Extended and Unscented Kitchen Sinks

Edwin V. Bonilla E.BONILLA@UNSW.EDU.AU
The University of New South Wales

Daniel Steinberg DANIEL.STEINBERG@NICTA.COM.AU
NICTA

Alistair Reid ALISTAIR.REID@NICTA.COM.AU
NICTA

Abstract

We propose a scalable multiple-output generalization of unscented and extended Gaussian processes. These algorithms have been designed to handle general likelihood models by linearizing them using a Taylor series or the Unscented Transform in a variational inference framework. We build upon random feature approximations of Gaussian process covariance functions and show that, on small-scale single-task problems, our methods can attain similar performance as the original algorithms while having less computational cost. We also evaluate our methods at a larger scale on MNIST and on a seismic inversion which is inherently a multi-task problem.

1. Introduction

Gaussian process (GP) models can be used as nonparametric probabilistic approaches to standard machine learning settings such as regression and classification (Rasmussen & Williams, 2006), where the latent functions modeled by the GP are only important as a means to an end, that of providing greater flexibility than their parametric counterparts. In other application areas such as inversion problems, the latent functions are quantities of interest themselves, and they are passed through a domain-specific *forward model* in order to generate the observations.

Standard machine learning tasks and inversion problems present three key challenges when having Gaussian process (GP) priors. The first challenge is scalability, as GPs are notorious for their poor scalability as a function of the number of training points. The second challenge is multi-output and multi-task task learning, as required by problems such as multi-output regression, multi-class classification or inversions over a multi-layer geological structure (where each layer is a task). Finally, the third challenge is that of dealing with nonlinear non-Gaussian likelihoods, for example in classification, regression with non-Gaussian noise, and seismic inversion, as the posterior over the latent functions is analytically intractable.

In order to address the latter challenge, dealing with nonlinear non-Gaussian likelihoods, Steinberg & Bonilla (2014) have shown recently that it is possible to obtain good posterior estimates in GP models using approximations of the nonlinear likelihood via a Taylor series expansion or via the Unscented Transform (Julier & Uhlmann, 2004). They refer to their methods as the extended Gaussian process (EGP) and the unscented Gaussian process (UGP). One of the fundamental reasons why such linearizations are effective is because of their locality and adaptivity, as they are constructed around the current posterior estimate, which is iteratively updated within a variational inference procedure. While such methods are an effective way to tackle nonlinearities in the likelihood, their approach does not deal with the other two challenges mentioned above, namely multi-task learning and scalability. Indeed their method is specific to single-output problems and inherits the cubic scalability of standard GPs on the number of training points.

In this paper we propose a scalable multiple-output generalization of the method of Steinberg & Bonilla (2014). We deal with multiple-output problems by using affine transformations of the latent functions and achieve scalability by introducing random feature approximations of the covariance function of the Gaussian processes, in the style of Rahimi & Recht (2008). Inference of all parameters and hyperparameters is carried out using a variational inference framework, and so the kernel learning methods introduced by Yang et al. (2015) can be applied to our methods. Since Rahimi & Recht (2008) refer to their approach as Random

Proceedings of the 33^{rd} International Conference on Machine Learning, New York, NY, USA, 2016. JMLR: W&CP volume 48. Copyright 2016 by the author(s).

Kitchen Sinks, we will refer to our methods as extended and unscented kitchen sinks (EKS, UKS), when using the Taylor series approximation or the Unscented Transform approximation in the conditional likelihood respectively.

Our approach naturally avoids the cubic scalability of the original EGP and UGP methods (Steinberg & Bonilla, 2014) as a function of the number of training points. Our algorithms' complexity is dominated by the inverse of the feature covariance of size D, which has a time complexity of $\mathcal{O}(D^3)$, where typically $D \ll N$.

Our experiments on small-scale synthetic nonlinear inversion tasks and on a classification task on the USPS dataset show that random feature approximations to the EGP and the UGP can attain similar performance to the original methods. This applies even when using a small number of features, hence reducing the complexity of inference significantly. Furthermore, experiments at a larger scale on MNIST show that our algorithms are competitive with recently developed approaches for inference in GP models, while the application of the EGP and UGP to this task is simply infeasible. Finally, on a multi-task (joint) nonlinear seismic inversion problem we show that our algorithms can recover accurate representations of the underlying geological structure and rock properties (seismic velocities).

2. Gaussian Process Models

We are given N input data points $\{\mathbf{x}_n\} \in \mathbb{R}^d$ and their corresponding targets $\{\mathbf{y}_n\} \in \mathbb{R}^P$, which will we describe compactly with $\{\mathbf{X}, \mathbf{Y}\}$, where $\mathbf{X} \in \mathbb{R}^{N \times d}$ and $\mathbf{Y} \in \mathbb{R}^{N \times P}$. Our goal is to learn a probabilistic mapping from inputs to outputs, which can be achieved through Q latent functions $\{f_q\}$ and a given non-linear forward model $\mathbf{g} : \mathbb{R}^Q \to \mathbb{R}^P$. Additionally, we are interested in estimating the posterior over the latent functions given the observed data. While the former problem is the standard multi-task supervised learning setting, we refer to the latter as a probabilistic joint inversion problem, as we are given a forward mapping from latent functions to noiseless outputs but not the reverse.

A flexible modeling approach places independent zero-mean Gaussian process (GP) priors over the latent functions $\{f_q\}$ with covariance functions $k_q(\cdot, \cdot)$ and assumes i.i.d observations given these latent functions. When these function are realized at the training data, we obtain the following prior and likelihood models:

$$p(\mathbf{F}) = \prod_{q=1}^{Q} \mathcal{N}(\mathbf{f}_{\cdot q}; \mathbf{0}, \mathbf{K}_q) \tag{1}$$

$$p(\mathbf{Y}|\mathbf{F}) = \prod_{n=1}^{N} p(\mathbf{y}_n|\mathbf{g}(\mathbf{f}_{n\cdot})), \tag{2}$$

where $\mathbf{K}_q$ is the covariance matrix induced by evaluating the covariance function k_q at all input data $\mathbf{X}$; $\mathbf{f}_{\cdot q}$ are the values of latent function q at all training inputs; and $\mathbf{f}_{n\cdot}$ are the values of all latent functions at input $\mathbf{x}_n$.

Having a nonlinear function $\mathbf{g}(\mathbf{f}_{n\cdot})$ in the conditional likelihood terms gives us the flexibility to go beyond standard regression with linear Gaussian noise, even when each $p(\mathbf{y}_n|\mathbf{g}(\mathbf{f}_{n\cdot}))$ is a Gaussian. For example, we can address problems such as classification, where $\mathbf{g}$ is e.g. a softmax function, and nonlinear inversion problems such as seismic inversion, where $\mathbf{g}$ maps depths and seismic velocities of geological layers to sound reflexion times.

From a probabilistic inference perspective, solving the inversion problem (and the subsequent prediction problem), boils down to computing the posterior distribution $p(\mathbf{f}|\mathbf{X}, \mathbf{Y})$. Unfortunately, this posterior distribution is, in general, intractable due to the non-linearities in $\mathbf{g}$, so one must resort to approximations.

2.1. Variational Inference in Linearized GP Models

Steinberg & Bonilla (2014) have recently proposed a variational algorithm that addresses the above problem for single-output observations ($Q = 1$). Their algorithm relies upon the linearization of the forward model around the posterior mean, allowing for an analytic approximation of the variational objective and enabling parameter learning within a simple but effective optimization procedure. To build such linearizations they use a Taylor series approximation and the unscented transform and refer to their methods as the Extended Gaussian Process (EGP) and the Unscented Gaussian Process (UGP). The main advantage of their algorithms is that their approximation is local and adaptive, as it is constructed around the posterior mean for a single data point n, and it gets updated at every iteration of the algorithm as a function of the variational parameters.

However, such algorithms have the fundamental problem of poor scalability, as they inherit the computational cost of traditional GP models, which is $\mathcal{O}(N^3)$. This problem is, of course, exacerbated when having multi-task learning settings or multiple outputs, which renders their approach impractical for large datasets.

3. Random Features Approximations

Our starting point to scale up linearized GP models builds upon the work of Rahimi & Recht (2008; 2009), who used Bochner's theorem regarding the relationship between a kernel and the Fourier transform of a non-negative measure. In particular, when such a non-negative measure exists, one obtains Wiener-Khintchine's theorem, which establishes the Fourier duality of the covariance function of a stationary process and its spectral density:

$$k(\boldsymbol{\tau}) = \int S(\mathbf{s}) e^{2\pi i \mathbf{s}^T \boldsymbol{\tau}} d\mathbf{s}, \tag{3}$$

$$S(\mathbf{s}) = \int k(\boldsymbol{\tau}) e^{-2\pi i \mathbf{s}^T \boldsymbol{\tau}} d\boldsymbol{\tau}. \tag{4}$$

Rahimi & Recht's main insight (2008) is that we can approximate the above kernel by explicitly constructing "suitable" random features and (Monte Carlo) averaging over samples from $S(\mathbf{s})$:

$$k(\mathbf{x} - \mathbf{x}') = k(\boldsymbol{\tau}) \approx \frac{1}{D} \sum_{i=1}^{D} \phi_i(\mathbf{x})\, \phi_i(\mathbf{x}')\,, \tag{5}$$

where $\phi(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}^D$ is a feature map and $\phi_i(\mathbf{x})$ corresponds to the ith component of that map. An example of a feature vector construction in the above approximation is:

$$[\phi_i(\mathbf{x})\,, \phi_{D+i}(\mathbf{x})] = \frac{1}{\sqrt{D}}[\cos(2\pi \mathbf{s}_i^T \mathbf{x}), \sin(2\pi \mathbf{s}_i^T \mathbf{x})], \\ \text{with } \mathbf{s}_i \sim \mathcal{N}\big(\mathbf{s}_i \big| \mathbf{0}, \sigma_\phi^2 \mathbf{I}_d\big)\,, \tag{6}$$

for $i = 1, \ldots, D$, which in fact is a mapping into a $2D$-dimensional feature space. Rahimi & Recht (2008) used the above feature map to approximate the commonly used (isotropic) squared exponential kernel, and showed that such an approximation converges in expectation to the true kernel. They refer to algorithms that use such randomized feature expansions as *random kitchen sinks* (RKS). If we use RKS bases such that $k(\mathbf{x}_i, \mathbf{x}_j) = \mathbb{E}[\phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j)]$, we can approximate GP models that involve nonlinear likelihoods with simple linear-in-the-parameters models. For our purposes, we are interested in using such random feature approximations within a variational inference framework in order to estimate the posterior distributions of models of the form given in Equations (1) and (2).

4. Multi-task GP Models

Our next step is to approximate our prior over latent functions in Equation (1) using the random feature approximation described above, and to specify our multi-output likelihood in Equation (2):

$$p(\mathbf{W}) = \prod_{q=1}^{Q} \mathcal{N}\big(\mathbf{w}_q \big| \mathbf{0}, \omega_q^2 \mathbf{I}_D\big)\,, \tag{7}$$

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{n=1}^{N} \mathcal{N}(\mathbf{y}_n | \mathbf{g}(\mathbf{W}\boldsymbol{\phi}_n)\,, \boldsymbol{\Sigma})\,, \tag{8}$$

where, $\boldsymbol{\phi}_n \overset{\text{def}}{=} \phi(\mathbf{x}_n)$ is the D-dimensional vector of features corresponding to datapoint n; $\mathbf{w}_q \in \mathbb{R}^D$; $\mathbf{W} \in \mathbb{R}^{Q \times D}$; ω_q^2 is the prior variance over the weights; and $\boldsymbol{\Sigma} = \text{diag}\big(\big[\sigma_1^2, \ldots, \sigma_P^2\big]\big)$ is the noise variance. Additionally, we note that we are effectively approximating our prior over latent functions as $\mathbf{f}_q = \boldsymbol{\Phi}\mathbf{w}_q$, with $\boldsymbol{\Phi} \overset{\text{def}}{=} \phi(\mathbf{X})$ being the $N \times D$ matrix of features evaluated at all the training data.

Having RKS-based approximations allows us to circumvent the inherent scalability problem in GP models. However, we note that the likelihood model in Equation (8), still involves a non linear transformation of the corresponding latent functions, which yields, as before, intractable posteriors. In order to address this problem, we will build upon the work of Steinberg & Bonilla (2014), and develop a variational inference procedure that exploits linearization methods around the posterior mean.

4.1. Posterior Approximation

Let us now make the simplifying assumption that the posterior factorizes over latent functions and has the form,

$$\tilde{q}(\mathbf{W}) = \prod_{q=1}^{Q} \mathcal{N}(\mathbf{w}_q | \mathbf{m}_q, \mathbf{C}_q)\,. \tag{9}$$

We can use variational inference to learn this posterior approximation, and thereby allowing us to infer the posterior latent tasks,

$$\tilde{q}(\mathbf{F}) = \prod_{q=1}^{Q} \mathcal{N}\Big(\mathbf{f}_{\cdot q} \Big| \boldsymbol{\Phi}\mathbf{m}_q, \boldsymbol{\Phi}\mathbf{C}_q\boldsymbol{\Phi}^T\Big)\,. \tag{10}$$

4.2. Evidence Lower Bound

With the prior specified in Equation (7); the likelihood specified in Equation (8); and the approximate posterior defined in Equation (9), we are now ready to write down the variational lower bound that we aim to maximize in order to learn the parameters of our model. Defining $\tilde{q}_{\mathbf{W}} \overset{\text{def}}{=} \tilde{q}(\mathbf{W})$, the variational log-evidence lower bound is,

$$\mathcal{L} = \langle \log p(\mathbf{Y}|\mathbf{W}, \mathbf{X}) \rangle_{\tilde{q}_{\mathbf{W}}} - \text{KL}[\tilde{q}(\mathbf{W}) \,\|\, p(\mathbf{W})]\,. \tag{11}$$

Here we can straight-forwardly arrive at the KL term,

$$\text{KL}[\tilde{q}(\mathbf{W}) \,\|\, p(\mathbf{W})] = \frac{1}{2} \sum_{q=1}^{Q} \Bigg[\frac{1}{\omega_q^2} \text{tr}(\mathbf{C}_q) + \frac{1}{\omega_q^2} \mathbf{m}_q^\top \mathbf{m}_q \\ - \log |\mathbf{C}_q| + D \log \omega_q^2 - D \Bigg]. \tag{12}$$

The expected log-likelihood term is less straight forward,

$$\langle \log p(\mathbf{Y}|\mathbf{W}, \mathbf{X}) \rangle_{\tilde{q}_{\mathbf{W}}} = -\frac{N}{2} \left[\log 2\pi + \log |\boldsymbol{\Sigma}|\right] \\ -\frac{1}{2} \sum_{n=1}^{N} \Big\langle (\mathbf{y}_n - \mathbf{g}(\mathbf{W}\boldsymbol{\phi}_n))^\top \boldsymbol{\Sigma}^{\text{-}1} (\mathbf{y}_n - \mathbf{g}(\mathbf{W}\boldsymbol{\phi}_n)) \Big\rangle_{\tilde{q}_{\mathbf{W}}}\,, \tag{13}$$

since this expectation cannot be easily evaluated. We make another approximation,

$$\mathbf{g}(\mathbf{W}_n\boldsymbol{\phi}_n) \approx \mathbf{A}_n\mathbf{W}\boldsymbol{\phi}_n + \mathbf{b}_n, \tag{14}$$

where $\mathbf{A}_n \in \mathbb{R}^{P\times Q}$ is some linearization matrix that we will define later, and $\mathbf{b}_n \in \mathbb{R}^P$ is an intercept term that we will also define later. Now we can evaluate the expectation in (13) as approximately,

$$\langle \log p(\mathbf{Y}|\mathbf{W},\mathbf{X})\rangle_{\tilde{q}\mathbf{W}} \approx -\frac{N}{2}\left[P\log 2\pi + \log|\mathbf{\Sigma}|\right] - \frac{1}{2}\sum_{n=1}^{N}\left[\boldsymbol{\epsilon}_n^\top\mathbf{\Sigma}^{\text{-}1}\boldsymbol{\epsilon}_n + \sum_{q=1}^{Q}\boldsymbol{\phi}_n^\top\mathbf{C}_q\boldsymbol{\phi}_n\mathbf{a}_{nq}^\top\mathbf{\Sigma}^{\text{-}1}\mathbf{a}_{nq}\right]. \tag{15}$$

Where we have defined $\boldsymbol{\epsilon}_n \overset{\text{def}}{=} \mathbf{y}_n - (\mathbf{A}_n\mathbf{M}\boldsymbol{\phi}_n + \mathbf{b}_n)$; used $\mathbf{A}_n\mathbf{W} = \sum_q \mathbf{a}_{nq}\mathbf{w}_q^\top$ defining $\mathbf{a}_{nq} \in \mathbb{R}^P$ and $\mathbf{A}_n = [\mathbf{a}_{n1},\ldots,\mathbf{a}_{nQ}]$. Here we can see that this objective easily factorizes over the data, and so it should be straightforward to apply parallel or stochastic gradient descent algorithms to learn the posterior parameters.

4.3. Learning the Variational Parameters

As in Steinberg & Bonilla (2014), we can use Newton's method to learn the approximate posterior parameters for each task,

$$\mathbf{m}_q^{(k+1)} = \mathbf{m}_q^{(k)} - \alpha_k\left(\nabla_{\mathbf{m}_q}\nabla_{\mathbf{m}_q}\mathcal{L}\right)^{-1}\nabla_{\mathbf{m}_q}\mathcal{L}\,\Big|_{\mathbf{m}_q=\mathbf{m}_q^{(k)}}. \tag{16}$$

Here $\alpha_k \in (0,1]$ is a step length, and the gradients of the variational lower bound with respect to the posterior mean are:

$$\nabla_{\mathbf{m}_q}\mathcal{L} = \sum_{n=1}^{N}\boldsymbol{\phi}_n\mathbf{a}_{nq}^\top\mathbf{\Sigma}^{\text{-}1}\left(\mathbf{y}_n - \mathbf{a}_{nq}\mathbf{m}_q^\top\boldsymbol{\phi}_n - \mathbf{b}_n\right) - \frac{1}{\omega_q^2}\mathbf{m}_q. \tag{17}$$

Similarly, the Hessian of the variational objective is:

$$\nabla_{\mathbf{m}_q}\nabla_{\mathbf{m}_q}\mathcal{L} = -\frac{1}{\omega_q^2}\mathbf{I}_D - \sum_{n=1}^{N}\boldsymbol{\phi}_n\mathbf{a}_{nq}^\top\mathbf{\Sigma}^{\text{-}1}\mathbf{a}_{nq}\boldsymbol{\phi}_n^\top. \tag{18}$$

When (16) has converged to $\mathbf{m}_q^+$ we can calculate the approximate posterior covariance,

$$\mathbf{C}_q = \left[\frac{1}{\omega_q^2}\mathbf{I}_D + \sum_{n=1}^{N}\boldsymbol{\phi}_n\mathbf{a}_{nq}^\top\mathbf{\Sigma}^{\text{-}1}\mathbf{a}_{nq}\boldsymbol{\phi}_n^\top\right]^{\text{-}1}. \tag{19}$$

4.4. Hyperparameter Learning

Once we have found the optimum $\mathbf{M}$ and $\mathbf{C}_q$'s, we can optimise the linearized $\mathcal{L}$ (which becomes an approximation to the lower bound) with respect to the parameters $\mathbf{\Sigma}$, $\{\omega_q^2\}$ and $\boldsymbol{\theta}$, assuming the features are parameterized $\boldsymbol{\phi}_n = \phi(\mathbf{x}_n, \boldsymbol{\theta})$. Once we have found the optimum $\mathbf{\Sigma}$, $\{\omega_q^2\}$, $\boldsymbol{\theta}$, we then re-optimize for the posterior parameters in a generalized variational-EM like procedure.

When we have determined the optimum posterior parameters, the trace term in Equation (12) cancels out with the only term in Equation (15) involving $\mathbf{C}_q$, i.e. $QD - \sum_q \text{tr}(\mathbf{C}_q)/\omega_q^2 = \sum_n\sum_q \boldsymbol{\phi}_n^\top\mathbf{C}_q\boldsymbol{\phi}_n\mathbf{a}_{nq}^\top\mathbf{\Sigma}^{\text{-}1}\mathbf{a}_{nq}$, to give,

$$\mathcal{L} \approx -\frac{N}{2}\left[P\log 2\pi + \log|\mathbf{\Sigma}|\right] - \frac{1}{2}\sum_{n=1}^{N}\left[(\boldsymbol{\epsilon}_n)^\top\mathbf{\Sigma}^{\text{-}1}(\boldsymbol{\epsilon}_n)\right] - \frac{1}{2}\sum_{q=1}^{Q}\left[\frac{1}{\omega_q^2}\mathbf{m}_q^\top\mathbf{m}_q - \log|\mathbf{C}_q| + D\log\omega_q^2\right]. \tag{20}$$

Unfortunately, the optimum mean ($\mathbf{M}^+$) is an implicit function of both $\mathbf{\Sigma}$ and $\boldsymbol{\theta}$, and so in the case of the EGP, we would require second and higher derivatives of $\mathbf{g}$ in order to calculate partial derivatives of (20) with respect to these parameters. However, if we assume $\mathcal{O}(10)$ tasks, P, and a lightly parameterized feature function, $\phi(\cdot)$, then we can use numerical methods for local derivative-free optimization such as COBYLA (Powell, 1994; 1998) or BOBYQA (Powell, 2009).

5. Linearization Methods

So far we have assumed that we are given the linearization parameters $\{\mathbf{A}_n, \mathbf{b}_n\}$ that allow us to approximate locally the nonlinear forward model given in Equation (14). As $\mathbf{g}(\cdot)$ is a function of our latent functions approximation $\mathbf{W}\boldsymbol{\phi}_n$, it makes sense to linearize around our current posterior estimates of these latent functions, $\hat{\mathbf{f}}_{n\cdot} = \mathbf{M}\boldsymbol{\phi}_n$. In this section we describe two methods for linearizing the forward model around these posterior estimates. The first method uses a Taylor series approximation and the second method uses a statistical linearization (Geist & Pietquin, 2010) based on the unscented transform (Julier & Uhlmann, 2004). Because of the relation of our algorithms to the Extended Gaussian Process (EGP) and the Unscented Gaussian Process (UGP), we refer to the proposed inference methods as the Extended Kitchen Sinks (EKS) and the Unscented Kitchen Sinks (UKS), respectively.

5.1. Taylor Series Linearization

We can use a first order Taylor series to linearize $\mathbf{g}(\cdot)$ at each iteration of (16),

$$\mathbf{g}(\mathbf{W}\boldsymbol{\phi}_n) \approx \mathbf{g}(\mathbf{M}\boldsymbol{\phi}_n) + \mathbf{J}_n(\mathbf{W} - \mathbf{M})\boldsymbol{\phi}_n, \tag{21}$$

where

$$\mathbf{J}_n = \frac{\partial\mathbf{g}(\mathbf{f}_{n\cdot})}{\partial\mathbf{f}_{n\cdot}}\Big|_{\mathbf{f}_{n\cdot}=\mathbf{M}\boldsymbol{\phi}_n}. \tag{22}$$

Equating coefficients with (14) we have that

$$\mathbf{A}_n = \mathbf{J}_n \qquad \text{and} \qquad \mathbf{b}_n = \mathbf{g}(\mathbf{M}\boldsymbol{\phi}_n) - \mathbf{J}_n\mathbf{M}\boldsymbol{\phi}_n. \quad (23)$$

We will refer to the method that uses this Taylor series linearization within the variational framework described in section 4 as Extended Kitchen Sinks (EKS).

5.2. Statistical Linearization

In order to have a statistical approach to estimating the linearization parameters in Equation (14), we can use, for example, weighted least squares. The main question is what "training" data can we use to fit the linear model? Although we can sample from $\mathbf{F}$ using Equation (10) to generate these data, the unscented transform (UT; Julier & Uhlmann, 2004) provides a deterministic and more elegant solution.

The main point to notice here is that we are interested in linearizing $\mathbf{g}$ as a function of $\mathbf{f}_{n\cdot}$, where $\mathbf{f}_{n\cdot}$ is a Q-dimensional random variable corresponding to the Q latent function values at datapoint n. Interestingly, our choice of variational distribution in Equation (14) assumes that the joint posterior factorizes across latent functions, and so does the marginal:

$$\tilde{q}(\mathbf{f}_{n\cdot}) = \mathcal{N}(\mathbf{f}_{n\cdot}|\boldsymbol{\mu}_n, \mathbf{E}_n)\text{, with } \boldsymbol{\mu}_n = \mathbf{M}\boldsymbol{\phi}_n\text{, and} \quad (24)$$

$$[\mathbf{E}_n]_{qq\prime} = \boldsymbol{\phi}_n^\top \mathbf{C}_q \boldsymbol{\phi}_n \text{ for } q = q\prime \text{ and } 0 \text{ otherwise.} \quad (25)$$

This greatly simplifies the computation of the UT, which involves the definition of $2Q+1$ so-called sigma-points:

$$\mathcal{F}_{0,n} = \boldsymbol{\mu}_n \quad (26)$$

$$\mathcal{F}_{i,n} = \boldsymbol{\mu}_n + [\sqrt{(Q+\kappa)\mathbf{E}_n}]_{\cdot,i} \quad 1 \le i \le Q \quad (27)$$

$$\mathcal{F}_{i,n} = \boldsymbol{\mu}_n - [\sqrt{(Q+\kappa)\mathbf{E}_n}]_{\cdot,i-Q} \quad Q < i \le 2Q \quad (28)$$

where $[\mathbf{A}]_{\cdot,i}$ denotes the ith column of matrix $\mathbf{A}$ and κ is a free parameter. The corresponding forward model evaluations $\mathcal{Y}_{i,n}$, and weights $u_{i,n}$:

$$\mathcal{Y}_{i,n} = \mathbf{g}(\mathcal{F}_{i,n}) \quad \text{for } 0 \le i \le 2Q \quad (29)$$

$$u_0 = \frac{\kappa}{Q+\kappa}, \quad u_i = \frac{1}{2(Q+\kappa)} \text{ for } 0 < i \le 2Q, \quad (30)$$

where we note that $\kappa = 1/2$ corresponds to uniform weights $u_i = 1/(2Q+1)$.

Solving the weighted linear least squares problems with inputs, outputs, and weights $\{\mathcal{F}_{i,n}, \mathcal{Y}_{i,n}, u_i\}$ yields the solution:

$$\mathbf{b}_n = \bar{\mathbf{y}}_n - \mathbf{A}_n\mathbf{M}\boldsymbol{\phi}_n, \quad (31)$$

$$\mathbf{A}_n = \boldsymbol{\Gamma}_n\mathbf{E}_n^{-1}, \quad (32)$$

where $\mathbf{E}_n$ is the diagonal matrix given in Equation (25), and $\bar{\mathbf{y}}_n$ and $\boldsymbol{\Gamma}_n$ are the sufficient statistics:

$$\bar{\mathbf{y}}_n = \sum_{i=0}^{2Q} u_i \mathcal{Y}_{i,n}, \quad (33)$$

$$\boldsymbol{\Gamma}_n = \sum_{i=0}^{2Q} u_i (\mathcal{Y}_{i,n} - \bar{\mathbf{y}}_n)(\mathcal{F}_{i,n} - \mathbf{M}\boldsymbol{\phi}_n)^\top. \quad (34)$$

We will refer to the method that uses this statistical linearization within the variational framework described in section 4 as Unscented Kitchen Sinks (UKS). One of the key advantages of the UKS over the EKS is that the UKS, like the original UGP, is a 'black-box' method in that it requires no gradient information of the nonlinearity in the likelihood function in order to learn the parameters/hyperparameters and carry out posterior approximation and predictions.

6. Prediction

The predictive distribution over the latent function values $\mathbf{f}_{*\cdot}$ can be computed similarly to Equation (24):

$$p(\mathbf{f}_{*\cdot}|\mathbf{x}_*) = \mathcal{N}(\mathbf{f}_{*\cdot}|\boldsymbol{\mu}_*, \mathbf{E}_*) \text{ with } \boldsymbol{\mu}_* = \mathbf{M}\boldsymbol{\phi}_* \text{ and} \quad (35)$$

$$[\mathbf{E}_*]_{qq\prime} = \boldsymbol{\phi}_*^\top \mathbf{C}_q \boldsymbol{\phi}_* \text{ for } q = q\prime \text{ and } 0 \text{ otherwise,} \quad (36)$$

where $\boldsymbol{\phi}_*$ is the Q-dimensional vector of test features resulting from evaluating $\phi(\mathbf{x}_*)$. We can also predict the noiseless observations by evaluating the integral:

$$\bar{\mathbf{g}}_* = \int \mathbf{g}(\mathbf{f}_{*\cdot}) p(\mathbf{f}_{*\cdot}|\mathbf{x}_*)\, d\mathbf{f}_{*\cdot}, \quad (37)$$

which we can estimate using Monte Carlo averaging.

7. Experiments

In this section we describe the experiments carried out in order to assess the performance and the behavior of our extended kitchen sinks (EKS) and unscented kitchen sinks (UKS) methods. we first look at experiments on small-scale synthetic inversion problems and a binary classification task on the USPS dataset. As these experiments were also carried out by Steinberg & Bonilla (2014) to evaluate the EGP and the UGP, our first goal is to investigate how well random kitchen sinks bases can approximate the original EGP and UGP. Additionally, we are interested in determining whether the complexity of the algorithms can be reduced significantly by having a number of basis functions smaller than the number of training points.

7.1. Synthetic Inversion Problem

In this experiment we generate latent function values ($\mathbf{f}$) from a GP with isotropic squared exponential covariance function (having a signal variance $\sigma_s^2 = 0.8^2$ and a length-scale $\ell = 0.6$) at 1000 input points, $\mathbf{x} \in \mathbb{R}$, which are

Table 1. Performance of the EKS and UKS methods compared to their GP counterparts (EGP and UGP) on a range of synthetic benchmarks when using $D = 200$ features. The mean for each measure is shown and the standard deviation in brackets.

g(f)	Method	SMSE-f*	NLPD-f*	SMSE-g*
linear	EKS	0.03 (0.01)	-0.99 (0.23)	0.03 (0.01)
	UKS	0.04 (0.00)	-0.97 (0.11)	0.04 (0.00)
	EGP	0.03 (0.01)	-1.01 (0.30)	0.03 (0.01)
	UGP	0.03 (0.02)	-0.94 (0.43)	0.03 (0.02)
poly3	EKS	0.02 (0.01)	-1.39 (0.20)	0.01 (0.00)
	UKS	0.06 (0.04)	0.41 (1.69)	0.02 (0.01)
	EGP	0.07 (0.02)	-0.36 (0.68)	0.02 (0.01)
	UGP	0.06 (0.04)	-0.35 (0.62)	0.01 (0.01)
exp	EKS	0.03 (0.01)	-1.29 (0.20)	0.01 (0.00)
	UKS	0.02 (0.01)	-1.26 (0.24)	0.01 (0.00)
	EGP	0.08 (0.04)	0.25 (1.52)	0.04 (0.02)
	UGP	0.03 (0.02)	-1.02 (0.58)	0.02 (0.01)
sin	EKS	0.03 (0.02)	-1.05 (0.19)	0.03 (0.01)
	UKS	0.03 (0.01)	-1.13 (0.10)	0.03 (0.01)
	EGP	0.04 (0.02)	-0.94 (0.25)	0.04 (0.02)
	UGP	0.06 (0.02)	-0.80 (0.22)	0.05 (0.03)
tanh	EKS	0.05 (0.03)	-1.14 (0.24)	0.02 (0.00)
	UKS	0.04 (0.03)	-0.85 (1.00)	0.03 (0.02)
	EGP	0.09 (0.06)	-0.85 (0.24)	0.04 (0.02)
	UGP	0.05 (0.03)	-0.87 (0.22)	0.03 (0.01)

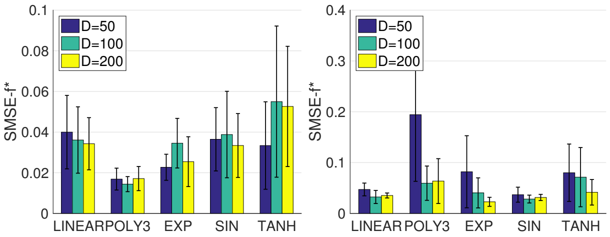

Figure 1. The performance of the EKS (left) and UKS (right) on the synthetic inversion problems as a function of the number of features.

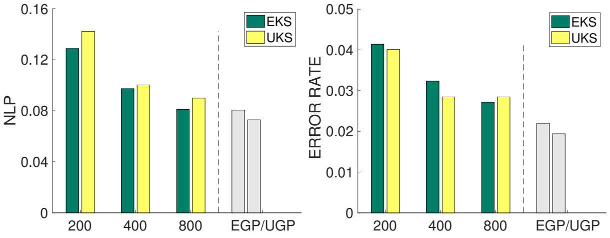

Figure 2. The performance of the EKS and UKS on the binary classification problem for the USPS dataset as a function of the number of bases used. EGP and UGP are the original (full) GP models.

uniformly spaced between $[-2\pi, 2\pi]$. We test our algorithms and the baselines (UGP, EGP) with five simple forward models; an identity function (linear), a 3rd order polynomial with no cross terms (poly3), an exponential function, a sinusoid, and a tangent function. We present the results of 5-fold cross validation (200 training, 800 testing) in Table 1, and also show the behavior of our algorithms as a function of the number of features in Figure 1. We use standardized mean square error (SMSE) and negative log probability density (NLPD) as the performance metrics.

We see in Table 1 that, in general, the EKS and UKS perform similarly to the EGP and UGP algorithms, and sometimes better. For example, the EGP with exponential forward model has larger mean SMSE and mean NLPD with associated high standard deviations, suggesting the algorithm converged sub-optimally in one or more of the folds. More importantly, this comparable or superior performance by the proposed algorithms is attained with significantly less computational cost since $D < N$. Finally, as seen in Figure 1, apart from the UKS with 50 basis functions, both algorithms appear to be relatively robust to using smaller numbers of features, which translates into less computational cost while maintaining similar performance levels. Analogous results are observed for the NLPD as a function of the number of features, see the supplementary material for details.

7.2. Binary Handwritten Digit Classification

This is a binary classification task to distinguish between images of the handwritten digits ‘3’ and ‘5’ in the USPS digits datasets (Rasmussen & Williams, 2006). There are 767 images in the training set, and 773 images in the test set. We use a logistic sigmoid as a forward model in this task and the same settings as in the original experiments for covariance functions and observation variance. The main aim of this experiment is to benchmark the performance of the EKS and UKS against the EGP and UGP for varying numbers of basis functions, as shown in Figure 2. It is important to emphasize that Steinberg & Bonilla (2014) showed that the EGP and UGP outperform or perform similarly to hard-coded inference methods based on variational inference and expectation propagation. We observe that there is a detriment in performance with a small number of features ($D = 200$), which is to be expected since the number of training points is $N = 767$, and so the exact covariance would be harder to represent with a low number of random basis functions. However, the performance of the EKS and UKS approaches that of the UGP and EGP with 400 basis functions, which indicates that our random-feature approaches are reasonable approximations to the original GP model. More importantly, for problems with a large number of training points, the EGP and the UGP are simply unfeasible, and this is the subject of discussion in the next section.

Table 2. The performance of the models on the MNIST dataset for the task of classifying the even digits vs the odd digits.

	NLP		Error Rate	
	D = 1000	D = 2000	D = 1000	D = 2000
EKS	0.129	0.088	0.043	0.026
UKS	0.129	0.088	0.043	0.026
HMG	0.069		0.022	
DB	0.068		0.022	

7.3. Large Scale Classification

Here we present results on a larger application on the MNIST dataset, which contains examples of handwritten digits, 50,000 for training, 10,000 for validation and 10,000 for testing. In our experiments, we always train on 60,000 examples that include the training and the validation set and tune the parameters of our models via optimization of the variational bound. This is probably a disadvantage when compared to other approaches that use cross-validation but our goal is only to show that our models can achieve competitive performance at this scale.

Odd digits vs even digits. We first consider the binary classification problem of distinguishing the odd digits from the even digits, a task that has also been investigated by Hensman et al. (2015a). The results are shown in Table 2, when using a logistic sigmoid function as the forward model in our methods (EKS and UKS) and the methods by Hensman et al. (2015a) and Dezfouli & Bonilla (2015). We refer to these methods as HMG and DB respectively. As before, we report the mean negative log probability and the error rate on the test set for different number of features. We see that our methods achieve similar performance to HMG and DB when using 2000 features. While DB is an inducing-point approach that uses 2000 inducing points fixed via clustering, HMG uses 200 inducing points with the extra overhead of learning their locations. Overall, we conclude that random feature approximations in our extended and unscented models are competitive with the state-of-the-art approaches to sparse GPs.

Multi-class classification: One of the contributions of our approach with respect to the original EGP and UGP models is its scalability to a large number of observations when having multiple outputs. In this experiment we applied our algorithm to the task of classifying all digits on MNIST using a softmax forward model. Our methods (EKS and UKS) achieved similar performance. For example, the EKS obtained an error rate of 4.75% and an NLP of -0.1887, when using $D = 1000$ features. When we increased the number of features to $D = 2000$, it attained an error rate of 3.81% and an NLP of -0.1304. These error rates are lower than that reported by Gal et al. (2014) of 5.95%, while Dezfouli & Bonilla (2015) reported an error rate of 2.51%. As a reference, linear classifiers achieve around 12% error rate on this task while the state of the art is less than 1%.

7.4. Seismic Inversion

In this experiment we perform an inversion of a seismic survey line using a one-dimensional seismic sensor forward model. The goal is to infer both the geometry of the interfaces between subsurface geological layers and the seismic propagation velocity within each layer from noisy surface observations of the sound reflection times, $\mathbf{Y}$. Our dataset is part of a real seismic survey of the Otway basin region in Victoria, Australia. The survey is interpreted, specifying reflection time estimates rather than raw amplitudes. Our transect contains 113 sites with four interface reflections (layers) per site.

The inputs, $\mathbf{X}$, are the surface positions (meters) of ground seismic sensors, and we use the following forward model,

$$\mathbf{g}_p^{\text{time}} = \begin{cases} 2\left(\frac{\mathbf{f}_p^{\text{depth}} - \mathbf{f}_{p-1}^{\text{depth}}}{\mathbf{f}_p^{\text{vel}}}\right) + \mathbf{g}_{p-1}^{\text{time}}, & \text{if } p > 0 \\ 2\left(\frac{\mathbf{f}_0^{\text{depth}}}{\mathbf{f}_0^{\text{vel}}}\right), & \text{if } p = 0 \end{cases} \tag{38}$$

where there are P output tasks – $\mathbf{g}_p^{\text{time}}$, the reflection times from each layer. These outputs depend on two latent input tasks; $\mathbf{f}_p^{\text{depth}}(\mathbf{X})$, the geological depth of layer p corresponding to each surface location, and $\mathbf{f}_p^{\text{vel}}(\mathbf{X})$, the velocity of layer p below each surface input location. The layer depths are also clipped to be at least as deep as the previous, $\mathbf{f}_p^{\text{depth}} = \max\{\mathbf{f}_p^{\text{depth}}, \mathbf{f}_{p-1}^{\text{depth}}\}\ \forall p$, to enforce a geologically valid depth structure.

We wish to infer the latent depths and velocities of each of the four layers. Consequently, $Q = 2P$, and the problem is under-constrained since there is an infinite set of layer depths and velocities that could result in the observed reflection times. Thus, we cannot evaluate the quality of our inference by its ability to predict a single truth, but instead must assess its ability to correctly condition our prior on these constraining observations. For this experiment, we define flat priors for each latent function centered on average depths and velocities of each layer. A baseline inference using MCMC has been applied to sample the posterior distribution. See the supplementary material for more details.

The results of both the EKS and UKS methods were very similar. Figure 3 shows the results from the EKS method inferring the distribution over layer boundary depths (left) and seismic transmission velocities (right). The MCMC solution is overlaid. Clearly the MCMC solution is in agreement with the structure of the layer velocities and depths. However, we do note that the MCMC draws are smoother. This may be because the limited number of radial basis

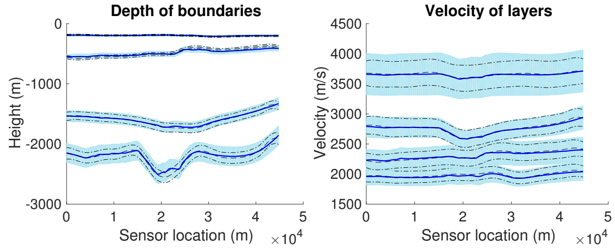

Figure 3. Results of the EKS on the seismic inversion problem. The inferred layer boundaries (left) and seismic velocities (right) are shown in blue, indicating the predictive means and standard deviation envelopes. Draws from the MCMC inversion are overlaid in dotted black.

functions used in the MCMC has restricted its ability to infer high frequency details. Additional evaluation of the results can be found in the supplementary material.

8. Related work

Most previous work in the GP community has focused on addressing the scalability, multi-task learning or non-linear likelihood challenges in isolation. For example, with regards to scalability, the seminal work of Quiñonero-Candela & Rasmussen (2005) allowed the community to understand most sparse approximations in GP models from a probabilistic perspective, and the framework of Titsias (2009) has become the underpinning machinery of most modern scalable approaches to GP regression and classification. Nevertheless, the scalability problem in GP models continues to be an intensive area of research, with the recently developed distributed inference framework of Gal et al. (2014), and the variational inference frameworks for scalable GP regression and classification by Hensman et al. (2013) and Hensman et al. (2015a), respectively. With regards to multi-output and multi-task learning, one of the most notable approaches has been developed by Álvarez & Lawrence (2009) using the convolution formalism, although their later work also focuses on developing efficient inference algorithms for such models (Álvarez & Lawrence, 2011). Finally, concerning non-linear likelihoods, Opper & Archambeau (2009) presented the seemingly surprising (but powerful) result of estimating a full Gaussian posterior for models with GP priors and general i.i.d likelihoods efficiently using variational inference.

Although the work by Rahimi & Recht (2008; 2009) has been highly influential in the area of deterministic kernel machines, it is surprising that their random kitchen sinks (RKS) approximations had not been investigated more thoroughly in probabilistic kernel frameworks such as Gaussian process models. For example, Dai et al. (2014) have used random features for kernel machines, although their main focus is on non-probabilistic approaches that are underpinned by convex optimization. Only very recently, Yang et al. (2015) have investigated the problem of developing scalable and flexible probabilistic kernel approaches for regression using fast approximations to the original RKS (Le et al., 2013).

Contemporary to our work, Hensman et al. (2015b) and Dezfouli & Bonilla (2015) have proposed scalable approaches to inference in GP models with general likelihoods. We see our extended and unscented kitchen sinks methods as alternative approaches to their work that builds upon the inducing-point formalism of Titsias (2009). A thorough comparison and evaluation of both approaches is an interesting area for immediate future work.

9. Conclusion & Discussion

We have presented the EKS and the UKS methods for posterior inference in GP models with nonlinear likelihoods. These methods are multi-task and scalable generalizations of the EGP and UGP algorithms of Steinberg & Bonilla (2014). The UKS, like the UGP, is a ‘black-box’ method in that it requires no gradients of the nonlinearity in the likelihood.

We have shown in our experiments that, by using RKS approximations, we can achieve similar prediction performance to the original algorithms, while drastically reducing their computational complexity and increasing their scalability. Furthermore, we have demonstrated the EKS and UKS successfully performing a multi-task Bayesian seismic inversion — which is an ideal use case for these algorithms as a fast and scalable alternative to MCMC. Because our linearization algorithms are local and adaptive, the EKS (and the original EGP) can be seen as refined (iterated) versions of the Laplace approximation, where the linearization gets updated at every iteration as a function of the variational parameters. The UKS (and the original UGP) can provide us with more elaborate and effective ways to propagate the first and second moments through nonlinearities.

Since the approximating model in Equations (7) and (8) is no longer a GP model, other inference methods such as MCMC can be used in practice. In fact, as mentioned in §7.4, we implemented MCMC algorithms for these problems but found them too slow to be applied to large datasets. Nevertheless, further study and improvement of sampling algorithms such as MCMC is indeed a very promising research direction. On an alternative vein, we would also like to use a stochastic gradient optimizer for learning all model (hyper)parameters, and extend the posterior representation to a mixture of Gaussians, in a similar fashion as Nguyen & Bonilla (2014) and Gershman et al. (2012).

Acknowledgments

DS and AR were supported by the Science Industry Endowment Fund (RP 04-174) Big Data Knowledge Discovery project. NICTA is funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program.

References

Álvarez, Mauricio and Lawrence, Neil D. Sparse convolved Gaussian processes for multi-output regression. In *NIPS*, pp. 57–64. 2009.

Álvarez, Mauricio A and Lawrence, Neil D. Computationally efficient convolved multiple output Gaussian processes. *JMLR*, 12(5):1459–1500, 2011.

Dai, Bo, Xie, Bo, He, Niao, Liang, Yingyu, Raj, Anant, Balcan, Maria-Florina F, and Song, Le. Scalable kernel methods via doubly stochastic gradients. In *NIPS*, 2014.

Dezfouli, Amir and Bonilla, Edwin V. Scalable inference for Gaussian process models with black-box likelihoods. In *NIPS*. 2015.

Gal, Yarin, van der Wilk, Mark, and Rasmussen, Carl. Distributed variational inference in sparse Gaussian process regression and latent variable models. In *NIPS*. 2014.

Geist, Matthieu and Pietquin, Olivier. Statistically linearized recursive least squares. In *IEEE International Workshop on Machine Learning for Signal Processing*, 2010.

Gershman, Samuel J., Hoffman, Matthew D., and Blei, David M. Nonparametric variational inference. In *ICML*, 2012.

Hensman, James, Fusi, Nicolo, and Lawrence, Neil D. Gaussian processes for big data. In *UAI*, 2013.

Hensman, James, Matthews, Alexander, and Ghahramani, Zoubin. Scalable variational Gaussian process classification. In *AISTATS*, 2015a.

Hensman, James, Matthews, Alexander G, Filippone, Maurizio, and Ghahramani, Zoubin. MCMC for variationally sparse Gaussian processes. In *NIPS*. 2015b.

Julier, S.J. and Uhlmann, J.K. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, Mar 2004.

Le, Quoc, Sarlos, Tamas, and Smola, Alex. Fastfood - approximating kernel expansions in loglinear time. In *ICML*, 2013.

Nguyen, Van Trung and Bonilla, Edwin. Collaborative multi-output Gaussian processes. In *UAI*, 2014.

Opper, Manfred and Archambeau, Cédric. The variational Gaussian approximation revisited. *Neural Computation*, 21(3):786–792, 2009.

Powell, M. J. D. A direct search optimization method that models the objective and constraint functions by linear interpolation. In *Advances in Optimization and Numerical Analysis*. 1994.

Powell, M. J. D. Direct search algorithms for optimization calculations. *Acta Numerica*, 7:287–336, 1998.

Powell, M. J. D. The BOBYQA algorithm for bound constrained optimization without derivatives. Technical report, Department of Applied Mathematics and Theoretical Physics, Cambridge England, 2009.

Quiñonero-Candela, Joaquin and Rasmussen, Carl Edward. A unifying view of sparse approximate Gaussian process regression. *JMLR*, 6:1939–1959, 2005.

Rahimi, Ali and Recht, Benjamin. Random features for large-scale kernel machines. In *NIPS*. 2008.

Rahimi, Ali and Recht, Benjamin. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In *NIPS*. 2009.

Rasmussen, Carl Edward and Williams, Christopher K. I. *Gaussian processes for machine learning*. The MIT Press, 2006.

Steinberg, Daniel M and Bonilla, Edwin V. Extended and unscented Gaussian processes. In *NIPS*. 2014.

Titsias, Michalis. Variational learning of inducing variables in sparse Gaussian processes. In *AISTATS*, 2009.

Yang, Zichao, Wilson, Andrew Gordon, Smola, Alexander J., and Song, Le. á la carte - learning fast kernels. In *AISTATS*, 2015.