Variational Model Selection for Sparse Gaussian Process Regression

Michalis K. Titsias
School of Computer Science,
University of Manchester, UK
mtitsias@cs.man.ac.uk

Abstract

Sparse Gaussian process methods that use inducing variables require the selection of the inducing inputs and the kernel hyperparameters. We introduce a variational formulation for sparse approximations that jointly infers the inducing inputs and the kernel hyperparameters by maximizing a lower bound of the true log marginal likelihood. The key property of this formulation is that the inducing inputs are defined to be variational parameters which are selected by minimizing the Kullback-Leibler divergence between the variational distribution and the exact posterior distribution over the latent function values. We apply this technique to regression and we compare it with other approaches in the literature.

1 Introduction

Gaussian processes (GPs) are stochastic processes that, in the context of Bayesian statistics, can be used as non-parametric priors over real-valued functions that can be combined with data to give posterior processes over these functions (O'Hagan, 1978; Wahba, 1990). In machine learning GPs offer a Bayesian kernel-based framework for solving supervised learning tasks such as regression and classification; see e.g. (Rasmussen and Williams, 2006).

However, the application of GP models is intractable for large datasets because the time complexity scales as $O(n^3)$ and the storage as $O(n^2)$ where n is the number of training examples. To overcome this limitation, many approximate or sparse methods have been proposed in the literature (Williams and Seeger, 2001; Smola and Bartlett, 2001; Csato and Opper, 2002; Lawrence et al., 2002; Seeger et al., 2003; Schwaighofer and Tresp, 2003; Keerthi and Chu, 2006; Snelson and Ghahramani, 2006; Quiñonero-Candela and Rasmussen, 2005; Walder et al., 2008). These methods construct an approximation based on a small set of m support or inducing variables that allow the reduction of the time complexity from $O(n^3)$ to $O(nm^2)$. They mainly differ in the strategies they use to select the inducing inputs which are typically selected from the training or test examples. Snelson and Ghahramani (2006) allow the inducing variables to be considered as auxiliary pseudo-inputs that are inferred along with kernel hyperparameters using continuous optimization.

The selection of inducing variables and kernel hyperparameters in a sparse GP method can be approached as a model selection problem. Therefore, the most relevant criterion for solving such problem is an approximation to the exact (intractable) marginal likelihood that can be maximized over inducing inputs and hyperparameters. Existing state-of-the-art methods (Snelson and Ghahramani, 2006; Seeger et al., 2003) derive such approximations by modifying the likelihood function or the GP prior (Quiñonero-Candela and Rasmussen, 2005) and then computing the marginal likelihood of the modified model. This approach turns the inducing inputs into additional kernel hyperparameters. While this can increase flexibility when we fit the data, it can also lead to overfitting when we optimize with respect to all unknown hyperparameters. Furthermore, fitting a modified model to the data is not so rigorous approximation procedure since there is no distance or divergence between the exact and the modified model that is minimized.

In this paper we introduce a variational method for sparse GP models that jointly selects the inducing inputs and the hyperparameters by maximizing a lower bound to the exact marginal likelihood. In this formulation we do not modify the likelihood or the GP prior in the training examples. Instead we follow the standard variational approach (Jordan et al., 1999) according to which a variational distribution is used to approximate the exact posterior over the latent function values. The important difference between this formulation and previous methods is that here the inducing inputs are defined to be variational parameters which are selected by minimizing the Kullback-Leibler (KL) divergence. This allows i) to avoid overfitting and ii) to rigorously approximate the exact GP model by minimizing a divergence between the sparse model and the exact one. The selection of the inducing inputs and hyperparameters is achieved either by assuming pseudo-inputs and applying continuous optimization over all unknown quantities, similarly to (Snelson and Ghahramani, 2006), or by using a variational EM algorithm where at the E step we greedily select the inducing inputs from the training data and at the M step we update the hyperparameters. In contrast to previous greedy approaches, e.g. (Seeger et al., 2003), our scheme monotonically increases the optimized objective function.

We apply the variational method to regression with additive Gaussian noise and we compare its performance to training schemes based on the projected process marginal likelihood (Seeger et al., 2003; Csato and Opper, 2002) and the sparse pseudo-inputs marginal likelihood (Snelson and Ghahramani, 2006).

Our method is most closely related to the variational sparse GP method described in (Csato, 2002; Csato and Opper, 2002; Seeger, 2003) that is applied to GP classification (Seeger, 2003). The main difference between our formulation and these techniques is that we maximize a variational lower bound in order to select the inducing inputs, while these methods use variational bounds for estimating only the kernel hyperparameters. This paper is a revised and extended version of (Titsias, 2009).

2 Gaussian process regression and sparse methods

A GP is a collection of random variables $\{\mathbf{f}(\mathbf{x}) | \mathbf{x} \in \mathcal{X}\}$, where $\mathcal{X}$ is an index or input set, for which any finite subset follows a Gaussian distribution. A GP is fully specified by the mean function $m(\mathbf{x})$ and the covariance or kernel function $k(\mathbf{x}, \mathbf{x}')$ defined by

$$m(\mathbf{x}) = E[f(\mathbf{x})], \tag{1}$$

$$k(\mathbf{x}, \mathbf{x}') = E[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))]. \tag{2}$$

In the context of regression and statistical learning, a GP can be used as a non-parametric prior over a real-valued function which can be combined with data to give a posterior over the function. Suppose that we wish to estimate a real-valued function $f(\mathbf{x})$, where for simplicity the input set $\mathcal{X}$ is taken to be the D-dimensional real space $\mathbb{R}^D$. We shall call $f(\mathbf{x})$ the unobserved or latent function. We further assume that the mean function of the GP prior is zero and the covariance function is specified through a set of hyperparameters $\boldsymbol{\theta}$. Suppose we collect a training dataset, $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, consisting of n noisy realizations of the latent function where each scalar y_i is obtained by adding Gaussian noise to $f(\mathbf{x})$ at input $\mathbf{x}_i$, i.e.

$$y_i = f_i + \epsilon_i, \quad \epsilon_i \sim N(0, \sigma^2), \quad f_i = f(\mathbf{x}_i).$$

We denote by X the $n \times D$ matrix of all training inputs, $\mathbf{y} = [y_1 \dots y_n]^T$ the vector of all outputs and $\mathbf{f} = [f_1 \dots f_n]^T$ the corresponding vector of the *training* latent function values. The marginalization property of GPs allows us to simplify the (initially infinite dimensional) prior so that after marginalizing out all function points not associated with the data, we obtain a n-dimensional Gaussian distribution, $p(\mathbf{f}) = N(\mathbf{f}|0, K_{nn})$, where $\mathbf{0}$ denotes the n-dimensional zero vector and K_{nn} is the $n \times n$ covariance matrix obtained by evaluating the kernel function on the observed inputs. The joint probability model of observed output and latent variables data can be written as

$$p(\mathbf{y}, \mathbf{f}) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}),$$

where $p(\mathbf{y}|\mathbf{f})$ is the likelihood function and $p(\mathbf{f})$ the GP prior. Notice that this is a conditional model since we condition on the observed inputs X. Nevertheless, for the sake of clarity we omit reference to X and the hyperparameters[1] throughout the paper. The training dataset induce a posterior process over the latent function $f(\mathbf{x})$ which (because of the Gaussian likelihood) is also a GP specified by a posterior mean function and a posterior covariance function. We can easily work out the functional form of these functions, e.g. by augmenting the marginal probability $p(\mathbf{y})$ of the observed outputs by two latent function values $f(\mathbf{x})$ and $f(\mathbf{x}')$, and find that

$$m_{\mathbf{y}}(\mathbf{x}) = K_{\mathbf{x}n}(\sigma^2 I + K_{nn})^{-1}\mathbf{y}, \tag{3}$$

$$k_{\mathbf{y}}(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}') - K_{\mathbf{x}n}(\sigma^2 I + K_{nn})^{-1}K_{n\mathbf{x}'}.$$

Here, $K_{\mathbf{x}n} = [k(\mathbf{x}, \mathbf{x}_1) \dots k(\mathbf{x}, \mathbf{x}_n)]$ is an n-dimensional row vector of kernel function values between $\mathbf{x}$ and the training inputs and $K_{n\mathbf{x}} = K_{\mathbf{x}n}^T$. Any query related to the posterior GP can be answered by the above mean and covariance functions. For instance, the Gaussian posterior distribution $p(\mathbf{f}|\mathbf{y})$ on the training latent variables $\mathbf{f}$ is computed by evaluating eq. (3) at the inputs X. Similarly the prediction of the output $y_* = f_* + \epsilon_*$ at some unseen input $\mathbf{x}_*$ is described by $p(y_*|\mathbf{y}) = N(y_*|m_{\mathbf{y}}(\mathbf{x}_*), k_{\mathbf{y}}(\mathbf{x}_*, \mathbf{x}_*) + \sigma^2)$. The posterior GP depends on the values of the hyperparameters $(\boldsymbol{\theta}, \sigma^2)$ which can be estimated by maximizing the log marginal likelihood given by

$$\log p(\mathbf{y}) = \log[N(\mathbf{y}|\mathbf{0}, \sigma^2 I + K_{nn})]. \tag{4}$$

Once we have obtained point estimates for the hyperparameters $(\sigma^2, \boldsymbol{\theta})$ by maximizing the above log marginal likelihood, we can use these estimates in eq. (3) in order to make prediction in unseen input points.

Although the above GP framework is elegant, it requires $O(n^3)$ computations as clearly we need to

[1] A precise notation is to write $p(\mathbf{y}, \mathbf{f}|X, \sigma^2, \boldsymbol{\theta}) = p(\mathbf{y}|\mathbf{f}, \sigma^2)p(\mathbf{f}|X, \boldsymbol{\theta})$.

invert a matrix of size $n \times n$; once when we evaluate the prediction in eq. (3) and multiple times when we maximize the marginal likelihood in eq. (4). Therefore, we need to consider approximate or sparse methods in order to deal with large datasets. Advanced sparse methods use a small set of m function points as support or inducing variables. This yields a time complexity that scales as $O(nm^2)$. Some important issues in these methods involve the selection of the inducing variables and the hyperparameters. For reviews of current approaches see chapter 8 in (Rasmussen and Williams, 2006) and (Quiñonero-Candela and Rasmussen, 2005). In section 3, we propose a variational framework to deal with the selection of the inducing variables and hyperparameters. Therefore, in the remaining of this section we analyze the most relevant previous methods which are based on maximizing an approximate marginal likelihood obtained by modifying the GP prior (Quiñonero-Candela and Rasmussen, 2005).

Suppose we wish to use m inducing variables to construct our sparse GP method. The inducing variables are latent function values evaluated at some inputs X_m. X_m can be a subset of the training inputs or auxiliary pseudo-points (Snelson and Ghahramani, 2006). Learning X_m and the hyperparameters $(\boldsymbol{\theta}, \sigma^2)$ is the crucial problem we need to solve in order to obtain a sparse GP method. An approximation to the true log marginal likelihood in eq. (4) can allow us to infer these quantities. The current state-of-the-art approximate marginal likelihood is given in the sparse pseudo-inputs GP method (SPGP) proposed in (Snelson and Ghahramani, 2006). A related objective function used in (Seeger et al., 2003) corresponds to the projected process approximation (PP). These approximate log marginal likelihoods have the form

$$\boxed{F = \log[N(\mathbf{y}|\mathbf{0}, \sigma^2 I + Q_{nn})],} \tag{5}$$

where Q_{nn} is an approximation to the true covariance K_{nn}. In PP, $Q_{nn} = K_{nm} K_{mm}^{-1} K_{mn}$, i.e. the exact covariance is replaced by the Nyström approximation. Here, K_{mm} is the $m \times m$ covariance matrix on the inducing inputs, K_{nm} is the $n \times m$ cross-covariance matrix between training and inducing points and $K_{mn} = K_{nm}^T$. In SPGP, $Q_{nn} = \text{diag}[K_{nn} - K_{nm} K_{mm}^{-1} K_{mn}] + K_{nm} K_{mm}^{-1} K_{mn}$, i.e. the Nyström approximation is corrected to be exact in the diagonal. By contrasting eq. (4) with (5), it is clear that F is obtained by modifying the GP prior. This implies that the inducing inputs X_m play the role of extra kernel hyperparameters (similar to $\boldsymbol{\theta}$) that parametrize the covariance matrix Q_{nn}. However, because the prior has changed, continuous optimization of F with respect to X_m does not reliably approximate the exact GP model because there is no any distance between the modified and the exact GP model that is minimized. Further, since F is heavily parametrized with the extra hyperparameters X_m, overfitting can occur especially when we jointly optimize over $(X_m, \boldsymbol{\theta}, \sigma^2)$. Despite all that, the flexibility introduced by the extra hyperparameters X_m can often be advantageous. For instance, unlike the exact GP model, the SPGP model can fit heteroscedastic noise in the output data (Snelson and Ghahramani, 2006).

In the next section, we propose a formulation for sparse GP regression that follows a different philosophy to what eq. (5) implies. Rather than modifying the exact GP model and maximizing the marginal likelihood of the modified model, we minimize the KL divergence between the exact posterior GP and a variational approximation. The inducing inputs X_m become now variational parameters which are selected so as the KL divergence is minimized.

3 Variational learning of inducing inputs

We wish to define a sparse method that directly approximates the mean and covariance functions in eq. (3) that characterize the posterior GP. This posterior GP can be also described by the predictive

Gaussian distribution

$$p(\mathbf{z}|\mathbf{y}) = \int p(\mathbf{z}|\mathbf{f})p(\mathbf{f}|\mathbf{y})d\mathbf{f},$$

where $\mathbf{z}$ is any finite set of function points, $p(\mathbf{z}|\mathbf{f})$ denotes the conditional GP prior and $p(\mathbf{f}|\mathbf{y})$ is the posterior distribution over the training latent function values. Suppose that we wish to approximate the above Bayesian integral by using a small set of m auxiliary inducing variables $\mathbf{f}_m$ evaluated at the pseudo-inputs X_m, which are independent from the training inputs. We further assume that $\mathbf{f}_m$ are function values drawn from the same GP prior[2] as the training function values $\mathbf{f}$. By using the augmented joint model $p(\mathbf{y}|\mathbf{f})p(\mathbf{z},\mathbf{f},\mathbf{f}_m)$, where $p(\mathbf{z},\mathbf{f}_m,\mathbf{f})$ is the GP prior jointly expressed over the function values $\mathbf{z}$, $\mathbf{f}$ and $\mathbf{f}_m$, we can equivalently write $p(\mathbf{z}|\mathbf{y})$ as

$$p(\mathbf{z}|\mathbf{y}) = \int p(\mathbf{z}|\mathbf{f}_m,\mathbf{f})p(\mathbf{f}|\mathbf{f}_m,\mathbf{y})p(\mathbf{f}_m|\mathbf{y})d\mathbf{f}d\mathbf{f}_m. \tag{6}$$

This expanded way of writing the predictive distribution is rather instructive since it indicates what one should expect from a good set of inducing variables and in fact it reveals the properties of an *optimal* set. More precisely, suppose that $\mathbf{f}_m$ is a *sufficient statistic* for the parameter $\mathbf{f}$ in the sense that $\mathbf{z}$ and $\mathbf{f}$ are independent given $\mathbf{f}_m$, i.e. it holds $p(\mathbf{z}|\mathbf{f}_m,\mathbf{f}) = p(\mathbf{z}|\mathbf{f}_m)$ for any $\mathbf{z}$. The above can be written as

$$\begin{aligned} q(\mathbf{z}) &= \int p(\mathbf{z}|\mathbf{f}_m)p(\mathbf{f}|\mathbf{f}_m)\phi(\mathbf{f}_m)d\mathbf{f}d\mathbf{f}_m \\ &= \int p(\mathbf{z}|\mathbf{f}_m)\phi(\mathbf{f}_m)d\mathbf{f}_m = \int q(\mathbf{z},\mathbf{f}_m)d\mathbf{f}_m, \end{aligned} \tag{7}$$

where $q(\mathbf{z}) = p(\mathbf{z}|\mathbf{y})$ and $\phi(\mathbf{f}_m) = p(\mathbf{f}_m|\mathbf{y})$. Here, $p(\mathbf{f}|\mathbf{f}_m) = p(\mathbf{f}|\mathbf{f}_m,\mathbf{y})$ is true since $\mathbf{y}$ is a noisy version of $\mathbf{f}$ and because of the assumption we made that any $\mathbf{z}$ is conditionally independent from $\mathbf{f}$ given $\mathbf{f}_m$[3]. The predictive distribution in eq. (7) requires reference to only m function points, the inducing variables $\mathbf{f}_m$, and it can be computed in $O(nm^2)$ time; see eq. (8) below. However, in practise the assumption of $\mathbf{f}_m$ being a sufficient statistic is unlikely to hold and we should expect $q(\mathbf{z})$ to be only an approximation to the exact predictive distribution $p(\mathbf{z}|\mathbf{y})$. In such case, and in order to think about how to optimize the quality of the approximation, we can let $\phi(\mathbf{f}_m)$ be a "free" variational Gaussian distribution, where in general $\phi(\mathbf{f}_m) \neq p(\mathbf{f}_m|\mathbf{y})$, that depends on a mean vector $\boldsymbol{\mu}$ and a covariance matrix A. Notice also that the quality of the approximation will crucially depend on the locations X_m of the inducing variables. By using eq. (7), we can read off the mean and covariance functions of the approximate posterior GP and obtain:

$$\begin{aligned} m_{\mathbf{y}}^q(\mathbf{x}) &= K_{\mathbf{x}m}K_{mm}^{-1}\boldsymbol{\mu}, \\ k_{\mathbf{y}}^q(\mathbf{x},\mathbf{x}') &= k(\mathbf{x},\mathbf{x}') - K_{\mathbf{x}m}K_{mm}^{-1}K_{m\mathbf{x}'} + K_{\mathbf{x}m}K_{mm}^{-1}AK_{mm}^{-1}K_{m\mathbf{x}'}. \end{aligned} \tag{8}$$

The above defines the general form of the sparse posterior GP which is tractably computed in $O(nm^2)$ time. The question that now arises is how can we select the ϕ distribution, i.e. $(\boldsymbol{\mu}, A)$, and the inducing inputs X_m. Notice that different sparse GP approaches such as the PP and SPGP methods, also called DTC and FITC in the unified view of Quiñonero-Candela and Rasmussen (2005), use different forms for the variational distribution ϕ and follow different strategies for the selection of the inducing inputs and hyperparameters.

[2] More general inducing variables are defined in section 6.

[3] From $p(\mathbf{z}|\mathbf{f}_m,\mathbf{y}) = \frac{\int p(\mathbf{y}|\mathbf{f})p(\mathbf{z},\mathbf{f}_m,\mathbf{f})d\mathbf{f}}{\int p(\mathbf{y}|\mathbf{f})p(\mathbf{z},\mathbf{f}_m,\mathbf{f})d\mathbf{f}d\mathbf{z}}$ and by using the fact $p(\mathbf{z}|\mathbf{f}_m,\mathbf{f}) = p(\mathbf{z}|\mathbf{f}_m)$, the result follows.

Next we apply a variational inference method that allows to jointly specify the quantities (X_m, ϕ) and crucially treat the inducing inputs as variational parameters which are rigorously selected by minimizing the KL divergence. As will be shown our method regarding the form of the ϕ distribution leads to the PP prediction (Csato and Opper, 2002; Seeger, 2003), while it gives a novel way of specifying the inducing inputs X_m and the hyperparameters $(\sigma^2, \boldsymbol{\theta})$ by maximizing a lower bound to the exact marginal likelihood.

3.1 The variational lower bound

To select the inducing inputs X_m, we intend to apply variational inference in an augmented probability space that involves both the training latent function values $\mathbf{f}$ and the pseudo-input inducing variables $\mathbf{f}_m$. Particularly, the initial joint model $p(\mathbf{y}, \mathbf{f})$ is augmented with the variables $\mathbf{f}_m$ to form the model

$$p(\mathbf{y}, \mathbf{f}, \mathbf{f}_m) = p(\mathbf{y}|\mathbf{f}) p(\mathbf{f}|\mathbf{f}_m) p(\mathbf{f}_m), \tag{9}$$

where the conditional GP prior is given by $p(\mathbf{f}|\mathbf{f}_m) = N(\mathbf{f}|K_{nm}K_{mm}^{-1}\mathbf{f}_m, K_{nn} - K_{nm}K_{mm}^{-1}K_{mn})$. As a consequence of the marginalization property of the Gaussian process according to which $p(\mathbf{f}) = \int p(\mathbf{f}, \mathbf{f}_m) d\mathbf{f}_m$, the two models are equivalent in terms of doing exact inference, i.e. computing the posterior $p(\mathbf{f}|\mathbf{y})$ and the marginal likelihood $p(\mathbf{y})$. For instance, the log marginal likelihood in eq. (4) can be equivalently written as

$$\log p(\mathbf{y}) = \log \int p(\mathbf{y}|\mathbf{f}) p(\mathbf{f}|\mathbf{f}_m) p(\mathbf{f}_m) d\mathbf{f} d\mathbf{f}_m. \tag{10}$$

However, the augmented probability model is more flexible in terms of doing approximate inference since it contains a set of parameters, that is the inducing inputs X_m, which are somehow arbitrary. More precisely, these parameters do not affect the exact GP model, $p(\mathbf{y}, \mathbf{f})$, because $p(\mathbf{f})$ is not changing by varying the values of X_m despite the fact that $p(\mathbf{f}|\mathbf{f}_m)$ and $p(\mathbf{f}_m)$ do change. Therefore, X_m are not model parameters (as $(\sigma^2, \boldsymbol{\theta})$ are) and by applying approximate inference in the augmented probability model we can turn them into variational parameters. This is what we do next.

We want to approximate the true posterior distribution $p(\mathbf{f}, \mathbf{f}_m|\mathbf{y}) = p(\mathbf{f}|\mathbf{f}_m, \mathbf{y}) p(\mathbf{f}_m|\mathbf{y})$ by introducing a variational distribution $q(\mathbf{f}, \mathbf{f}_m)$ and minimizing the KL divergence:

$$\mathrm{KL}(q(\mathbf{f}, \mathbf{f}_m)||p(\mathbf{f}, \mathbf{f}_m|\mathbf{y})) = \int q(\mathbf{f}, \mathbf{f}_m) \log \frac{q(\mathbf{f}, \mathbf{f}_m)}{p(\mathbf{f}, \mathbf{f}_m|\mathbf{y})} d\mathbf{f} d\mathbf{f}_m. \tag{11}$$

This is based on the standard variational approach widely used in machine learning (Jordan et al., 1999). To specify the form of the variational distribution $q(\mathbf{f}, \mathbf{f}_m)$, we follow the arguments exposed earlier in section 3. Particularly, for an optimal setting of the inducing variables, the exact posterior $p(\mathbf{f}, \mathbf{f}_m|\mathbf{y})$ factorizes as $p(\mathbf{f}, \mathbf{f}_m|\mathbf{y}) = p(\mathbf{f}|\mathbf{f}_m) p(\mathbf{f}_m|\mathbf{y})$. This tells us that the variational distribution must satisfy the same factorization as well, in order for the minimization of the KL divergence to search for these optimal inducing variables. Thus,

$$q(\mathbf{f}, \mathbf{f}_m) = p(\mathbf{f}|\mathbf{f}_m) \phi(\mathbf{f}_m), \tag{12}$$

where $\phi(\mathbf{f}_m)$ is an unconstrained variational distribution over $\mathbf{f}_m$ and $p(\mathbf{f}|\mathbf{f}_m)$ is the conditional GP prior. Notice also that the form of this distribution directly follows from eq. (7). To determine the variational quantities (X_m, ϕ), we minimize the KL divergence in eq. (11), which is equivalently expressed as the maximization of the following variational lower bound on the true log marginal likelihood:

$$\begin{aligned}\log p(\mathbf{y}) \geq F_V(X_m, \phi) &= \int p(\mathbf{f}|\mathbf{f}_m)\phi(\mathbf{f}_m) \log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{f}_m)p(\mathbf{f}_m)}{p(\mathbf{f}|\mathbf{f}_m)\phi(\mathbf{f}_m)} d\mathbf{f} d\mathbf{f}_m, \\ &= \int \phi(\mathbf{f}_m) \left\{ \int p(\mathbf{f}|\mathbf{f}_m) \log p(\mathbf{y}|\mathbf{f}) d\mathbf{f} + \log \frac{p(\mathbf{f}_m)}{\phi(\mathbf{f}_m)} \right\} d\mathbf{f}_m, \qquad (13)\end{aligned}$$

where the term $p(\mathbf{f}|\mathbf{f}_m)$, in the first line inside the log, cancels out[4]. We can firstly maximize the bound by analytically solving for the optimal choice of the variational distribution ϕ. To do this, we firstly compute the integral

$$\begin{aligned}\log G(\mathbf{f}_m, \mathbf{y}) &= \int p(\mathbf{f}|\mathbf{f}_m) \log p(\mathbf{y}|\mathbf{f}) d\mathbf{f} \\ &= \log \left[N(\mathbf{y}|\boldsymbol{\alpha}, \sigma^2 I) \right] - \frac{1}{2\sigma^2} Tr(K_{nn} - Q_{nn}). \qquad (14)\end{aligned}$$

Here, $\boldsymbol{\alpha} = E[\mathbf{f}|\mathbf{f}_m] = K_{nm}K_{mm}^{-1}\mathbf{f}_m$ and $Q_{nn} = K_{nm}K_{mm}^{-1}K_{mn}$. Notice that $\boldsymbol{\alpha}$ and $K_{nn} - Q_{nn}$ are the mean vector and covariance matrix, respectively, of the conditional GP prior $p(\mathbf{f}|\mathbf{f}_m)$. Eq. (13) is written as

$$F_V(X_m, \phi) = \int \phi(\mathbf{f}_m) \log \frac{G(\mathbf{f}_m, \mathbf{y})p(\mathbf{f}_m)}{\phi(\mathbf{f}_m)} d\mathbf{f}_m. \qquad (15)$$

We can now maximize the bound with respect to the distribution ϕ, without computing the optimal distribution, called ϕ^*, itself. This is done by reversing the Jensen's inequality, i.e. moving the log outside of the integral, which gives

$$F_V(X_m) = \log \left[N(\mathbf{y}|\mathbf{0}, \sigma^2 I + Q_{nn}) \right] - \frac{1}{2\sigma^2} Tr(K_{nn} - Q_{nn}), \qquad (16)$$

where $Q_{nn} = K_{nm}K_{mm}^{-1}K_{mn}$. More details of the derivation of this bound are given in the appendix A. The novelty of the above objective function is that it contains a regularization trace term: $-\frac{1}{2\sigma^2}Tr(K_{nn} - Q_{nn})$. This clearly differentiates F_V from all marginal likelihoods, described by eq. (5), that were previously applied to sparse GP regression. We will analyze the trace term shortly.

The quantity in eq. (16) is computed in $O(nm^2)$ time and is a lower bound of the true log marginal likelihood for any value of the inducing inputs X_m. Further maximization of the bound can be achieved by optimizing over X_m and optionally over the number of these variables. Notice that the inducing inputs determine the flexibility of the variational distribution $q(\mathbf{f}, \mathbf{f}_m) = p(\mathbf{f}|\mathbf{f}_m)\phi^*(\mathbf{f}_m)$ since by tuning X_m we adapt both $p(\mathbf{f}|\mathbf{f}_m)$ and the underlying optimal distribution ϕ^*. To compute this optimal ϕ^*, we differentiate eq. (15) with respect to $\phi(\mathbf{f}_m)$ without imposing any constraints on the functional form of $\phi(\mathbf{f}_m)$ apart from being a distribution. This gives (see appendix A):

$$\phi^*(\mathbf{f}_m) = N(\mathbf{f}_m|\boldsymbol{\mu}, A), \qquad (17)$$

where $\boldsymbol{\mu} = \sigma^{-2}K_{mm}\Sigma K_{mn}\mathbf{y}$, $A = K_{mm}\Sigma K_{mm}$ and $\Sigma = (K_{mm} + \sigma^{-2}K_{mn}K_{nm})^{-1}$. This now fully specifies our variational GP and we can substitute $(\boldsymbol{\mu}, A)$ in eq. (8) in order to make predictions in unseen input points. Clearly, the predictive distribution is exactly the one used by the projected process

[4] This cancellation is actually not needed as we can arrive faster at eq. (13) by writing $\log p(\mathbf{y}) \geq \int \phi(\mathbf{f}_m) \log \frac{p(\mathbf{f}_m) \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{f}_m)d\mathbf{f}}{\phi(\mathbf{f}_m)} d\mathbf{f}_m$ and then apply once again Jensen's inequality to lower bound the term $\log \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{f}_m)d\mathbf{f}$ with respect to $p(\mathbf{f}|\mathbf{f}_m)$.

approximation (PP) that has been previously proposed in (Csato and Opper, 2002; Seeger et al., 2003). Thus, as far as the predictive distribution is concerned the above method is equivalent to PP.

However, the variational method is very different to PP and SPGP as far as the selection of the inducing inputs and the kernel hyperparameters is concerned. This is because of the extra regularization term that appears in the bound in eq. (16) and does not appear in the approximate log marginal likelihoods used in PP (Seeger et al., 2003) and SPGP (Snelson and Ghahramani, 2006). As discussed in section 2, for the latter objective functions, the role of X_m is to form a set of extra kernel hyperparameters. In contrast, for the lower bound, the inputs X_m become variational parameters due to the KL divergence that is minimized.

To look into the functional form of the bound, note that F_V is the sum of the PP log likelihood and the regularization trace term $-\frac{1}{2}\sigma^{-2}Tr(K_{nn} - Q_{nn})$. Thus, F_V attempts to maximize the PP log likelihood and simultaneously minimize the trace $Tr(K_{nn} - Q_{nn})$. This trace represents the total variance of the conditional prior $p(\mathbf{f}|\mathbf{f}_m)$ which also corresponds to the squared error of predicting the training latent values $\mathbf{f}$ from the inducing variables $\mathbf{f}_m$: $\int p(\mathbf{f}, \mathbf{f}_m)||K_{nm}K_{mm}^{-1}\mathbf{f}_m - \mathbf{f}||^2 d\mathbf{f}d\mathbf{f}_m$. When the trace term is zero, the Nyström approximation is exact, i.e. $K_{nn} = K_{nm}K_{mm}^{-1}K_{mn}$, which means that the variational distribution $q(\mathbf{f}, \mathbf{f}_m)$ matches exactly the true posterior distribution. Note that the trace $Tr(K_{nn} - Q_{nn})$ itself has been used as a criterion for selecting the inducing points from the training data in (Smola and Schölkopf, 2000) and is similar to the criterion used in (Lawrence et al., 2002).

When we maximize the variational lower bound, the hyperparameters $(\sigma^2, \boldsymbol{\theta})$ are regularized. It is easy to see how this is achieved for the noise variance σ^2. At a local maxima, σ^2 satisfies:

$$\sigma^2 = \frac{1}{n}\int \phi^*(\mathbf{f}_m)||\mathbf{y} - \boldsymbol{\alpha}||^2 d\mathbf{f}_m + \frac{1}{n}Tr(K_{nn} - Q_{nn}), \tag{18}$$

where $||\mathbf{z}||$ denotes the Euclidean norm and $\boldsymbol{\alpha} = E[\mathbf{f}|\mathbf{f}_m] = K_{nm}K_{mm}^{-1}\mathbf{f}_m$. This decomposition reveals that the obtained σ^2 will be equal to the estimated “actual” noise plus a “correction” term that is the average squared error associated with the prediction of the training latent values $\mathbf{f}$ from the inducing variables $\mathbf{f}_m$. Thus, the variational lower bound naturally prefers to set σ^2 larger than the “actual” noise in a way that is proportional to the inaccuracy of the approximation.

So far we assumed that the inducing variables correspond to pseudo-inputs which are are selected by applying gradient-based optimization. However, this can be difficult in high dimensional input spaces as the number of variables that need to be optimized becomes very large. Further, the kernel function might not be differentiable with respect to the inputs. see e.g. kernels defined in strings. In such cases we can still apply the variational method by selecting the inducing inputs from the training inputs. An important property of this discrete optimization scheme is that the variational lower bound monotonically increases when we greedily select inducing inputs and adapt the hyperparameters. Next we discuss this greedy selection method.

3.2 Greedy selection from the training inputs

Let $m \subset \{1, \ldots, n\}$ be the indices of a subset of data that are used as the inducing variables. The training points that are not part of the inducing set are indexed by $n - m$ and are called the remaining points, e.g. $\mathbf{f}_{n-m}$ denotes the remaining latent function values. The variational method is applied similarly to the pseudo-inputs case. Assuming the variational distribution $q(\mathbf{f}) = p(\mathbf{f}_{n-m}|\mathbf{f}_m)\phi(\mathbf{f}_m)$, we can express a variational bound that has the same form as the bound in eq. (16) with the only difference that the covariance matrix in the trace term is now given by $\text{Cov}(\mathbf{f}_{n-m}|\mathbf{f}_m) = K_{(n-m)(n-m)} - K_{(n-m)m}K_{mm}^{-1}K_{m(n-m)}$.

The selection of inducing variables among the training data requires a prohibitive combinatorial search. A suboptimal solution can be based on a greedy selection scheme where we start with an empty inducing set $m = \emptyset$ and a remaining set $n - m = \{1, \dots, n\}$. At each iteration, we add a training point $j \in J \subset n - m$, where J is a randomly chosen working set, into the inducing set that maximizes the selection criterion Δ_j.

It is important to interleave the greedy selection process with the adaption of the hyperparameters $(\sigma^2, \boldsymbol{\theta})$. This can be viewed as an EM-like algorithm; at the E step we add one point into the inducing set and at the M step we update the hyperparameters. To obtain a reliable convergence, the approximate marginal likelihood must monotonically increase at each E or M step. The PP and SPGP log likelihoods do not satisfy such a requirement because they can also decrease as we add points into the inducing set. In contrast, the bound F_V is guaranteed to monotonically increase since now the EM-like algorithm is a variational EM. To clarify this, we state the following proposition.

Proposition 1. Let $(X_m, \mathbf{f}_m)$ be the current set of inducing points and m the corresponding set of indices. Any point $i \in n - m$ added into the inducing set can never decrease the lower bound.

Proof: Before the new point $(f_i, \mathbf{x}_i)$ is added, the variational distribution is $p(\mathbf{f}_{n-m}|\mathbf{f}_m)\phi^*(\mathbf{f}_m) = p(\mathbf{f}_{n-(m\cup i)}|f_i, \mathbf{f}_m)p(f_i|\mathbf{f}_m)\phi^*(\mathbf{f}_m)$. When we add the new point, the term $p(f_i|\mathbf{f}_m)\phi^*(\mathbf{f}_m)$ is replaced by the optimal $\phi^*(f_i, \mathbf{f}_m)$ distribution. This can either increase the lower bound or leave it invariant. A more detailed proof is given in the appendix B.

A consequence of the above proposition is that the greedy selection process monotonically increases the lower bound and this holds for any possible criterion Δ. An obvious choice is to use F_V as the criterion, which can be evaluated in $O(nm)$ time for any candidate point in the working set J. Such a selection process maximizes the decrease in the divergence $\text{KL}(q(\mathbf{f})||p(\mathbf{f}|\mathbf{y}))$.

Finally, we should point out that the selection of the inducing variables from the training data should not necessarily be performed based on the greedy selection strategy discussed above. One could consider different strategies such as adding groups of training points simultaneously into the active set or swapping points between the active and the remaining sets. Any strategy for reselecting the active set simply updates the variational distribution $q(\mathbf{f})$, and as long as this is done so that the bound increases, the whole procedure leads to a valid variational EM algorithm.

4 Comparison of the objective functions

In this section we compare the lower bound F_V, the PP and the SPGP log likelihood in some toy problems. All these functions are continuous with respect to $(X_m, \sigma^2, \boldsymbol{\theta})$ and can be maximized using gradient-based optimization.

Our working example will be the one-dimensional dataset[5] considered in Snelson and Ghahramani (2006) that consists of 200 training points; see Figure 1. We train a sparse GP model using the squared exponential kernel defined by $\sigma_f^2 \exp(-\frac{1}{2\ell^2}||x_i - x_j||^2)$. Since the dataset is small and the full GP model is tractable, we compare the sparse approximations with the exact GP prediction. The plots in the first row of Figure 1 show the predictive distributions for the three methods assuming 15 inducing inputs. The left plot displays the mean prediction with two-standard error bars (shown as blue solid lines) obtained by the maximization of F_V. The prediction of the full GP model is displayed using dashed red lines. The middle

[5] obtained from `www.gatsby.ucl.ac.uk/~snelson/`.

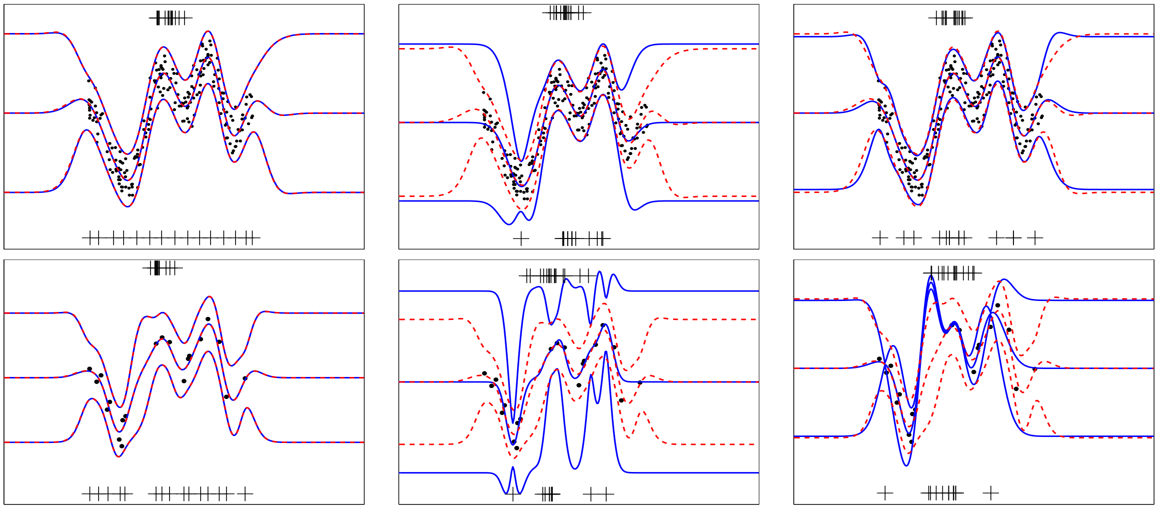


Figure 1: The first row corresponds to 200 training points and the second row to 20 training points. The first column shows the prediction (blue solid lines) obtained by maximizing F_V over the 15 pseudo-inputs and the hyperparameters. The full GP prediction is shown with red dashed lines. Initial locations of the pseudo-inputs are shown on the top as crosses, while final positions are given on the bottom as crosses. The second column shows the predictive distributions found by PP and similarly the third column for SPGP.

plot shows the corresponding solution found by PP and the right plot the solution found by SPGP. The prediction obtained by the variational method almost exactly reproduces the full GP prediction. The final value of the variational lower bound was -55.5708, while the value of the maximized true log marginal likelihood was -55.5647. Further, the estimated hyperparameters found by F_V match the hyperparameters found by maximizing the true log marginal likelihood. In contrast, training the sparse model using the PP log likelihood gives a poor approximation. The SPGP method gave a much more satisfactory answer than PP although not as good as the variational method.

To consider a more challenging problem, we decrease the number of the original 200 training examples by maintaining only 20 of them[6]. We repeat the experiment above using exactly the same setup. The second row of Figure 1, displays the predictive distributions of the three methods. The prediction of the variational method is identical to the full GP prediction and the hyperparameters match those obtained by full GP training. On the other hand, the PP log likelihood leads to a significant overfitting of the training data since the mean curve interpolates the training points and the error bars are very noisy. SPGP provides a solution that significantly disagrees with the full GP prediction both in terms of the mean prediction and the errors bars. Notice that the width of the error bars found by SPGP varies a lot in different input regions. This nonstationarity is achieved by setting σ^2 very close to zero and modelling the actual noise by the heteroscedastic diagonal matrix diag$[K_{nn} - K_{nm}K_{mm}^{-1}K_{mn}]$. The fact that this diagonal matrix (the sum of its elements is the trace $Tr(K_{nn} - Q_{nn})$) is large indicates that the full GP model is not well approximated.

The reason PP and SPGP do not recover the full GP model when we optimize over $(X_m, \sigma^2, \boldsymbol{\theta})$ is not the local maxima. To clarify this point, we repeated the experiments by initializing the PP and SPGP log likelihoods to optimal inducing inputs and hyperparameters values where the later are obtained by full GP training. The predictions found are similar to those shown in Figure 1. A way to ensure that

[6] The points were chosen from the original set according to the MATLAB command: X = X(1:10:end).

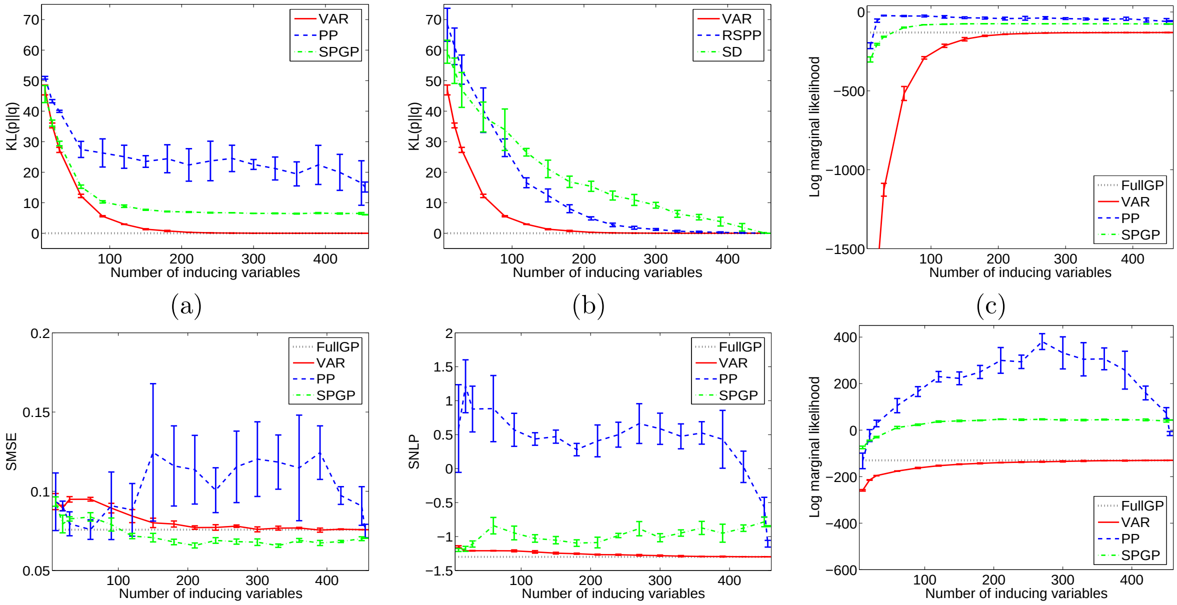
the full GP model will be recovered as we increase the number of inducing inputs is to select them from the training inputs. This, however, turns the continuous optimization problem into a discrete one and moreover PP and SPGP face the non-smooth convergence problem.

Regarding F_V, it is clear from section 3 that by maximizing over X_m we approach the full GP model in the sense of $\mathrm{KL}(q(\mathbf{f}, \mathbf{f}_m)|p(\mathbf{f}, \mathbf{f}_m|\mathbf{y}))$. Something less clear is that F_V efficiently regularizes the hyperparameters $(\sigma^2, \boldsymbol{\theta})$ so as overfitting is avoided. This is achieved by the regularization trace term: $-\frac{1}{2}\sigma^{-2}Tr(K_{nn} - Q_{nn})$. When this trace term is large because there are not sufficiently many inducing variables, this term favours kernel parameters that give a smoother function. Also, when the trace term is large the decomposition in eq. (18) implies that σ^2 must increase as well. These properties are useful for avoiding overfitting and also imply that the prediction obtained by F_V will tend to be smoother than the prediction of the full GP model. In contrast, the PP and SPGP log likelihoods can find more flexible solutions than the full GP prediction which indicates that they are prone to overfitting.

5 Experiments

In this section we compare the variational lower bound (VAR), the projected process approximate log likelihood (PP) and the sparse pseudo-inputs GP (SPGP) log likelihood in four real datasets. As a baseline technique, we use the subset of data (SD) method. For all sparse GP methods we jointly maximize the alternative objective functions w.r.t. hyperparameters $(\boldsymbol{\theta}, \sigma^2)$ and the inducing inputs X_m using the conjugate gradients algorithm. X_m is initialized to a randomly chosen subset of training inputs. In each run all methods are initialized to the same inducing inputs and hyperparameters. The performance criteria we use are the standardized mean squared error (SMSE), given by $\frac{1}{T}\frac{||\mathbf{y}_* - \mathbf{f}_*||^2}{var(\mathbf{y}_*)}$, and the standardized negative log probability density (SNLP) as defined in (Rasmussen and Williams, 2006). Smaller values for both error measures imply better performance. In all the experiments we use the squared-exponential kernel with varied length-scale.

Firstly, we consider the Boston-housing dataset, which consists of 455 training examples and 51 test examples. Since the dataset is small, full GP training is tractable. In the first experiment, we fix the parameters $(\boldsymbol{\theta}, \sigma^2)$ to values obtained by training the full GP model. Thus we can investigate the difference of the methods solely on how the inducing inputs are selected. We rigorously compare the methods by calculating the moments-matching divergence $\mathrm{KL}(p(\mathbf{f}_*|\mathbf{y})||q(\mathbf{f}_*))$ between the true test posterior $p(\mathbf{f}_*|\mathbf{y})$ and each of the approximate test posteriors. For the SPGP method the approximate test posterior distribution is computed by using the exact test conditional $p(\mathbf{f}_*|\mathbf{f}_m)$. Figure 2(a) show the KL divergence as the number of inducing points increases. Means and one-standard error bars were obtained by repeating the experiment 10 times. Note that only the VAR method is able to match the full GP model; for around 200 points we closely match the full GP prediction. Interestingly, when the inducing inputs are initialized to all training inputs, i.e. $X_m = X$, PP and SPGP still give a different solution from the full GP model despite the fact that the hyperparameters are kept fixed to the values of the full GP model. The reason this is happening is that they are not lower bounds to the true log marginal likelihood and as shown in Figure 2(c) they become upper bounds. To show that the effective selection of the inducing inputs achieved by VAR is not a coincidence, we compare it with the case where the inputs are kept fixed to their initial randomly selected training inputs. Figure 2(b) displays the evolution of the KL divergence for the VAR, the random selection plus PP (RSPP) and the SD method. Note that the only difference between VAR and RSPP is that VAR optimizes the lower bound over the initial values of



(a) (b) (c)

(d) (e) (f)

Figure 2: (a) show the KL divergence as the number of inducing variables increases for the VAR the PP and SPGP methods. Similarly (b) show the divergence for the VAR, RSPP and SD methods. (c) displays the approximate log marginal likelihoods; the true log marginal likelihood value is displayed by using the dotted horizontal line. (d) and (e) show the SMSE and SNLP errors (obtained by joint learning hyperparameters and inducing-inputs) against the number of inducing variables. (f) shows the corresponding log marginal likelihoods.

the inducing inputs, while RSPP just keep them fixed. Clearly RSPP significantly improves over the SD prediction, and VAR significantly improves over RSPP.

In a second experiment, we jointly learn inducing variables and hyperparameters and compare the methods in terms of the SMSE and SNLP errors. The results are displayed in the second row of Figure 2. Note that the PP and SPGP methods achieve a much higher log likelihood value (Figure 2(f)) than the true log marginal likelihood. However, the error measures clearly indicate that the PP log likelihood significantly overfits the data. SPGP gives better SMSE error than the full GP model but it overfits w.r.t. the SNLP error. The variational method matches the full GP model.

We now consider three large datasets: the KIN40K dataset, the SARCOS and the ABALONE datasets[7] that have been widely used before. Note that the ABALONE dataset is small enough so as we will be able to train the full GP model. The inputs were normalized to have zero mean and unit variance on the training set and the outputs were centered so as to have zero mean on the training set. For the KIN40K and the SARCOS datasets, the SD method was obtained in a subset of 2000 training points. We vary the size of the inducing variables in powers of two from 16 to 1024. For the SARCOS dataset, the experiment for 1024 was not performed since is was unrealistically expensive. All the objective functions were jointly maximized over inducing inputs and hyperparameters. The experiment was repeated 5 times. Figure 3 shows the results.

From the plots in Figure 3, we can conclude the following. The PP log likelihood is significantly prone to overfitting as the SNLP errors clearly indicate. However, note that in the KIN40K and SARCOS

[7] KIN40K: 10000 training, 30000 test, 8 attributes, ida.first.fraunhofer.de/ anton/data.html.
SARCOS: 44, 484 training, 4, 449 test, 21 attributes, www.gaussianprocess.org/gpml/data/.
ABALONE: 3, 133 training, 1, 044 test, 8 attributes, www.liaad.up.pt/ ltorgo/Regression/DataSets.html.

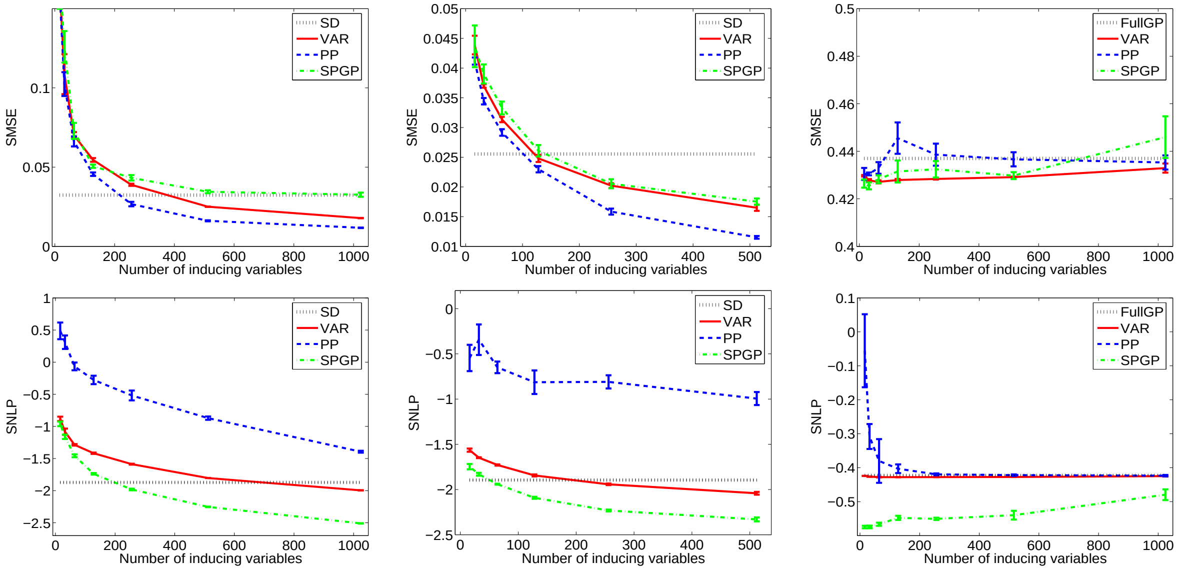


Figure 3: The first column displays the SMSE (top) and SNLP (bottom) errors for the KIN40K dataset with respect to the number of inducing points. The second column shows the corresponding plots for the SARCOS dataset and similarly the third column shows the results for the ABALONE dataset.

datasets, PP gave the best performance w.r.t. to SMSE error. This is probably because of the ability of PP to interpolate the training examples that can lead to good SMSE error when the actual observation noise is low. SPGP often has the worst performance in terms of the SMSE error and almost always the best performance in terms of the SNLP error. In the ABALONE dataset, SPGP had significantly better SNLP error than the full GP model. Since the SNLP error depends on the predictive variances, we believe that the good performance of SPGP is due to its heteroscedastic ability. For example, in the KIN40K dataset, SPGP makes σ^2 almost zero and thus the actual noise in the likelihood is modelled by the heteroscedastic covariance $\text{diag}[K_{nn} - K_{nm}K_{mm}^{-1}K_{mn}]$. The fact that the latter term is large may indicate that the full GP model is not well approximated. Finally the variational method has good performance. VAR never had the worst performance and it didn't exhibit overfitting. The examples in section 4, the Boston-housing and the ABALONE dataset indicate that the VAR method remains much closer to the full GP model than the other methods.

6 Adding "jitter" and more general inducing variables

In this section, and in order to address future research regarding sparse GP methods using inducing variables, we discuss the use of more general inducing variables than those being just function points. We do this by giving an example of how to slightly generalize the variational inference method so that to numerically stabilize the lower bound and the sparse GP prediction. This involves the usual trick of adding a "jitter" factor to the covariance matrix K_{mm} of the inducing points. We will show that this is actually rigorous and has a precise mathematical interpretation. Our discussion will have a more general outlook.

To start with, observe that when we maximize the exact log marginal likelihood in eq. (4) (assuming for now that we deal with small datasets) we need to invert the matrix $K_{nn} + \sigma^2 I$. This is usually carried out by computing the Cholesky decomposition of this matrix. While the kernel matrix K_{nn} can be (in computer precision) singular, i.e. it will have some zero eigenvalues, the matrix $K_{nn} + \sigma^2$ will often be strictly positive definite due to the addition of σ^2 in the diagonal of K_{nn}. Thus, σ^2 naturally acts as a "jitter" term that numerically stabilizes the computation of the Cholesky decomposition and subsequently the whole optimization of the log marginal likelihood. This, however, is not true for the sparse approximations to the exact marginal likelihood, neither for the PP and SPGP log marginal likelihoods in eq. (5) nor for the variational lower bound in eq. (16). The reason is that now we have to explicitly invert the kernel matrix K_{mm} evaluated at the inducing inputs without adding a positive number to the diagonal. To deal with that in a software implementation the common approach is to add a small "jitter" in the diagonal of K_{mm}. But what is the consequence of this regarding the mathematical properties of the objective function that we optimize? Next we show that for the variational method adding "jitter" leads to a generalized lower bound on the exact log marginal likelihood. In fact "jitter" will turn out to be another variational parameter and we can optimize over it.

The variational method in section 3, applies inference in an expanded probability space. Particularly, we started with the joint model $p(\mathbf{y}, \mathbf{f})$, we augmented it to form $p(\mathbf{y}, \mathbf{f}, \mathbf{f}_m)$ and then fitted an augmented variational distribution $q(\mathbf{f}, \mathbf{f}_m)$ to the exact posterior $p(\mathbf{f}, \mathbf{f}_m|\mathbf{y})$. The key property that made this a correct[8] inference procedure is the consistency condition between the original and the augmented model:

$$p(\mathbf{y}, \mathbf{f}) = \int p(\mathbf{y}, \mathbf{f}, \mathbf{f}_m) d\mathbf{f}_m,$$

which is true because $\mathbf{f}_m$ are function values drawn from the GP prior, i.e. it holds $p(\mathbf{f}) = \int p(\mathbf{f}, \mathbf{f}_m) d\mathbf{f}_m$. Its rather interesting to observe now that there is nothing in the above *augmentation* procedure that says that the newly introduced random variables must be function points of exactly the same type as $\mathbf{f}$. So let us augment with a different type of inducing variables, denoted by $\boldsymbol{\lambda}$, that are not precisely function values drawn from the GP prior, but instead they are noisy versions of functions values:

$$\boldsymbol{\lambda} = \mathbf{f}_m + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim N(\mathbf{0}, vI_m), \tag{19}$$

where $\mathbf{f}_m$, as before, is a vector of m auxiliary GP function values evaluated at the pseudo-inputs X_m and v is a noise variance parameter. Note that when $v = \sigma^2$, then $\boldsymbol{\lambda}$ can be considered as pseudo-output data, however, this is rather restrictive and it is better to let v be a free parameter. The covariance of $\boldsymbol{\lambda}$ is $K_{\lambda\lambda} = K_{mm} + vI$ and the cross covariance matrix between $\mathbf{f}$ and $\boldsymbol{\lambda}$ is $K_{n\lambda} = K_{nm}$. Thus augmenting the Gaussian prior $p(\mathbf{f})$ with inducing variables $\boldsymbol{\lambda}$ we obtain $p(\mathbf{f}, \boldsymbol{\lambda}) = p(\mathbf{f}|\boldsymbol{\lambda})p(\boldsymbol{\lambda})$ which, of course, satisfies the consistency condition $p(\mathbf{f}) = \int p(\mathbf{f}, \boldsymbol{\lambda}) d\boldsymbol{\lambda}$. Therefore, exact inference in the model $p(\mathbf{y}, \mathbf{f})$ (e.g. computing the marginal likelihood $p(\mathbf{y})$) can be equivalently performed using the augmented model

$$p(\mathbf{y}, \mathbf{f}, \boldsymbol{\lambda}) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\boldsymbol{\lambda})p(\boldsymbol{\lambda}). \tag{20}$$

However, doing approximate inference in the augmented model can be more flexible because the "free" parameters (X_m, v) can be treated as variational parameters. Thus, following the method described in section 3, and assuming the variational distribution $q(\mathbf{f}, \boldsymbol{\lambda}) = p(\mathbf{f}|\boldsymbol{\lambda})\phi(\boldsymbol{\lambda})$, we compute a lower bound to

[8] In the sense that when the KL divergence becomes zero, then the variational distribution allows to compute exactly the true posterior $p(\mathbf{f}|\mathbf{y})$.

the exact log marginal likelihood:

$$F_V(X_m, v) = \log\left[N(\mathbf{y}|\mathbf{0}, \sigma^2 I + K_{nm}(K_{mm} + vI_m)^{-1}K_{mn})\right] - \frac{1}{2\sigma^2}Tr(K_{nn} - K_{nm}(K_{mm} + vI_m)^{-1}K_{mn}). \tag{21}$$

This only differs from the initial bound of eq. (16) in that the "jitter" term, that is v, has been added to the diagonal of K_{mm} which makes the above bound slightly more general. To summarize, we just showed that adding "jitter" in the diagonal of K_{mm} gives always a lower bound on $\log p(\mathbf{y})$. The sparse GP prediction in eq. (8) is also affected since now it is expressed through $q(\mathbf{z}|\mathbf{y}) = \int p(\mathbf{z}|\boldsymbol{\lambda})\phi(\lambda)d\boldsymbol{\lambda}$ (analogously to eq. (7)) where the role of approximate sufficient statistics is played by $\boldsymbol{\lambda}$. Notice that the "jitter" v is actually a variational parameters as X_m is, so we can optimize over both of them. A sensible optimization strategy is to initialize v to a large value. This will help to numerically stabilize the optimization at the first crucial iterations and allow also to escape from local maxima. After few iterations v will typically become close to zero and we can constrain it to be larger than a certain minimum "jitter" value. The fact that v will approach zero as we optimize is because the noisy-free latent function values $\mathbf{f}_m$ are more informative about $\mathbf{f}$ compared to their noisy counterparts $\mathbf{f}_m + \boldsymbol{\epsilon}$. More precisely, the accuracy of the variational distribution $q(\mathbf{f}, \boldsymbol{\lambda}) = p(\mathbf{f}|\boldsymbol{\lambda})\phi(\boldsymbol{\lambda})$ depends on how much informative (i.e. deterministic) is the conditional GP prior $p(\mathbf{f}|\boldsymbol{\lambda})$ and among all $\boldsymbol{\lambda}$s is eq. (19) the most deterministic conditional prior is obtained when $v = 0$.

We address now the issue of what random variables could be used as inducing variables. A general answer is that any linear functional involving the GP function $f(\mathbf{x})$ can be an inducing variable. An useful set of inducing variables, $\boldsymbol{\lambda}$, should allow to decrease significantly the uncertainty in the conditional prior $p(\mathbf{f}|\boldsymbol{\lambda})$, which is part of the variational distribution. So good $\boldsymbol{\lambda}$s are those that can predict $\mathbf{f}$ with high certainty. From practical point of view inducing variables must be such that the covariance matrix $K_{\lambda\lambda}$ and the cross covariance matrix $K_{n\lambda}$ are computed both in closed-form and in a reasonable time so that the complexity of maximizing the variational lower bound will remain of order $O(nm^2)$.

We shall now draw a more general picture regarding the variational sparse GP method and the methods based on modifying the GP model. Assume we define a set of inducing variables $\boldsymbol{\lambda}$ for which the $K_{n\lambda}$ and $K_{\lambda\lambda}$ are computed analytically. Then, we can augment the joint model similarly to eq. (20). This involves the introduction of some parameters $\boldsymbol{\theta}_\lambda$, for example, when we use the "jitter" inducing variables $\boldsymbol{\lambda} = \mathbf{f}_m + \boldsymbol{\epsilon}$, these parameters are $\boldsymbol{\theta}_\lambda = (X_m, v)$. The variational method has the property that automatically turns the augmentation parameters $\boldsymbol{\theta}_\lambda$ into variational parameters that are selected so that the KL divergence is minimized. In contrast, for the SPGP or PP method, or any other method based on the philosophy of modifying the GP prior or likelihood, the $\boldsymbol{\theta}_\lambda$ parameters will be part of the model hyperparameters that are tuned in order to fit the data and not to approximate the exact GP model.

7 Discussion

We proposed a variational framework for sparse GP regression that can reliably learn inducing inputs and hyperparameters by minimizing the KL divergence between the true posterior GP and an approximate one. This approach starts with the full GP model and applies standard variational inference without introducing any approximation to the likelihood or the GP prior. The variational distribution follows the factorization that holds for optimally selected inducing variables. This method can be more generally applicable. An interesting topic for the future is to apply this method to GP models that assume multiple latent functions and consider also models with non-Gaussian likelihood functions, such as classification.

Furthermore, for the standard GP regression model discussed in this paper, the variational inference method gave rise to the PP/DTC prediction (Csato and Opper, 2002; Seeger et al., 2003). However, the SPGP/FITC prediction is considered to be more accurate that the PP/DTC prediction (Snelson, 2007; Quiñonero-Candela and Rasmussen, 2005). This arises the question: can we reformulate FITC so that the selection of the inducing inputs and hyperparameters can be achieved by maximizing a variational lower bound. An attempt to define such a bound is given in the appendix C. This bound is derived by starting with a full GP model that does not correspond to the standard GP regression, but instead it is a more flexible model that explains heteroscedastic noise in the output data.

Acknowledgements

I am grateful to Neil Lawrence for his help. Also, I wish to thank Magnus Rattray, Mauricio Alvarez, David Luengo Garcia and Chris Williams for useful comments. This work is funded by EPSRC Grant No EP/F005687/1 ”Gaussian Processes for Systems Identification with Applications in Systems Biology”.

References

Csato, L. (2002). *Gaussian Processes – Iterative Sparse Approximations*. PhD thesis, Aston University,UK.

Csato, L. and Opper, M. (2002). Sparse online Gaussian processes. *Neural Computation*, 14:641–668.

Jordan, M., Ghahramani, Z., Jaakkola, T., and Saul, L. (1999). An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233.

Keerthi, S. and Chu, W. (2006). A Matching Pursuit approach to sparse Gaussian process regression. In Weiss, Y., Schölkopf, B., and Platt, J., editors, *Neural Information Processing Systems 18*. MIT Press.

Lawrence, N. D., Seeger, M., and Herbrich, R. (2002). Fast sparse Gaussian process methods: the informative vector machine. In *Neural Information Processing Systems, 13*. MIT Press.

O’Hagan, A. (1978). Curve fitting and optimal design for prediction. *Journal of the Royal Statistical Society, Series B*, 40(1):1–42.

Quiñonero-Candela, J. and Rasmussen, C. E. (2005). A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. MIT Press.

Schwaighofer, A. and Tresp, V. (2003). Transductive and inductive methods for approximate Gaussian process regression. In *Neural Information Processing Systems 15*. MIT Press.

Seeger, M. (2003). *Bayesian Gaussian Process Models: PAC-Bayesian Generalisation Error Bounds and Sparse Approximations*. PhD thesis, University of Edinburgh.

Seeger, M., Williams, C. K. I., and Lawrence, N. D. (2003). Fast forward selection to speed up sparse Gaussian process regression. In *Ninth International Workshop on Artificial Intelligence*. MIT Press.

Smola, A. J. and Bartlett, P. (2001). Sparse greedy Gaussian process regression. In *Neural Information Processing Systems, 13*. MIT Press.

Smola, A. J. and Schölkopf, B. (2000). Sparse greedy matrix approximations for machine learning. In *International Conference on Machine Learning*.

Snelson, E. (2007). *Flexible and efficient Gaussian process models for machine learning*. PhD thesis, Gatsby Computational Neuroscience Unit, University College London.

Snelson, E. and Ghahramani, Z. (2006). Sparse Gaussian process using pseudo-inputs. In *Neural Information Processing Systems, 13*. MIT Press.

Titsias, M. K. (2009). Variational learning of inducing variables in sparse gaussian processes. In *Twelfth International Conference on Artificial Intelligence and Statistics, JMLR: W and CP*, volume 5, pages 567–574.

Wahba, G. (1990). Spline Models for Observational Data. *Society for Industrial and Applied Mathematics*, 59.

Walder, C., Kim, I. K., and Schölkopf, B. (2008). Sparse multiscale gaussian process regression. In *ICML*, pages 1112–1119.

Williams, C. K. I. and Seeger, M. (2001). Using the Nyström method to speed up kernel machines. In *Neural Information Processing Systems 13*. MIT Press.

A Variational lower bound and optimal distribution ϕ^*

The true log marginal likelihood is written as follows:

$$\log p(\mathbf{y}) = \log \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{f}_m)p(\mathbf{f}_m)d\mathbf{f}d\mathbf{f}_m. \tag{22}$$

Introducing the variational distribution $q(\mathbf{f}, \mathbf{f}_m) = p(\mathbf{f}|\mathbf{f}_m)\phi(\mathbf{f}_m)$ and applying Jensen's inequality we obtain the lower bound:

$$\begin{aligned} F_V(X_m, \phi) &= \int p(\mathbf{f}|\mathbf{f}_m)\phi(\mathbf{f}_m) \log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{f}_m)p(\mathbf{f}_m)}{p(\mathbf{f}|\mathbf{f}_m)\phi(\mathbf{f}_m)} d\mathbf{f}d\mathbf{f}_m \\ &= \int p(\mathbf{f}|\mathbf{f}_m)\phi(\mathbf{f}_m) \log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}_m)}{\phi(\mathbf{f}_m)} d\mathbf{f}d\mathbf{f}_m. \end{aligned} \tag{23}$$

It can be written as

$$F_V(X_m, \phi) = \int \phi(\mathbf{f}_m) \left\{ \int p(\mathbf{f}|\mathbf{f}_m) \log p(\mathbf{y}|\mathbf{f})d\mathbf{f} + \log \frac{p(\mathbf{f}_m)}{\phi(\mathbf{f}_m)} \right\} d\mathbf{f}_m. \tag{24}$$

The integral involving $\mathbf{f}$ is computed as follows

$$\begin{aligned} \log G(\mathbf{f}_m, \mathbf{y}) &= \int p(\mathbf{f}|\mathbf{f}_m) \log p(\mathbf{y}|\mathbf{f})d\mathbf{f} \\ &= \int p(\mathbf{f}|\mathbf{f}_m) \left\{ -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2} Tr \left[\mathbf{y}\mathbf{y}^T - 2\mathbf{y}\mathbf{f}^T + \mathbf{f}\mathbf{f}^T \right] \right\} d\mathbf{f} \\ &= -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2} Tr \left[\mathbf{y}\mathbf{y}^T - 2\mathbf{y}\boldsymbol{\alpha}^T + \boldsymbol{\alpha}\boldsymbol{\alpha}^T + K_{nn} - Q_{nn} \right], \end{aligned} \tag{25}$$

where $\boldsymbol{\alpha} = E[\mathbf{f}|\mathbf{f}_m] = K_{nm}K_{mm}^{-1}\mathbf{f}_m$ and $Q_{nn} = K_{nm}K_{mm}^{-1}K_{mn}$. From the above we can recognize the expression

$$\log G(\mathbf{f}_m, \mathbf{y}) = \log \left[N(\mathbf{y}|\boldsymbol{\alpha}, \sigma^2 I) \right] - \frac{1}{2\sigma^2} Tr(K_{nn} - Q_{nn}). \tag{26}$$

$F_V(X_m, \phi)$ is now written as

$$F_V(X_m, \phi) = \int \phi(\mathbf{f}_m) \log \frac{N(\mathbf{y}|\boldsymbol{\alpha}, \sigma^2 I)p(\mathbf{f}_m)}{\phi(\mathbf{f}_m)} d\mathbf{f}_m - \frac{1}{2\sigma^2} Tr(K_{nn} - Q_{nn}). \tag{27}$$

We can now maximize this bound w.r.t. the distribution ϕ. The usual way of doing this is to take the derivative w.r.t. $\phi(\mathbf{f}_m)$, set to zero and obtain the optimal $\phi^*(\mathbf{f}_m) = \frac{N(\mathbf{y}|\boldsymbol{\alpha},\sigma^2 I)p(\mathbf{f}_m)}{\int N(\mathbf{y}|\boldsymbol{\alpha},\sigma^2 I)p(\mathbf{f}_m)d\mathbf{f}_m}$. Then by substituting $\phi^*(\mathbf{f}_m)$ back to eq. (27) we can compute the optimal bound w.r.t. to the distribution ϕ. However, since ϕ was not constrained to belong to any restricted family of distributions, a faster and by far simpler way to compute the optimal bound is by reversing the Jensen's inequality, i.e. moving the log outside of the integral in eq. (27). This gives

$$\begin{aligned} F_V(X_m) &= \log \int N(\mathbf{y}|\boldsymbol{\alpha},\sigma^2 I)p(\mathbf{f}_m)d\mathbf{f}_m - \frac{1}{2\sigma^2}Tr(K_{nn} - Q_{nn}) \\ &= \log\left[N(\mathbf{y}|\mathbf{0},\sigma^2 I + Q_{nn})\right] - \frac{1}{2\sigma^2}Tr(K_{nn} - Q_{nn}). \end{aligned} \tag{28}$$

The optimal distribution ϕ that gives rise to this bound is given by

$$\begin{aligned} \phi^*(\mathbf{f}_m) &\propto N(\mathbf{y}|\boldsymbol{\alpha},\sigma^2 I)p(\mathbf{f}_m) \\ &= \mathrm{c}\exp\left\{-\frac{1}{2}\mathbf{f}_m^T(K_{mm}^{-1} + \frac{1}{\sigma^2}K_{mm}^{-1}K_{mn}K_{nm}K_{mm}^{-1})\mathbf{f}_m + \frac{1}{\sigma^2}\mathbf{y}^T K_{nm}K_{mm}^{-1}\mathbf{f}_m\right\}, \end{aligned} \tag{29}$$

where c is a constant. Completing the quadratic form we recognize the Gaussian

$$\phi^*(\mathbf{f}_m) = N(\mathbf{f}_m|\sigma^{-2}K_{mm}\Sigma^{-1}K_{mn}\mathbf{y}, K_{mm}\Sigma^{-1}K_{mm}), \tag{30}$$

where $\Sigma = K_{mm} + \sigma^{-2}K_{mn}K_{nm}$.

B Detailed proof of Proposition 1

Let the inducing variables be a subset of the training examples indexed by $m \subset \{1, \dots, n\}$. We also use m to denote the number of these variables. The training points that are not part of the inducing set are indexed by $n - m$ and are called the remaining points, e.g. $\mathbf{f}_{n-m}$ denotes the remaining latent function values.

The variational distribution we use to bound the true log marginal likelihood is

$$q_m(\mathbf{f}) = p(\mathbf{f}_{n-m}|\mathbf{f}_m)\phi_m(\mathbf{f}_m), \tag{31}$$

where we introduced m as an index of q and ϕ to emphasize that this variational distribution assumes m inducing variables. The variational bound takes the form

$$\begin{aligned} \log p(\mathbf{y}) &\geq \int_{\mathbf{f}} p(\mathbf{f}_{n-m}|\mathbf{f}_m)\phi_m(\mathbf{f}_m)\log\frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}_{n-m}|\mathbf{f}_m)p(\mathbf{f}_m)}{p(\mathbf{f}_{n-m}|\mathbf{f}_m)\phi_m(\mathbf{f}_m)}d\mathbf{f} \\ &= \int_{\mathbf{f}} p(\mathbf{f}_{n-m}|\mathbf{f}_m)\phi(\mathbf{f}_m)\log\frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}_m)}{\phi(\mathbf{f}_m)}d\mathbf{f} \\ &= F_V(X_m, \phi). \end{aligned} \tag{32}$$

The maximum value of this bound that corresponds to the optimal $\phi^*(\mathbf{f}_m)$ is

$$F_V(X_m) = \log\left[N(\mathbf{y}|\mathbf{0},\sigma^2 I + K_{nm}K_{mm}^{-1}K_{mn})\right] - \frac{1}{2\sigma^2}Tr(K_{(n-m)(n-m)} - K_{(n-m)m}K_{mm}^{-1}K_{m(n-m)}). \tag{33}$$

Note that $F_V(X_m) \geq F_V(X_m, \phi_m)$ for any distribution $\phi_m(\mathbf{f}_m)$.

Proposition 1. Let $(X_m, \mathbf{f}_m)$ be the current set of inducing variables and m the corresponding set of indices. Any point $i \in n - m$ added into the inducing set can never decrease the lower bound in eq. (33).

Proof: For the added training input $\mathbf{x}_i$, we have to show that $F_V(X_m, \mathbf{x}_i) \geq F_V(X_m)$. Let $\mathbf{f}_{m+1} = (\mathbf{f}_m, f_i)$ and $X_{m+1} = (X_m, \mathbf{x}_i)$. The variational distribution $q_m(\mathbf{f})$ that corresponds to the bound $F_V(X_m)$ can be written in the form

$$\begin{aligned} q_m(\mathbf{f}) &= p(\mathbf{f}_{n-(m+1)}, f_i|\mathbf{f}_m)\phi^*_m(\mathbf{f}_m), \\ &= p(\mathbf{f}_{n-(m+1)}|f_i, \mathbf{f}_m)p(f_i|\mathbf{f}_m)\phi^*_m(\mathbf{f}_m), \end{aligned} \tag{34}$$

where $\phi^*_m(\mathbf{f}_m)$ is the optimal choice for ϕ_m. This quantity has the same form with the variational distribution $q_{m+1}(\mathbf{f})$ where $\widetilde{\phi}_{m+1}(\mathbf{f}_{m+1}) = p(f_i|\mathbf{f}_m)\phi^*(\mathbf{f}_m)$. Notice that this $\widetilde{\phi}_{m+1}$ may not be optimal. Thus, by construction $F_V(X_{m+1}, \widetilde{\phi}_{m+1}(\mathbf{f}_{m+1})) = F_V(X_m)$ with $F_V(X_{m+1}, \widetilde{\phi}_{m+1})$ computed by eq. (32). Computing the bound for the optimal $\phi^*_{m+1}(\mathbf{f}_{m+1})$ in order to maximize w.r.t that distribution we obtain

$$F_V(X_{m+1}) \geq F_V(X_{m+1}, \widetilde{\phi}_{m+1}(\mathbf{f}_{m+1})) = F_V(X_m).$$

This completes the proof. Note that strict inequality can hold when the optimal $\phi^*_{m+1}(\mathbf{f}_{m+1})$ differs than $p(f_i|\mathbf{f}_m)\phi^*_m(\mathbf{f}_m)$.

C Variational reformulation of the SPGP model

The input-constant noise GP model with likelihood $p(\mathbf{y}|\mathbf{f}) = N(\mathbf{y}|\mathbf{f}, \sigma^2)$ can be restrictive when the actual noise of the data varies across the input space. A sparse GP method that can model input-dependent noise was proposed by Snelson and Ghahramani (2006). Assume a set of pseudo-inputs X_m and inducing latent function values $\mathbf{f}_m$, where $\mathbf{f}_m$ is drawn from the GP prior. The covariance matrix of the conditional prior $p(\mathbf{f}|\mathbf{f}_m)$, i.e. $K_{nn} - K_{nm}K^{-1}_{mm}K_{mn}$, can have non-stationary properties although the kernel function of the GP prior is stationary. The diagonal of this conditional matrix can be used to parametrize an input-dependent variance. The likelihood takes the form

$$p(\mathbf{y}|\mathbf{f}) = N(\mathbf{y}|\mathbf{f}, \Lambda), \tag{35}$$

where $\Lambda = \sigma^2 + \text{diag}[K_{nn} - K_{nm}K^{-1}_{mm}K_{mn}]$. The likelihood parameters are $(\sigma^2, X_m, \boldsymbol{\theta})$ where $\boldsymbol{\theta}$ are the kernel parameters of the GP prior. Note that the parameters $\boldsymbol{\theta}$ are common for the prior and the likelihood.

The marginal likelihood can be written as

$$\begin{aligned} p(\mathbf{y}) &= \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f})d\mathbf{f} \\ &= \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{f}_m)p(\mathbf{f}_m)d\mathbf{f}d\mathbf{f}_m \end{aligned} \tag{36}$$

where we augmented the GP prior on the training latent function $\mathbf{f}$ with the latent function values $\mathbf{f}_m$ evaluated at the pseudo-inputs X_m. We assume the variational distribution

$$q(\mathbf{f}, \mathbf{f}_m) = p(\mathbf{f}|\mathbf{f}_m)\phi(\mathbf{f}_m). \tag{37}$$

Using this distribution we can bound the true log marginal likelihood and obtain the lower bound:

$$\begin{aligned} F_V(X_m, \phi) &= \int p(\mathbf{f}|\mathbf{f}_m)\phi(\mathbf{f}_m)\log\frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}_m)}{\phi(\mathbf{f}_m)}d\mathbf{f}d\mathbf{f}_m \\ &= \int \phi(\mathbf{f}_m)\left\{\int p(\mathbf{f}|\mathbf{f}_m)\log p(\mathbf{y}|\mathbf{f})d\mathbf{f} + \log\frac{p(\mathbf{f}_m)}{\phi(\mathbf{f}_m)}\right\}d\mathbf{f}_m. \end{aligned} \tag{38}$$

The integral involving $\mathbf{f}$ is computed as follows

$$\begin{aligned}\log G(\mathbf{f}_m, \mathbf{y}) &= \int p(\mathbf{f}|\mathbf{f}_m) \log p(\mathbf{y}|\mathbf{f}) d\mathbf{f} \\ &= \int p(\mathbf{f}|\mathbf{f}_m) \left\{ -\frac{n}{2}\log(2\pi) - \frac{1}{2}\log|\Lambda| - \frac{1}{2}Tr\left[\Lambda^{-1}(\mathbf{y}\mathbf{y}^T - 2\mathbf{y}\mathbf{f}^T + \mathbf{f}\mathbf{f}^T)\right]\right\} d\mathbf{f} \\ &= -\frac{n}{2}\log(2\pi) - \frac{1}{2}\log|\Lambda| - \frac{1}{2}Tr\left[\Lambda^{-1}(\mathbf{y}\mathbf{y}^T - 2\mathbf{y}\boldsymbol{\alpha}^T + \boldsymbol{\alpha}\boldsymbol{\alpha}^T + K_{nn} - Q_{nn}\right], \qquad (39)\end{aligned}$$

where $\boldsymbol{\alpha} = E[\mathbf{f}|\mathbf{f}_m] = K_{nm}K_{mm}^{-1}\mathbf{f}_m$ and $Q_{nn} = K_{nm}K_{mm}^{-1}K_{mn}$. The above quantity can also be written as

$$\log G(\mathbf{f}_m, \mathbf{y}) = \log\left[N(\mathbf{y}|\boldsymbol{\alpha}, \Lambda)\right] - \frac{1}{2}Tr[\Lambda^{-1}(K_{nn} - Q_{nn})]. \qquad (40)$$

$F_V(X_m, \phi)$ is now written as

$$F_V(X_m, \phi) = \int \phi(\mathbf{f}_m) \log \frac{N(\mathbf{y}|\boldsymbol{\alpha}, \Lambda)p(\mathbf{f}_m)}{\phi(\mathbf{f}_m)} d\mathbf{f}_m - \frac{1}{2}Tr[\Lambda^{-1}(K_{nn} - Q_{nn})]. \qquad (41)$$

We can maximize this bound by reversing the Jensen's inequality:

$$\begin{aligned}F_V(X_m) &= \log \int N(\mathbf{y}|\boldsymbol{\alpha}, \Lambda)p(\mathbf{f}_m)d\mathbf{f}_m - \frac{1}{2}Tr[\Lambda^{-1}(K_{nn} - Q_{nn})] \\ &= \log\left[N(\mathbf{y}|\mathbf{0}, \Lambda + Q_{nn})\right] - \frac{1}{2}Tr[\Lambda^{-1}(K_{nn} - Q_{nn})]. \qquad (42)\end{aligned}$$

To compute the optimal distribution ϕ^* we differentiate eq. (41) with respect to $\phi(\mathbf{f}_m)$ and set to zero. This gives

$$\begin{aligned}\phi^*(\mathbf{f}_m) &\propto N(\mathbf{y}|\boldsymbol{\alpha}, \Lambda)p(\mathbf{f}_m) \\ &= \mathrm{c}\exp\left\{-\frac{1}{2}\mathbf{f}_m^T(K_{mm}^{-1} + K_{mm}^{-1}K_{mn}\Lambda^{-1}K_{nm}K_{mm}^{-1})\mathbf{f}_m + \mathbf{y}^T\Lambda^{-1}K_{nm}K_{mm}^{-1}\mathbf{f}_m\right\}. \qquad (43)\end{aligned}$$

From the above expression we can recognize the Gaussian

$$\phi^*(\mathbf{f}_m) = N(\mathbf{f}_m|K_{mm}\Sigma^{-1}K_{mn}\Lambda^{-1}\mathbf{y}, K_{mm}\Sigma^{-1}K_{mm}) \qquad (44)$$

where $\Sigma = K_{mm} + K_{mn}\Lambda^{-1}K_{nm}$. This distribution is exactly the one used by the SPGP model of Snelson and Ghahramani (2006). Particularly, when we do predictions assuming the exact test conditional $p(\mathbf{f}_*|\mathbf{f}_m)$, the above framework gives rises to the FITC prediction. The difference of the variational formulation with the SPGP/FITC method is that the former starts with an input-dependent noise full GP model and derives a variational bound for this model. The first term of the lower bound is the log marginal likelihood used by the SPGP model and the second term involves the trace of $\Lambda^{-1}(K_{nn} - Q_{nn})$.