{K}^{-1}\cdot\partial_{\boldsymbol{\vartheta}}\partial_{\boldsymbol{\vartheta}'}\mathbf{K}).\end{aligned} \tag{2.19}$$

Again, these are not the same as the derivatives in (2.9). These expressions for the gradient and the Hessian of the hyperlikelihood, maximized or marginalized over σ_f^2, share the same advantages as the analogous expressions in §2: they may be evaluated in $\mathcal{O}(n^2)$ time once the hyperlikelihood itself has been evaluated in $\mathcal{O}(n^3)$ time.

3. Numerical results

In order to perform model comparison calculations between competing covariance functions, we must first specify at least two different covariance functions. We choose the two functions in (3.1) and (3.2), where $(t,t') \equiv (x,x')$. These functions are both based on the periodic covariance function proposed by [2]. The first function k_1 is the product of a single periodic component with time scale T_1 and a simple compact-support polynomial covariance function [18] to describe any non-periodic component of the data. The choice of a compact-support covariance function is especially useful when working with large datasets; this is precisely the situation where the techniques described above are also designed to be of maximum benefit. The second function k_2 includes an additional periodic component with time scale T_2. In order to avoid double-counting in k_2, we impose the constraint $T_2 \geq T_1$. Both covariance functions also include an uncorrelated noise term; we define this in such a way that σ_f remains an overall scale

[4]Here, $\hat{\sigma}_f$ retains its maximum $\ln P$ value from (2.15), although the new maximum $\ln P_{\mathrm{marg}}$ value has actually now shifted to $(\hat{\sigma}_f')^2 = n\hat{\sigma}_f^2/(n-1)$ due to the effect of the hyperprior. For large datasets ($n \gg 1$), the difference between the two is negligible (the hyperprior becomes uninformative as it is overwhelmed by the hyperlikelihood).

hyperparameter which can be maximized or marginalized over analytically as described in §3.2.

$$k_1(t,t') = \sigma_f^2 C\left(\frac{|t-t'|}{T_0}\right)\exp\left[-\frac{2}{l_1^2}\sin^2\left(\frac{\pi(t-t')}{T_1}\right)\right] + \sigma_f^2\sigma_n^2\delta_{tt'}, \tag{3.1}$$

$$k_2(t,t') = \sigma_f^2 C\left(\frac{|t-t'|}{T_0}\right)\exp\left[-\frac{2}{l_1^2}\sin^2\left(\frac{\pi(t-t')}{T_1}\right) - \frac{2}{l_2^2}\sin^2\left(\frac{\pi(t-t')}{T_2}\right)\right] + \sigma_f^2\sigma_n^2\delta_{tt'} \tag{3.2}$$

and

$$C(\tau) = \begin{cases} (1-\tau)^5\dfrac{48\tau^2+15\tau+3}{3} & \tau < 1 \\ 0 & \tau > 1 \end{cases}. \tag{3.3}$$

The covariance functions are completely specified by the hyperparameters σ_f (overall scale), T_j ($j = 0, 1, 2$; time scales) and l_j ($j = 1, 2$; smoothing parameters for the periodic components). The noise parameter σ_n could also be taken to be a hyperparameter; instead, for simplicity, we here take σ_n to be fixed. As σ_n appears in k multiplied by the overall scale, σ_f, fixing σ_n is roughly equivalent to specifying a fixed fractional error.

We want to perform model comparison using the Laplace approximation outlined previously. This technique requires reparametrizing the covariance function such that the hyperpriors are flat. For the time-scale hyperparameters, which are dimensionful, we choose to use the scale-invariant Jeffreys prior, $P(T_j) \propto 1/T_j$. This prior is improper if the range of T_j is $(0, \infty)$, so we restrict the range to $(\delta t, \Delta T)$, where δt and ΔT are respectively the smallest and largest separations between the sampling points. If there was a time scale in the problem outside of this range, we would be unable to resolve it from the data. We now seek a transformation $\phi_j \equiv \phi_j(T_j)$ to a new hyperparameter ϕ_j such that the prior is flat in this parameter, $P(\phi_j) = \text{const}$. The conservation of probability gives a differential equation relating the two

$$P(T_j)\,\mathrm{d}T_j = P(\phi_j)\,\mathrm{d}\phi_j \;\;\Rightarrow\;\; T_j = \exp\left(\frac{\phi_j}{A_j}\right), \quad \{j = 0, 1, 2\}, \tag{3.4}$$

where the A_js are constants which we can set equal to 1. The range of these new hyperparameters is $\phi_j \in (\ln(\delta t), \ln(\Delta T))$ and $P(\phi_j) = 1/\ln(\Delta T/\delta t)$.

For the smoothness parameters l_j, we choose to use lognormal priors, $P(l_j) = \exp[-(\mu - \log l_j)^2/(2\sigma_l^2)]\Big/\sqrt{2\pi\sigma_l^2}$, with mean $\mu = 1$ and variance $\sigma_l^2 = 4$. As before, we seek a transformation to some new hyperparameters ξ_j in which the prior is flat. The desired transformation is given by

$$l_j = \exp[\mu + \sqrt{2}\sigma_l \mathrm{erf}^{-1}(2\xi_j)], \quad \{j = 1, 2\}, \tag{3.5}$$

where $\xi_j \in (-0.5, 0.5)$.

3.1. Synthetic data

Shown in figure 1 are realizations of GPs with covariance functions k_1 and k_2.[5] In order to perform test model comparison calculations, a realization of the k_2 GP with n points was drawn and analysed using both the k_1 and k_2 covariance functions. For each covariance, the peak hyperlikelihood was found by numerically maximizing (2.14) using a conjugate gradient method, making use of the gradient in (2.17). The hyperevidence was estimated using (2.13) and the expression for the Hessian in (2.19); the results are summarized in table 1. To verify the accuracy of this estimate, the hyperevidence was also integrated numerically using MULTINEST [19–21], which implements a nested sampling algorithm [22]. This was repeated for three different values of n (in the case $n = 100$, the synthetic data are plotted in the right-hand panel of figure 1), and the results are also summarized in table 1.

From table 1 it can be seen that as n is increased, the Bayes factors increasingly favour the more complicated covariance function (and in this case the correct covariance function from which the data was drawn). In almost all cases, the Laplace approximation gives a value $\ln \mathcal{Z}_{\text{est}}$ which is in agreement at better than 2σ with the numerically integrated value $\ln \mathcal{Z}_{\text{num}}$. There is one exception which is highlighted in italic; this occurs for the most complicated covariance function (with the largest number of hyperparameters) and when the number of data points is smallest. In this situation, it would be expected that the posterior distribution on the hyperparameters may be highly multimodal and/or exhibit strong degeneracies (both of these expectations were confirmed by examining the posterior distribution on the

[5]The code, optimized for use on a GPU, and the synthetic used to produce these numerical results is available at http://hdl.handle.net/10283/1924.

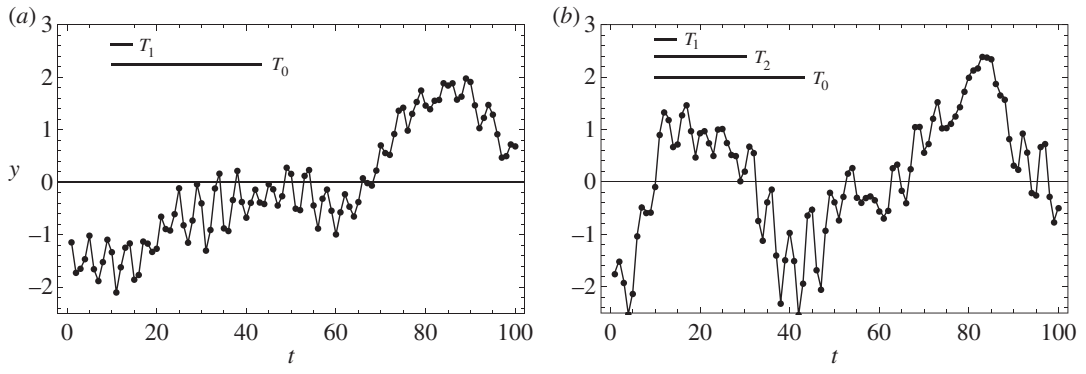

Figure 1. Realizations of the GPs $k_1(t, t')$ and $k_2(t, t')$ from (3.1) and (3.2) for values of $t = 1, 2, 3 \ldots, 100$ are shown in the (*a*,*b*) panels, respectively. The horizontal black lines indicate the length scales associated with the different terms in the covariance functions. The hyperparameters for k_1 were chosen to be $\sigma_f = 1$, $\phi_0 = 3.5$, $\phi_1 = 1.5$ and $x_1 = 0$. The hyperparameters for k_2 were chosen to be the same as for k_1 and $\phi_2 = 3$ and $x_2 = 0$. In both cases, the noise was fixed to $\sigma_n = 10^{-2}$.

Table 1. A summary of the results of the analysis of synthetic data for three different-sized datasets. The first set of two columns is for a dataset drawn from the k_2 covariance function and analysed with the k_1 covariance function. The first column is the estimated hyperevidence using the Laplace approximation where $\mathcal{Z}$ is as given in equation (2.13), while the second is the numerically calculated hyperevidence. The second set of two columns shows results for the same data, but analysed with the k_2 covariance function. The final pair of columns shows the log Bayes factor, $\ln \mathcal{B} \equiv \ln \mathcal{Z}^{k_2} - \ln \mathcal{Z}^{k_1}$, calculated using the approximate and numerical values for the hyperevidence.

n	$\ln \mathcal{Z}^{k_1}_{\text{est}}$	$\ln \mathcal{Z}^{k_1}_{\text{num}}$	$\ln \mathcal{Z}^{k_2}_{\text{est}}$	$\ln \mathcal{Z}^{k_2}_{\text{num}}$	$\ln \mathcal{B}_{\text{est}}$	$\ln \mathcal{B}_{\text{num}}$
30	−17.77	−17.87 ± 0.08	*−18.82*	−17.73 ± 0.09	−1.05	0.14 ± 0.12
100	−20.17	−20.17 ± 0.10	−19.22	−19.22 ± 0.11	0.95	0.95 ± 0.15
300	−49.94	−50.12 ± 0.11	−40.21	−40.36 ± 0.13	9.73	9.76 ± 0.17

hyperparameters returned by MULTINEST). This exceptional case serves to highlight situations in which the Laplace approximation should not be trusted. The MULTINEST posteriors in all other cases were verified to be well approximated by a single Gaussian mode. Figure 2 shows the posterior distribution for the parameters of k_2 obtained from the largest ($n = 300$) synthetic dataset.

Our method of model comparison is proposed as a faster alternative to model comparison using numerically evaluated Bayes factors. Simply comparing the peak hyperlikelihood (marginal likelihood) values would also give a measure of the goodness of fit, but this tends to favour more complex models and incurs the risk of overfitting. More sophisticated methods of model selection exist in the literature (see [23,24] and references within), e.g. the comparison of models based on estimated predictive criteria [25–27], or the construction of a larger reference model and the subsequent selection of a simpler submodel with similar predictions [28,29]. A detailed numerical comparison to these methods is left for future investigation.

The $\ln \mathcal{Z}_{\text{num}}$ values in table 1, evaluated using MULTINEST, required between 20 000 and 50 000 likelihood evaluations. The maximization routines typically took fewer than 100 likelihood evaluations to find the peak, and then one additional evaluation to calculate the Hessian and hence $\ln \mathcal{Z}_{\text{est}}$. In order to guard against the possibility of the maximization routines becoming trapped in local maxima, as opposed to the global maximum, the algorithm was run multiple times from randomly selected starting positions. The typical number of runs required to find the global maximum was approximately 10. After these duplicate runs are accounted for, the speed-up factor in calculating $\ln \mathcal{Z}_{\text{est}}$ compared with $\ln \mathcal{Z}_{\text{num}}$ was between 20 and 50 in all cases.

3.2. Tidal data from Woods Hole

In order to illustrate the effectiveness of the techniques described above on real data, we consider several tidal datasets of different sizes from Woods Hole, MA, USA [30].[6] We consider the mean sea-level offset

[6]Data from the National Oceanic and Atmospheric Administration, http://tidesandcurrents.noaa.gov/waterlevels.html (accessed August 2015).

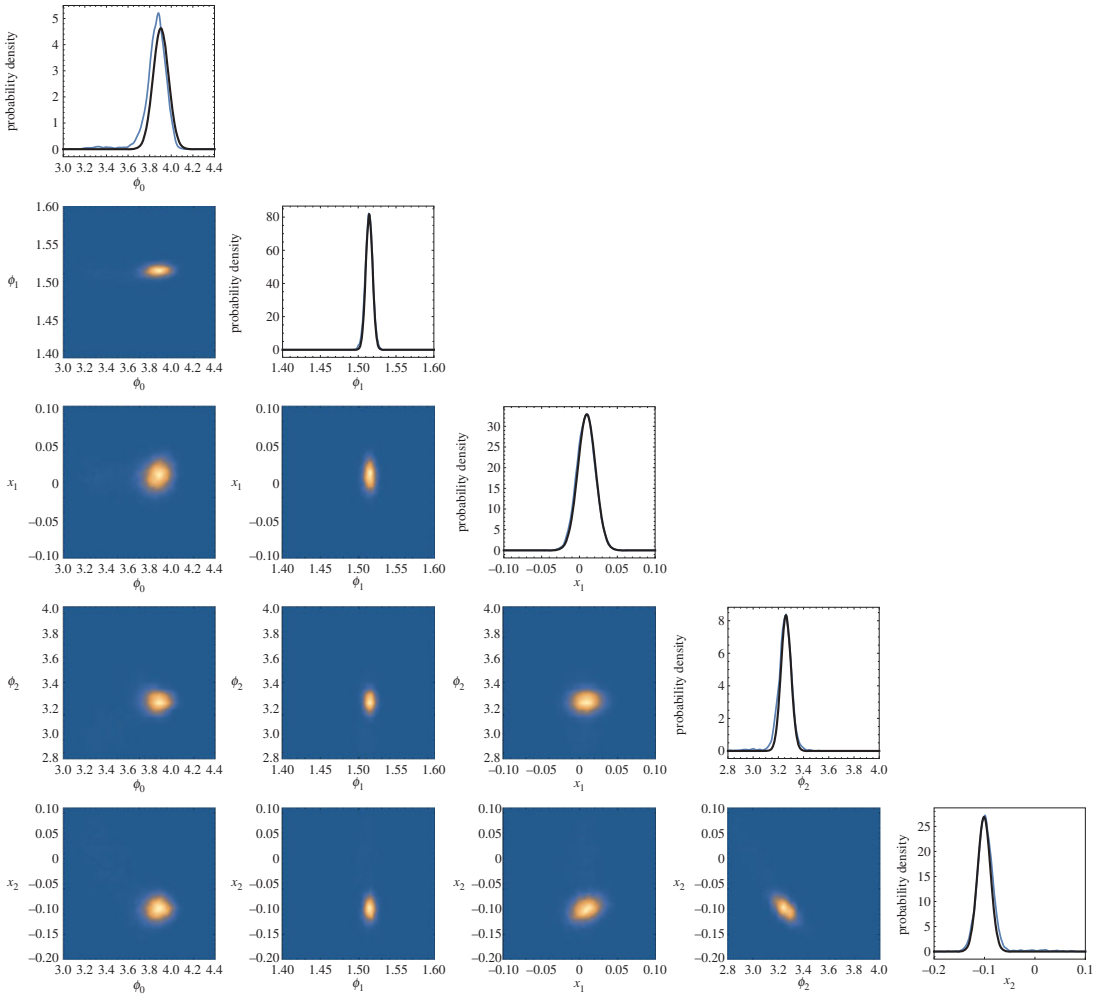

Figure 2. The one- and two-dimensional marginalized posterior distributions on the hyperparameters of the k_2 covariance function obtained from the largest ($n = 300$) synthetic dataset. The posterior is well approximated as a normal distribution. Shown in the black curves in the one-dimensional marginalized posterior distributions along the diagonal are the normal approximations obtained by using the techniques described in §2 to maximize and find the Hessian. Using the Hessian to approximate the integral of this distribution (the hyperevidence) leads to an error of approximately 10% (table 1).

recorded between 3 January and 15 June 2014, six lunar months, sampled at 2 h intervals (giving $n = 1968$ data points). This is plotted in figure 3. We also consider a smaller subset of the data (the first lunar month), with $n = 328$ data points.[7]

We interpolate the data using the two covariance functions in (3.1) and (3.2); these functions are well suited to the data, as we expect the sea level to contain harmonics of the various time scales associated with the daily, monthly and yearly cycles of the tides. For simplicity, we fix $\sigma_n = 10^{-2}$, which is the typical fractional error in the sea-level measurements. As in §3.1, we reparametrize the covariance functions so that the T_j have Jeffreys priors, and the smoothness parameters have lognormal priors. We use a conjugate gradient maximization algorithm with (2.17) and the Hessian in (2.19) to evaluate the volume in (2.13) and perform model comparison between the two covariance functions.

For the smaller dataset, we find the time scale $T_1 = (12.8 \pm 0.2)$ h with k_1, which corresponds to the two main tides per day. With k_2, we find the time scales $T_1 = (12.44 \pm 0.07)$ h and $T_2 = (24.3 \pm 1.0)$ h; the second time scale corresponds to the height difference between the first and second tides of the day. The two time-scale model is highly favoured with a log Bayes factor of 57.8.

[7]This dataset is regularly sampled in time, and therefore, the covariance matrix will be a Toeplitz matrix. This structure could be exploited to accelerate the inversion of the covariance matrix; we choose not to use this here so that our code can be applied to irregularly sampled data in arbitrary dimensions.

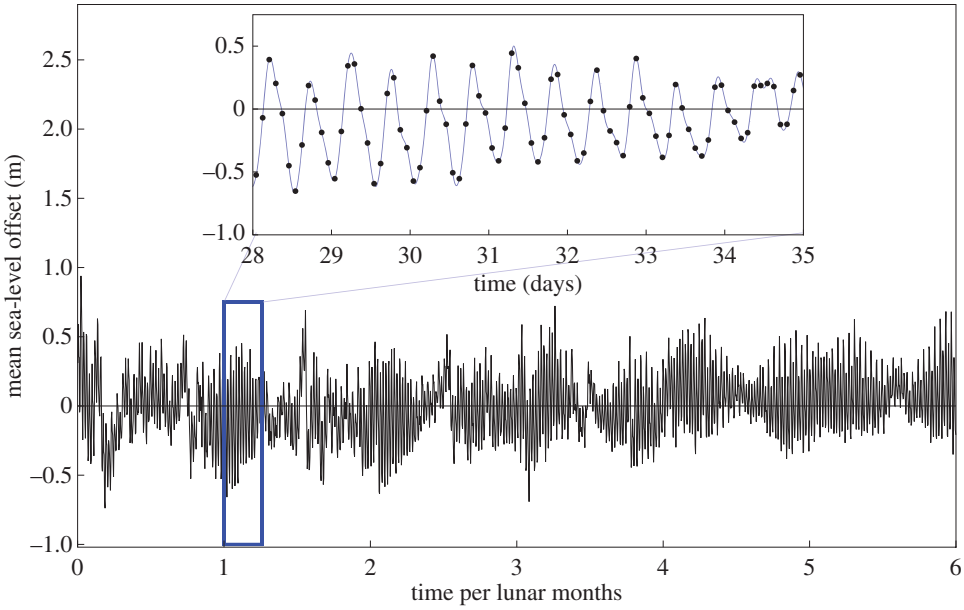

Figure 3. Shown in the main figure are six lunar months of tidal height data (black), from which the lunar tidal cycle can be discerned. Shown in the inset plot are several days of the tidal data (black points), from which the daily cycles can be clearly seen. Overlaid in the inset plot are both GP interpolants (blue), which are identical on this time scale.

For the larger dataset, we find $T_1 = (12.80 \pm 0.11)$ h with k_1, and $T_1 = (12.40 \pm 0.03)$ h and $T_2 = (23.3 \pm 0.3)$ h with k_2. In all cases, the (squared) errors are estimated using the diagonal components of the inverse Hessian; it can be seen that the time scales are more precisely measured for the larger dataset, as expected. The two time-scale model is even more conclusively favoured for the larger dataset, with a log Bayes factor of 538. We also find a number of subsidiary hyperlikelihood peaks associated with other time scales in the data, but all subsidiary peaks are strongly suppressed relative to the global peak (by at least $\Delta \ln P$ of approx. 100) and so we expect our Bayes factor estimates to be robust.

Sea-level data are known to contain a large number of different frequencies, which necessitates the use of harmonic analysis in tidal modelling; the number of constituents included in tide prediction calculations has increased from tens [31] to thousands [32] over the past century. Clearly any k_2-like covariance function with fewer than 10 time scales is simplistic, but the construction of a more detailed tidal model is beyond the scope of this paper.[8]

The number of evaluations of (2.13) needed to obtain these results was comparable with the numbers for the synthetic data discussed in §3.1. However, each evaluation here was more expensive (approx. 10 s) due to the size of the dataset. Based on the speed-ups found in §3.1, it would be expected that MULTINEST would take up to approximately one week to calculate the Bayes factor.

Shown in the inset plot in figure 3 are the two interpolants from k_1 and k_2 for the larger dataset, which both perform equally well on the time scale of one week. These interpolants, which are the result of the regression analysis, may be used to estimate the tidal height at a time where a measurement is not available.

4. Summary

We have described some simple ways in which the computationally expensive training stage of implementing GPR can be accelerated. The analytic maximization of the hyperlikelihood over a single-scale hyperparameter of the covariance function aids in speeding up the maximization of the hyperlikelihood by reducing the dimensionality of the problem; the advantages of this will be most keenly felt in (common) problems where relatively simple covariance functions are used. Meanwhile, the analytic evaluation of the Hessian matrix, either in the manner of (2.9) or (2.19), aids in speeding up the process of model comparison between different types of covariance function. We have successfully

[8]We have conducted preliminary investigations of a three time-scale model. The hyperlikelihood surface for this covariance function is more structured and non-Gaussian than for k_2 and k_1. Estimates of the Bayes factors indicate that the inclusion of additional time scales is favoured, as expected based on the known large number of modes present.

demonstrated these techniques on a synthetic dataset where the data was drawn from one of two covariance functions under consideration. In the case of the synthetic data, the size of the dataset was limited to fewer than 300 points, so that the results could be verified by using the MULTINEST algorithm to numerically sample and integrate the posteriors. We also demonstrated the techniques by applying them to a larger real dataset of mean sea-level measurements, where the full MULTINEST calculation would have taken too long to perform. It is to be hoped that these techniques will aid in the wider application of GP methods to larger datasets.

Data accessibility. The original source of the tidal data is the National Oceanic and Atmospheric Administration (http://tidesandcurrents.noaa.gov/waterlevels.html, accessed August 2015). The subset of data used here, as well as our synthetic datasets and our code is available at http://hdl.handle.net/10283/1924.

Authors' contributions. C.J.M. devised the study, jointly performed the analysis and wrote the manuscript. A.J.K.C. jointly performed the analysis and wrote the manuscript. C.P.L.B. helped with the analysis, and the writing and editing of the manuscript. J.R.G. helped devise the study and edited the manuscript. All authors gave final approval for the publication.

Competing interests. We have no competing interests.

Funding. C.J.M. and C.P.L.B. are supported by the STFC. A.J.K.C.'s work is supported by the Cambridge Commonwealth, European and International Trust. J.R.G.'s work is supported by the Royal Society.

References

1. Rasmussen CE, Williams CKI. 2006 *Gaussian processes for machine learning*. Cambridge, MA: MIT Press.
2. MacKay DJC. 2003 *Information theory, inference and learning algorithms*. Cambridge, UK: Cambridge University Press.
3. Barry D. 1986 Nonparametric Bayesian regression. *Ann. Stat.* **14**, 934–953. (doi:10.1214/aos/1176350043)
4. Wahba G. 1978 Improper priors, spline smoothing and the problem of guarding against model errors in regression. *J. R. Stat. Soc. B* **40**, 364–372.
5. O'Hagan A, Kingman JFC. 1978 Curve fitting and optimal design for prediction. *J. R. Stat. Soc. B* **40**, 1–42.
6. Smola AJ, Bartlett PL. 2001 Sparse greedy Gaussian process regression. in *Advances in neural information processing systems*, vol. 13 (eds TK Leen, T Dietterich, V Tresp), pp. 619–625. Cambridge, MA: MIT Press.
7. Quiñonero-Candela J, Rasmussen CE. 2005 A unifying view of sparse approximate Gaussian process regression. *J. Mach. Learn. Res.* **6**, 1939–1959.
8. Cressie N, Johannesson G. 2008 Fixed rank kriging for very large spatial data sets. *J. R. Stat. Soc. B* **70**, 209–226. (doi:10.1111/j.1467-9868.2007.00633.x)
9. Banerjee A, Dunson DB, Tokdar ST. 2013 Efficient Gaussian process regression for large datasets. *Biometrika* **100**, 75–89. (doi:10.1093/biomet/ass068)
10. Moore CJ, Berry CPL, Chua AJK, Gair JR. 2016 Improving gravitational-wave parameter estimation using Gaussian process regression. *Phys. Rev. D* **93**, 064001. (doi:10.1103/PhysRevD.93.064001)
11. Moore CJ, Gair JR. 2014 Novel method for incorporating model uncertainties into gravitational wave parameter estimates. *Phys. Rev. Lett.* **113**, 251101. (doi:10.1103/PhysRevLett.113.251101)
12. MacKay DJC. 1999 Comparison of approximate methods for handling hyperparameters. *Neural Comput.* **11**, 1035–1068. (doi:10.1162/089976699300016331)
13. Snelson E, Ghahramani Z. 2006 Sparse Gaussian processes using pseudo-inputs. In *Advances in neural information processing systems*, vol. 18 (eds Y Weiss, B Schölkopf, JC Platt), pp. 1257–1264. Cambridge, MA: MIT Press.
14. Quiñonero-Candela J, Rasmussen CE, Williams CKI. 2007 Approximation methods for Gaussian process regression. In *Large-scale Kernel machines* (eds L Bottou, O Chapelle, D DeCoste, J Weston), ch. 9, pp. 203–223. Cambridge, MA: MIT Press.
15. Blum M, Riedmiller M. 2013 Optimization of Gaussian process hyperparameters using Rprop. In *European Symp. on Artificial Neural Networks, Computational Intelligence and Machine Learning*, p. 339. ESANN.
16. MacKay DJC. 1996 Hyperparameters: optimize, or integrate out? In *Maximum entropy and Bayesian methods* (ed. GR Heidbreder). *Fundamental theories of physics*, vol. 62, pp. 43–59. Dordrecht, The Netherlands: Springer.
17. Jeffreys H. 1946 An invariant form for the prior probability in estimation problems. *Proc. R. Soc. A* **186**, 453–461. (doi:10.1098/rspa.1946.0056)
18. Wendland H. 2005 *Scattered data approximation*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge, UK: Cambridge University Press.
19. Feroz F, Hobson MP, Bridges M. 2009 MultiNest: an efficient and robust Bayesian inference tool for cosmology and particle physics. *Mon. Not. R. Astron. Soc.* **398**, 1601–1614. (doi:10.1111/j.1365-2966.2009.14548.x)
20. Feroz F, Hobson MP. 2008 Multimodal nested sampling: an efficient and robust alternative to Markov Chain Monte Carlo methods for astronomical data analyses. *Mon. Not. R. Astron. Soc.* **384**, 449–463. (doi:10.1111/j.1365-2966.2007.12353.x)
21. Feroz F, Hobson MP, Cameron E, Pettitt AN. 2013 Importance nested sampling and the multinest algorithm. (http://xxx.lanl.gov/abs/1306.2144)
22. Skilling J. 2006 Nested sampling for general Bayesian computation. *Bayesian Anal.* **1**, 833–859. (doi:10.1214/06-BA127)
23. O'Hara RB, Sillanpää MJ. 2009 A review of Bayesian variable selection methods: what, how and which. *Bayesian Anal.* **4**, 85–117. (doi:10.1214/09-BA403)
24. Piironen J, Vehtari A. 2015 Comparison of Bayesian predictive methods for model selection. (http://arxiv.org/abs/1503.08650)
25. Geisser S, Eddy WF. 1979 A predictive approach to model selection. *J. Am. Stat. Assoc.* **74**, 153–160. (doi:10.1080/01621459.1979.10481632)
26. Watanabe S. 2009 *Algebraic geometry and statistical learning theory*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge, UK: Cambridge University Press.
27. Gelman A, Hwang J, Vehtari A. 2013 Understanding predictive information criteria for Bayesian models. *Stat. Comput.* **24**, 997–1016. (doi:10.1007/s11222-013-9416-2)
28. Lindley DV. 1968 The choice of variables in multiple regression (with discussion). *J. R. Stat. Soc. B* **30**, 31–66.
29. San Martini A, Spezzaferri F. 1984 A predictive model selection criterion. *J. R. Stat. Soc. B* **46**, 296–303.
30. Scherer W *et al.* 2001 NOAA Special Publication NOS CO-OPS 1, National Oceanic and Atmospheric Administration, Silver Spring, MD, USA.
31. Lisitzin E. 1974 *Sea-level changes*. Oxford, UK: Elsevier.
32. Casotto S, Biscani F. 2004 A fully analytical approach to the harmonic development of the tide-generating potential accounting for precession, nutation, and perturbations due to figure and planetary terms. In *AAS/Division of Dynamical Astronomy Meeting #35*. Bulletin of the American Astronomical Society, vol. 36, p. 862.