Wilkinson, William; Chang, Paul; Riis Andersen, Michael; Solin, Arno

State Space Expectation Propagation: Efficient Inference Schemes for Temporal Gaussian Processes

Published in:
Proceedings of the 37th International Conference on Machine Learning

Published: 13/07/2020

Document Version
Publisher's PDF, also known as Version of record

Please cite the original version:
Wilkinson, W., Chang, P., Riis Andersen, M., & Solin, A. (2020). State Space Expectation Propagation: Efficient Inference Schemes for Temporal Gaussian Processes. In *Proceedings of the 37th International Conference on Machine Learning* (pp. 10270-10281). (Proceedings of Machine Learning Research ; Vol. 119). JMLR.
http://proceedings.mlr.press/v119/wilkinson20a.html

State Space Expectation Propagation: Efficient Inference Schemes for Temporal Gaussian Processes

William J. Wilkinson [1] **Paul E. Chang** [1] **Michael Riis Andersen** [2] **Arno Solin** [1]

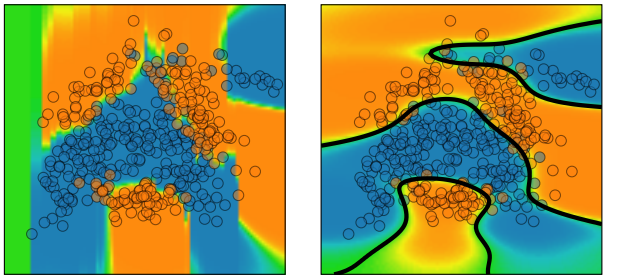
Abstract

We formulate approximate Bayesian inference in non-conjugate temporal and spatio-temporal Gaussian process models as a simple parameter update rule applied during Kalman smoothing. This viewpoint encompasses most inference schemes, including expectation propagation (EP), the classical (Extended, Unscented, *etc.*) Kalman smoothers, and variational inference. We provide a unifying perspective on these algorithms, showing how replacing the power EP moment matching step with linearisation recovers the classical smoothers. EP provides some benefits over the traditional methods via introduction of the so-called cavity distribution, and we combine these benefits with the computational efficiency of linearisation, providing extensive empirical analysis demonstrating the efficacy of various algorithms under this unifying framework. We provide a fast implementation of all methods in JAX.

1. Introduction

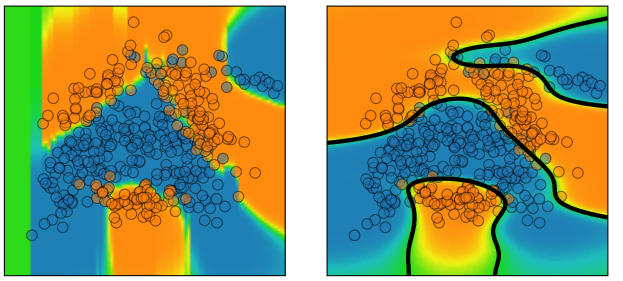
Gaussian processes (GPs, Rasmussen & Williams, 2006) are a nonlinear probabilistic modelling tool that combine well calibrated uncertainty estimates with the ability to encode prior information, and as such they are an increasingly effective method for many difficult machine learning tasks. The well known limitations of GPs are *(i)* their cubic scaling in the number of data, and *(ii)* their intractability when the observation model is non-Gaussian.

For *(i)*, a wide variety of methods have been proposed (*e.g.* Hensman et al., 2013; Salimbeni & Deisenroth, 2017; Wang et al., 2019), with perhaps the most common being the sparse-GP approach (Quiñonero-Candela & Rasmussen, 2005) which summarises the GP posterior through a subset of 'inducing' points. However, when the data exhibits a natural ordering—as in temporal or spatio-temporal tasks—many GP priors can be rewritten in closed-form in terms of stochastic differential equations (SDEs, Särkkä & Solin, 2019), allowing for linear-time exact inference via Kalman filtering (Hartikainen & Särkkä, 2010; Reece & Roberts, 2010). This link is beneficial in scenarios such as climate modelling, or audio signal analysis, which exhibit both high and low-frequency behaviour. A sparse-GP analogy to the Shannon–Nyquist theorem (Tobar, 2019) tells us that audio signals, for example, necessarily require tens of thousands of inducing points per second of data, rendering sparse ap-



Filtering / Forward pass $\rightarrow$ $\leftarrow$ Smoothing / Backward pass

(a) EKF forward and RTS backward pass



Filtering / Forward pass $\rightarrow$ $\leftarrow$ Smoothing / Backward pass

(b) Extended Expectation Propagation (EEP)

Figure 1. Filtering and smoothing in the *Banana* classification task. Training data represented by coloured points, the decision boundaries by black lines, and the predictive mean for the class label by colour map. The vertical dimension is treated as the 'spatial' input and the horizontal as the sequential ('temporal') dimension. Forward sweep on the left, backward sweep on the right. Top panels (a) show the EKF; bottom (b) is the 2nd iteration of EEP.

[1]Department of Computer Science, Aalto University, Finland [2]Department of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark. Correspondence to: William J. Wilkinson <william.wilkinson@aalto.fi>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

proximations infeasible for all but the shortest of time series. This strongly motivates our reformulation of temporal GPs as SDEs for efficient inference.

For limitation *(ii)*, a wide variety of approximative inference methods have been considered, with the current gold-standard being various sampling schemes (see Gelman et al., 2013, for an overview), variational methods (Opper & Archambeau, 2009; Titsias, 2009; Wainwright & Jordan, 2008), and expectation propagation (EP, Bui et al., 2017; Jylänki et al., 2011; Kuss & Rasmussen, 2006; Minka, 2001). Despite Minka's original work having its foundations in filtering and smoothing, all the special characteristics of temporal models have not been thoroughly leveraged in the machine learning community. We extend recent work on approximate inference under the state space paradigm, and provide a framework that unifies EP and traditional methods such as the Extended (EKF, Bar-Shalom et al., 2001) and Unscented Kalman filters (UKF, Julier et al., 1995; 2000). Our framework provides ways to trade off accuracy and computation, and we show that an iterated version of the EKF with EP-style updates can be efficient and easy to implement, whilst providing good performance in cases where the likelihood model is not locally highly nonlinear. For completeness, we also formulate variational inference in the same setting.

We show that such tools are not limited to one-dimensional-input models; instead they only require us to treat a single dimension of the data sequentially (regardless of whether it is actually ordered, or represents time). We apply our methods to multi-dimensional problems such as 2D classification (see Fig. 1) and 2D log-Gaussian Cox processes.

Our main contributions are: *(i)* We formulate EP as a Kalman smoother, showing how it unifies many classical smoothing methods, providing an efficient framework for inference in temporal GPs. *(ii)* We show how to rewrite common machine learning tasks (likelihoods) into canonical state space form, and provide extensive analysis demonstrating performance in many modelling scenarios. *(iii)* We show how the state space framework can be extended beyond the one-dimensional case, applying it to multidimensional classification and regression tasks, where we still enjoy linear-time inference over the sequential dimension. *(iv)* We provide fast JAX code for inference and learning with all the methods described in this paper, available at https://github.com/AaltoML/kalman-jax.

2. Background

Gaussian processes (GPs, Rasmussen & Williams, 2006) form a non-parametric family of probability distributions on function spaces, and are completely characterized by a mean function $\mu(t) : \mathbb{R} \to \mathbb{R}$ and a covariance function $\kappa(t, t') : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$. Let $\{(t_k, y_k)\}_{k=1}^n$ denote a set of n input–output pairs for a time series (we first consider the 1D input case), then GP models typically take the form

$$f(t) \sim \mathcal{GP}(\mu(t), \kappa(t, t')), \quad \mathbf{y} \,|\, \mathbf{f} \sim \prod_{k=1}^{n} p(y_k \,|\, f(t_k)), \quad (1)$$

which defines the prior for the latent function $f : \mathbb{R} \to \mathbb{R}$ and the observation model for y_k. For Gaussian observation models, the posterior distribution $p(\mathbf{f} \,|\, \mathbf{y})$ is also Gaussian and can be obtained analytically, but non-Gaussian likelihoods render the posterior intractable and approximate inference methods must be applied.

2.1. State Space Models for Gaussian Processes

In signal processing, the canonical (discrete-time) state space model formulation is (*e.g.*, Bar-Shalom et al., 2001):

$$\mathbf{x}_k = \mathbf{g}(\mathbf{x}_{k-1}, \mathbf{q}_k), \quad (2a)$$
$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \boldsymbol{\sigma}_k), \quad (2b)$$

for time instances t_k, where $\mathbf{x}_k \in \mathbb{R}^s$ is the discrete-time state sequence, $\mathbf{y}_k \in \mathbb{R}^d$ is a measurement sequence, $\mathbf{q}_k \sim \mathrm{N}(\mathbf{0}, \mathbf{Q}_k)$ is the process noise, and $\boldsymbol{\sigma}_k \sim \mathrm{N}(\mathbf{0}, \boldsymbol{\Sigma}_k)$ is Gaussian measurement noise. The model dynamics (prior) are defined by the nonlinear mapping $\mathbf{g}(\cdot, \cdot)$, while the observation/measurement model (likelihood) is given in terms of the mapping $\mathbf{h}(\cdot, \cdot)$. We restrict the model dynamics $\mathbf{g}(\cdot, \cdot)$ to be linear-Gaussian—defining an s-dimensional Gaussian process. The dynamical model Eq. (2a) becomes,

$$\mathbf{x}_k = \mathbf{A}_k \mathbf{x}_{k-1} + \mathbf{q}_k, \quad \mathbf{q}_k \sim \mathrm{N}(\mathbf{0}, \mathbf{Q}_k), \quad (3)$$

which is characterised by the transition matrix $\mathbf{A}_k$ and process noise covariance $\mathbf{Q}_k$. Whilst we restrict our interest to latent Gaussian dynamics, the inference methods presented later apply to more general nonlinear state estimation settings, where the prior is not necessarily a GP (see Sec. 5).

The motivation for linking the machine learning GP formalism with state space models comes from the special structure in temporal or spatio-temporal problems, where the data points have a natural ordering with respect to the temporal dimension. If the GP prior in Eq. (1) admits a Markovian structure, the model can be rewritten in the form of Eq. (3). We leverage the link between the kernel and state space forms of GPs (Särkkä & Solin, 2019; Särkkä et al., 2013), which comes through linear time-invariant SDEs:

$$\dot{\mathbf{x}}(t) = \mathbf{F}\,\mathbf{x}(t) + \mathbf{L}\,\mathbf{w}(t), \quad \text{such that} \quad \mathbf{f}(t) = \mathbf{H}\mathbf{x}(t), \quad (4)$$

where $\mathbf{w}(t)$ is a white noise process, and $\mathbf{F} \in \mathbb{R}^{s \times s}$, $\mathbf{L} \in \mathbb{R}^{s \times v}$, $\mathbf{H} \in \mathbb{R}^{m \times s}$ are the feedback, noise effect, and measurement matrices, respectively. Many widely used covariance functions admit this form exactly or approximately (*e.g.*, the Matérn class, polynomial, noise, constant, squared-exponential, rational quadratic, periodic, and sums/products

thereof). Särkkä & Solin (2019) discuss methods for constructing the required matrices for many GP models. Key to this formulation is that linear time-invariant SDEs are guaranteed to have a closed-form discrete-time solution in the form of a linear Gaussian state space model as in Eq. (3). We leverage this link in order to apply sequential inference schemes to temporal and spatio-temporal GP models.

2.2. Extended and Unscented State Estimation

Many nonlinear variants of the Kalman filter have been developed to deal with the measurement model in Eq. (2b) (see Särkkä, 2013, for an overview). The most widely known are the EKF (*e.g.* Bar-Shalom et al., 2001) and UKF (Julier et al., 1995). The EKF linearises $\mathbf{h}(\mathbf{x}_k, \boldsymbol{\sigma}_k)$ via a first-order Taylor series expansion, which in turn results in linear Gaussian approximations to all the required Kalman update equations. We discuss the approach in detail in Sec. 3.2.

The UKF is a member of a wider class of Gaussian filtering methods (Ito & Xiong, 2000), which approximate the Kalman update equations via statistical linearisation rather than a Taylor expansion. Statistical linearisation is generally intractable, involving expectations that must be computed numerically (shown in App. B). Choosing the Unscented transform as the numerical integration method results in the UKF, but other sigma-point methods can also be used (see, *e.g.*, Ito & Xiong, 2000; Kokkala et al., 2016; Šimandl & Duník, 2009; Wu et al., 2005; 2006)—*e.g.* using Gauss–Hermite cubature gives the Gauss–Hermite Kalman filter.

2.3. Expectation Propagation

Expectation propagation (EP) is a general framework for approximating probability distributions proposed by Minka (2001). EP and its extension Power-EP (PEP, Minka, 2004) have been extensively studied for Gaussian process models and shown to provide state-of-the-art results (Bui et al., 2017; Jylänki et al., 2011; Kuss & Rasmussen, 2006). EP approximates the target distribution $p(\mathbf{f} \mid \mathbf{y})$ with an approximation $q(\mathbf{f})$ that factorises in the same way as the target,

$$p(\mathbf{f} \mid \mathbf{y}) \propto \prod_{k=1}^{n} p(\mathbf{y}_k \mid \mathbf{f}_k) p(\mathbf{f}) \approx q(\mathbf{f}) \propto \prod_{k=1}^{n} q_k^{\text{site}}(\mathbf{f}_k) p(\mathbf{f}) \quad (5)$$

The likelihood approximations $q_k^{\text{site}}(\mathbf{f}_k) \approx p(\mathbf{y}_k \mid \mathbf{f}_k)$ are usually referred to as *sites*. For GP models, the sites are chosen to be Gaussians and hence the global approximation $q(\mathbf{f})$ is also Gaussian. The sites are updated in an iterative fashion by minimizing local Kullback–Leibler divergences between the so-called *tilted distributions*, $\hat{p}_k(\mathbf{f}_k) = \frac{1}{Z_k} p(\mathbf{y}_k \mid \mathbf{f}_k) q_k^{\text{cav}}(\mathbf{f}_k)$, and its approximation using the site,

$$q_k^{\text{site}*} = \arg\min_{q_k^{\text{site}}} \text{KL}\left[\hat{p}_k(\mathbf{f}_k) \,\|\, q_k^{\text{site}}(\mathbf{f}_k) q_k^{\text{cav}}(\mathbf{f}_k)\right], \quad (6)$$

where $q_k^{\text{cav}}(\mathbf{f}_k)$ is the *cavity distribution*: $q_k^{\text{cav}}(\mathbf{f}_k) \propto q(\mathbf{f}_k)/q_k^{\text{site}}(\mathbf{f}_k)$. The KL-divergence in Eq. (6) is minimized using *moment matching* (Minka, 2001), *i.e.* q_k is chosen such that the approximation $q_k^{\text{site}} q_k^{\text{cav}}$ matches the first two moments of the tilted distribution $\hat{p}_k$. This process is iterated for all sites until convergence. Power EP is a generalization of EP, where the KL-divergence is generalized to the α-divergence (Minka, 2005). Minka (2004) showed that PEP can be implemented using the EP algorithm, by raising the site terms in the tilted distributions to a power of α.

3. Methods

We consider non-conjugate (*i.e.* non-Gaussian likelihood) Gaussian process models with input t, *i.e.* time, which have a dual kernel (*left*) and discrete state space (*right*) form for the prior (Särkkä et al., 2013),

$$\mathbf{f}(t) \sim \mathcal{GP}\big(\boldsymbol{\mu}(t), \mathbf{K}_{\boldsymbol{\theta}}(t, t')\big), \quad \mathbf{x}_k = \mathbf{A}_{\boldsymbol{\theta},k} \mathbf{x}_{k-1} + \mathbf{q}_k, \quad (7)$$

where $\mathbf{f}(t) = \big(f^{(1)}(t), \ldots, f^{(m)}(t)\big)^\top \in \mathbb{R}^m$ are GPs, $\mathbf{x}_k = \big(\mathbf{x}_k^{(1)}, \ldots, \mathbf{x}_k^{(m)}\big)^\top \in \mathbb{R}^s$ is the latent state vector containing the GP dynamics, and $\mathbf{y}_k \in \mathbb{R}^d$ are observations. Each $\mathbf{x}_k^{(i)}$ contains the state dynamics for one GP. Using notation $\mathbf{f}_k = \mathbf{f}(t_k)$, we define a time-varying linear map $\mathbf{H}_k \in \mathbb{R}^{m \times s}$ from state space to function space, such that $\mathbf{f}_k = \mathbf{H}_k \mathbf{x}_k$ (the time-varying mapping allows us to naturally incorporate spacial inducing points when considering multidimensional input models, see Sec. 3.6). The likelihood (*left*) / state observation model (*right*) are

$$\mathbf{y}_k \sim p(\mathbf{y}_k \mid \mathbf{f}_k) \quad \text{vs.} \quad \mathbf{y}_k = \mathbf{h}(\mathbf{f}_k, \boldsymbol{\sigma}_k). \quad (8)$$

Measurement model $\mathbf{h}(\mathbf{f}_k, \boldsymbol{\sigma}_k)$ is a (nonlinear) function of $\mathbf{f}_k$ and observation noise $\boldsymbol{\sigma}_k \sim \text{N}(\mathbf{0}, \boldsymbol{\Sigma}_k)$, and can generally be derived for continuous likelihoods or approximated for discrete ones by letting $\mathbf{h}(\mathbf{f}_k, \boldsymbol{\sigma}_k) \approx \mathbb{E}[\mathbf{y}_k \mid \mathbf{f}_k] + \boldsymbol{\varepsilon}_k$, $\boldsymbol{\varepsilon}_k \sim \text{N}(\mathbf{0}, \text{Cov}[\mathbf{y}_k \mid \mathbf{f}_k])$. See Sec. 4 and App. I for derivations of some common models. We aim to calculate the posterior over the states, $p(\mathbf{x}_k \mid \mathbf{y}_1, \ldots, \mathbf{y}_n)$, known as the *smoothing* solution, which can be obtained via application of a Gaussian filter (to obtain the *filtering* solution $p(\mathbf{x}_k \mid \mathbf{y}_1, \ldots, \mathbf{y}_k)$) followed by a Gaussian smoother. If $\mathbf{h}(\cdot, \cdot)$ is linear, *i.e.* $p(\mathbf{y}_k \mid \mathbf{f}_k)$ is Gaussian, then the Kalman filter and Rauch–Tung–Striebel (RTS, Rauch et al., 1965) smoother return the closed-form solution.

3.1. Power EP as a Gaussian Smoother

Our inference methods approximate the filtering distributions with Gaussians, $p(\mathbf{x}_k \mid \mathbf{y}_{1:k}) \approx \text{N}(\mathbf{x}_k \mid \mathbf{m}_k^{\text{filt}}, \mathbf{P}_k^{\text{filt}})$. The prediction step remains the same as in the standard Kalman filter: $\mathbf{m}_k^{\text{pred}} = \mathbf{A}_{\boldsymbol{\theta},k} \mathbf{m}_{k-1}^{\text{filt}}$, and $\mathbf{P}_k^{\text{pred}} = \mathbf{A}_{\boldsymbol{\theta},k} \mathbf{P}_{k-1}^{\text{filt}} \mathbf{A}_{\boldsymbol{\theta},k}^\top + \mathbf{Q}_{\boldsymbol{\theta},k}$. The resulting distribution provides a means by which to calculate the EP cavity,

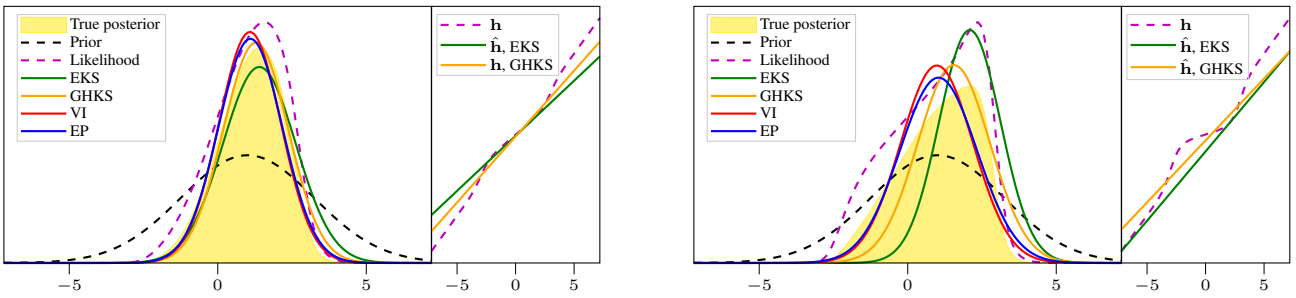


Figure 2. Comparison between EP, VI and iterated linearisation (EKS, GHKS). When measurement function $\mathbf{h}$ is approximately linear in the region of the prior (or the cavity / posterior in the full algorithm), (**left**), linearisation $\hat{\mathbf{h}}$ provides a similar result to EP / VI. When $\mathbf{h}$ is highly nonlinear (**right**), the posterior approximations have different properties. 20-point Gauss–Hermite quadrature used for all methods except EKS. All methods are iterated 10 times except EP which does not require iteration for a single data point.

$q_k^{\text{cav}}(\mathbf{f}_k) = \mathrm{N}(\mathbf{f}_k \,|\, \boldsymbol{\mu}_k^{\text{cav}}, \boldsymbol{\Sigma}_k^{\text{cav}})$, on the first forward pass:

$$\boldsymbol{\mu}_k^{\text{cav}} = \mathbf{H}_k \mathbf{m}_k^{\text{pred}}, \quad \boldsymbol{\Sigma}_k^{\text{cav}} = \mathbf{H}_k \mathbf{P}_k^{\text{pred}} \mathbf{H}_k^\top. \tag{9}$$

In this sense, we can view the first pass of the Kalman filter as an effective way to *initialise* the EP parameters. To account for the non-Gaussian likelihood in the update step we follow Nickisch et al. (2018), introducing an intermediary step in which the parameters of the *sites*, $q_k^{\text{site}}(\mathbf{f}_k) = \mathrm{N}(\mathbf{f}_k \,|\, \boldsymbol{\mu}_k^{\text{site}}, \boldsymbol{\Sigma}_k^{\text{site}})$, are set via *moment matching* and stored before continuing with the Kalman updates.

This PEP formulation, with power α, makes use of the fact that the required moments can be calculated via the derivatives of the log-normaliser, $\mathcal{L}_k$, of the tilted distribution (see Seeger, 2005). Letting $\boldsymbol{\nabla}\mathcal{L}_k \in \mathbb{R}^m$ and $\boldsymbol{\nabla}^2\mathcal{L}_k \in \mathbb{R}^{m\times m}$ be the Jacobian and Hessian of $\mathcal{L}_k$ w.r.t. $\boldsymbol{\mu}_k^{\text{cav}}$ respectively, this gives the following site update rule,

Power expectation propagation

$$\begin{aligned}
\mathcal{L}_k &= \log \mathbb{E}_{\mathrm{N}(\mathbf{f}_k \,|\, \boldsymbol{\mu}_k^{\text{cav}}, \boldsymbol{\Sigma}_k^{\text{cav}})}\left[p^\alpha(\mathbf{y}_k \,|\, \mathbf{f}_k)\right], \\
\boldsymbol{\Sigma}_k^{\text{site}} &= -\alpha \left(\boldsymbol{\Sigma}_k^{\text{cav}} + \left(\boldsymbol{\nabla}^2\mathcal{L}_k\right)^{-1}\right), \\
\boldsymbol{\mu}_k^{\text{site}} &= \boldsymbol{\mu}_k^{\text{cav}} - \left(\boldsymbol{\nabla}^2\mathcal{L}_k\right)^{-1} \boldsymbol{\nabla}\mathcal{L}_k.
\end{aligned} \tag{10}$$

After the mean and covariance of our new likelihood approximation have been calculated, we proceed with a modified set of linear Kalman updates,

$$\begin{aligned}
\mathbf{S}_k &= \mathbf{H}_k \mathbf{P}_k^{\text{pred}} \mathbf{H}_k^\top + \boldsymbol{\Sigma}_k^{\text{site}}, \\
\mathbf{K}_k &= \mathbf{P}_k^{\text{pred}} \mathbf{H}_k^\top \mathbf{S}_k^{-1}, \\
\mathbf{m}_k^{\text{filt}} &= \mathbf{m}_k^{\text{pred}} + \mathbf{K}_k(\boldsymbol{\mu}_k^{\text{site}} - \mathbf{H}_k \mathbf{m}_k^{\text{pred}}), \\
\mathbf{P}_k^{\text{filt}} &= \mathbf{P}_k^{\text{pred}} - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^\top.
\end{aligned} \tag{11}$$

As in Wilkinson et al. (2019), we augment the RTS smoother with another moment matching step where the cavity distribution is calculated by removing (a fraction α of) the local site from the marginal smoothing distribution, *i.e.* the posterior, $p(\mathbf{x}_k \,|\, \mathbf{y}_{1:n}) = \mathrm{N}(\mathbf{x}_k \,|\, \mathbf{m}_k^{\text{post}}, \mathbf{P}_k^{\text{post}})$,

$$\begin{aligned}
\boldsymbol{\Sigma}_k^{\text{cav}} &= \left[\left(\mathbf{H}_k \mathbf{P}_k^{\text{post}} \mathbf{H}_k^\top\right)^{-1} - \alpha\left(\boldsymbol{\Sigma}_k^{\text{site}}\right)^{-1}\right]^{-1}, \\
\boldsymbol{\mu}_k^{\text{cav}} &= \boldsymbol{\Sigma}_k^{\text{cav}}\left[\left(\mathbf{H}_k \mathbf{P}_k^{\text{post}} \mathbf{H}_k^\top\right)^{-1} \mathbf{H}_k \mathbf{m}_k^{\text{post}} - \alpha\left(\boldsymbol{\Sigma}_k^{\text{site}}\right)^{-1} \boldsymbol{\mu}_k^{\text{site}}\right].
\end{aligned} \tag{12}$$

Moment matching is again performed via Eq. (10) using this new cavity. The site parameters, $\boldsymbol{\mu}_k^{\text{site}}$, $\boldsymbol{\Sigma}_k^{\text{site}}$, are stored to be used on the next forward (filtering) pass. App. F discusses methods for avoiding numerical issues that could occur due to the subtraction of covariance matrices in Eq. (12). Algorithm 1 summarises the full learning algorithm, and App. G describes how the marginal likelihood, $p(\mathbf{y} \,|\, \boldsymbol{\theta})$, is computed to enable hyperparameter learning.

3.2. Unifying Power EP and Extended Kalman Filtering

In the above inference scheme, a computational saving can be gained by noticing that when $\mathbf{h}(\cdot,\cdot)$ is linear, $\mathcal{L}_k$ can be calculated in closed form. This fact has been exploited previously to aid inference in GP dynamical systems (Deisenroth & Mohamed, 2012). Fig. 2 demonstrates that such an approximation can be accurate when $\mathbf{h}(\cdot,\cdot)$ is locally linear, or when the cavity variance is small. Using a first-order Taylor series expansion about the mean $\boldsymbol{\mu}_k^{\text{cav}}$, we obtain

$$\mathbf{h}(\mathbf{f}_k, \boldsymbol{\sigma}_k) \approx \mathbf{J}_{\mathbf{f}_k}(\mathbf{f}_k - \boldsymbol{\mu}_k^{\text{cav}}) + \mathbf{h}(\boldsymbol{\mu}_k^{\text{cav}}, \mathbf{0}) + \mathbf{J}_{\boldsymbol{\sigma}_k}\boldsymbol{\sigma}_k, \tag{13}$$

a linear function of $\mathbf{f}_k$ and $\boldsymbol{\sigma}_k \sim \mathrm{N}(\mathbf{0}, \boldsymbol{\Sigma}_k)$, such that $p(\mathbf{y}_k \,|\, \mathbf{f}_k) \approx \mathrm{N}(\mathbf{y}_k \,|\, \hat{\mathbf{h}}(\mathbf{f}_k), \mathbf{J}_{\boldsymbol{\sigma}_k}\boldsymbol{\Sigma}_k\mathbf{J}_{\boldsymbol{\sigma}_k}^\top)$, where $\hat{\mathbf{h}}(\mathbf{f}_k) = \mathbf{J}_{\mathbf{f}_k}(\mathbf{f}_k - \boldsymbol{\mu}_k^{\text{cav}}) + \mathbf{h}(\boldsymbol{\mu}_k^{\text{cav}}, \mathbf{0})$. Here $\mathbf{J}_{\mathbf{f}_k} = \mathbf{J}_{\mathbf{f}}|_{\boldsymbol{\mu}_k^{\text{cav}}, \mathbf{0}} \in \mathbb{R}^{d\times m}$ and $\mathbf{J}_{\boldsymbol{\sigma}_k} = \mathbf{J}_{\boldsymbol{\sigma}}|_{\boldsymbol{\mu}_k^{\text{cav}}, \mathbf{0}} \in \mathbb{R}^{d\times d}$ are the Jacobians of $\mathbf{h}(\cdot,\cdot)$ w.r.t. $\mathbf{f}_k$ and $\boldsymbol{\sigma}_k$ evaluated at the mean, respectively.

In order to frame approximate inference in the same setting as EP, we seek the site update rule implied by this linearisation. If $\mathbf{J}_{\mathbf{f}}$ is invertible, then writing down such a rule would be trivial, but since this is not generally the case we instead use the EP moment matching steps, Eq. (10), which give,

$$\begin{aligned}
\mathcal{L}_k &= \log \mathbb{E}_{\mathrm{N}(\mathbf{f}_k \,|\, \boldsymbol{\mu}_k^{\text{cav}}, \boldsymbol{\Sigma}_k^{\text{cav}})}\left[\mathrm{N}^\alpha\left(\mathbf{y}_k \,|\, \hat{\mathbf{h}}(\mathbf{f}_k), \mathbf{J}_{\boldsymbol{\sigma}_k}\boldsymbol{\Sigma}_k\mathbf{J}_{\boldsymbol{\sigma}_k}^\top\right)\right] \\
&= c + \log \mathrm{N}\left(\mathbf{y}_k \,|\, \mathbf{h}(\boldsymbol{\mu}_k^{\text{cav}}, \mathbf{0}), \alpha^{-1}\hat{\boldsymbol{\Sigma}}_k\right),
\end{aligned} \tag{14}$$

where $\hat{\boldsymbol{\Sigma}}_k = \mathbf{J}_{\boldsymbol{\sigma}_k} \boldsymbol{\Sigma}_k \mathbf{J}_{\boldsymbol{\sigma}_k}^\top + \alpha \mathbf{J}_{\mathbf{f}_k} \boldsymbol{\Sigma}_k^{\text{cav}} \mathbf{J}_{\mathbf{f}_k}^\top$. Taking the derivatives of this log-Gaussian w.r.t. the cavity mean, we get

$$\begin{aligned} \boldsymbol{\nabla} \mathcal{L}_k &= \frac{\partial \mathcal{L}_k}{\partial \boldsymbol{\mu}_k^{\text{cav}}} = \alpha \mathbf{J}_{\mathbf{f}_k}^\top \hat{\boldsymbol{\Sigma}}_k^{-1} \mathbf{v}_k, \\ \boldsymbol{\nabla}^2 \mathcal{L}_k &= \frac{\partial^2 \mathcal{L}_k}{\partial \boldsymbol{\mu}_k^{\text{cav}} \partial (\boldsymbol{\mu}_k^{\text{cav}})^\top} = -\alpha \mathbf{J}_{\mathbf{f}_k}^\top \hat{\boldsymbol{\Sigma}}_k^{-1} \mathbf{J}_{\mathbf{f}_k}, \end{aligned} \tag{15}$$

where $\mathbf{v}_k = \mathbf{y}_k - \mathbf{h}(\boldsymbol{\mu}_k^{\text{cav}}, \mathbf{0})$. It is important to note that we have assumed the derivative of $\hat{\boldsymbol{\Sigma}}_k$ to be zero, even though it depends on $\boldsymbol{\mu}_k^{\text{cav}}$. This assumption is crucial in ensuring that the updates are consistent, since it reflects the knowledge that the model is now linear (see Deisenroth & Mohamed (2012) for detailed discussion). Now we update the site in closed form (App. C gives the derivation),

Extended expectation propagation

$$\begin{aligned} \boldsymbol{\Sigma}_k^{\text{site}} &= \left(\mathbf{J}_{\mathbf{f}_k}^\top \left(\mathbf{J}_{\boldsymbol{\sigma}_k} \boldsymbol{\Sigma}_k \mathbf{J}_{\boldsymbol{\sigma}_k}^\top \right)^{-1} \mathbf{J}_{\mathbf{f}_k} \right)^{-1}, \\ \boldsymbol{\mu}_k^{\text{site}} &= \boldsymbol{\mu}_k^{\text{cav}} + \left(\boldsymbol{\Sigma}_k^{\text{site}} + \alpha \boldsymbol{\Sigma}_k^{\text{cav}} \right) \mathbf{J}_{\mathbf{f}_k}^\top \hat{\boldsymbol{\Sigma}}_k^{-1} \mathbf{v}_k. \end{aligned} \tag{16}$$

The result when we use Eq. (16) (with $\alpha = 1$) to modify the filter updates, Eq. (11), is *exactly* the EKF (see App. D for the proof). Additionally, since these updates are now available in closed form, taking the limit $\alpha \to 0$ is now possible and avoids the matrix subtractions and inversions in Eq. (12), which can be costly and unstable. This is not possible prior to linearisation because the intractable integrals also depend on α. App. H describes our full algorithm.

3.3. Power EP and the Unscented/GH Kalman Filters

We now consider the relationship between EP and general Gaussian filters, which use the likelihood approximation

$$\begin{aligned} p(\mathbf{y}_k \,|\, \mathbf{f}_k) \approx \mathrm{N}(\mathbf{y}_k \,|\, \boldsymbol{\mu}_k + \mathbf{C}_k^\top (\boldsymbol{\Sigma}_k^{\text{cav}})^{-1} (\mathbf{f}_k - \boldsymbol{\mu}_k^{\text{cav}}), \\ \mathbf{S}_k - \mathbf{C}_k^\top (\boldsymbol{\Sigma}_k^{\text{cav}})^{-1} \mathbf{C}_k), \end{aligned} \tag{17}$$

where $\boldsymbol{\mu}_k$, $\mathbf{S}_k$ and $\mathbf{C}_k$ are the Kalman mean, innovation and cross-covariance terms respectively, given in App. B. Eq. (17) amounts to *statistical linear regression* (Särkkä, 2013) of $\mathbf{h}(\mathbf{f}_k, \boldsymbol{\sigma}_k)$. Letting $\boldsymbol{\mu}_k^{\text{cav}} = \mathbf{H}_k \mathbf{m}_k^{\text{pred}}$, $\boldsymbol{\Sigma}_k^{\text{cav}} = \mathbf{H}_k \mathbf{P}_k^{\text{pred}} \mathbf{H}_k^\top$ and using the Unscented transform / Gauss–Hermite to approximate $\boldsymbol{\mu}_k$, $\mathbf{S}_k$ and $\mathbf{C}_k$ results in the UKF / GHKF. This approximation has a similar form to the EKF (which uses *analytical* linearisation, see Fig. 2 for comparison), and as in Sec. 3.2 we can insert the Gaussian likelihood approximation into Eq. (10) to derive an iterated algorithm that matches the Gaussian filters on the first forward pass, but then refines the linearisation using EP style updates. This provides the following site update rule (see App. E):

Algorithm 1 Sequential inference & learning algorithm

Input: $\{t_k, \mathbf{y}_k\}_{k=1}^n$, $\boldsymbol{\theta}_0$, α, and learning rate ρ,
`update_rule` $\leftarrow$ Eq. (10), Eq. (16), Eq. (18) or Eq. (20)
for $i = 1$ **to** `num_iters` **do**
  build model, Eq. (7), with $\boldsymbol{\theta}_{i-1}$. $\mathbf{m}_0^{\text{filt}}, \mathbf{P}_0^{\text{filt}} \leftarrow \mathbf{0}, \mathbf{P}_\infty$
  for $k = 1$ **to** n **do**
    $\mathbf{m}_k^{\text{pred}}, \mathbf{P}_k^{\text{pred}} \leftarrow$ `predict` $(\mathbf{m}_{k-1}^{\text{filt}}, \mathbf{P}_{k-1}^{\text{filt}})$
    if $i = 1$ **then**
      initialise $\boldsymbol{\mu}_k^{\text{site}}$, $\boldsymbol{\Sigma}_k^{\text{site}}$ via `update_rule` using $\alpha = 1$ and $\mathbf{m}_k^{\text{pred}}, \mathbf{P}_k^{\text{pred}}$ as cavity / posterior
    end if
    $\mathbf{m}_k^{\text{filt}}, \mathbf{P}_k^{\text{filt}} \leftarrow$ `update`$(\mathbf{m}_k^{\text{pred}}, \mathbf{P}_k^{\text{pred}}, \boldsymbol{\mu}_k^{\text{site}}, \boldsymbol{\Sigma}_k^{\text{site}})$
    $\mathbf{e}_k = -\log p(\mathbf{y}_k \,|\, \mathbf{y}_{1:k-1}, \boldsymbol{\theta}_{i-1})$ see App. G
  end for
  for $k = n - 1$ **to** 1 **do**
    $\mathbf{m}_k^{\text{post}}, \mathbf{P}_k^{\text{post}} \leftarrow$ `smooth`$(\mathbf{m}_{k+1}^{\text{post}}, \mathbf{P}_{k+1}^{\text{post}}, \mathbf{m}_k^{\text{filt}}, \mathbf{P}_k^{\text{filt}})$
    update $\boldsymbol{\mu}_k^{\text{site}}$, $\boldsymbol{\Sigma}_k^{\text{site}}$ via `update_rule`
  end for
  $\boldsymbol{\theta}_i = \boldsymbol{\theta}_{i-1} + \rho \nabla_{\boldsymbol{\theta}} \sum_k \mathbf{e}_k$ update hyper.
end for
Return: posterior mean and covariance: $\mathbf{m}^{\text{post}}$, $\mathbf{P}^{\text{post}}$

Statistically linearised expectation propagation

$$\begin{aligned} \boldsymbol{\Sigma}_k^{\text{site}} &= -\alpha \boldsymbol{\Sigma}_k^{\text{cav}} + \left(\boldsymbol{\Omega}_k^\top \tilde{\boldsymbol{\Sigma}}_k^{-1} \boldsymbol{\Omega}_k \right)^{-1}, \\ \boldsymbol{\mu}_k^{\text{site}} &= \boldsymbol{\mu}_k^{\text{cav}} + \left(\boldsymbol{\Omega}_k^\top \tilde{\boldsymbol{\Sigma}}_k^{-1} \boldsymbol{\Omega}_k \right)^{-1} \boldsymbol{\Omega}_k^\top \tilde{\boldsymbol{\Sigma}}_k^{-1} \mathbf{v}_k. \end{aligned} \tag{18}$$

where $\mathbf{v}_k = \mathbf{y}_k - \boldsymbol{\mu}_k$, $\tilde{\boldsymbol{\Sigma}}_k = \mathbf{S}_k + (\alpha - 1) \mathbf{C}_k^\top (\boldsymbol{\Sigma}_k^{\text{cav}})^{-1} \mathbf{C}_k$,

$$\begin{aligned} \boldsymbol{\Omega}_k = \frac{\partial \boldsymbol{\mu}_k}{\partial \boldsymbol{\mu}_k^{\text{cav}}} = \iint \mathbf{h}(\mathbf{f}_k, \boldsymbol{\sigma}_k) (\boldsymbol{\Sigma}_k^{\text{cav}})^{-1} (\mathbf{f}_k - \boldsymbol{\mu}_k^{\text{cav}}) \\ \times \mathrm{N}(\mathbf{f}_k \,|\, \boldsymbol{\mu}_k^{\text{cav}}, \boldsymbol{\Sigma}_k^{\text{cav}}) \mathrm{N}(\boldsymbol{\sigma}_k \,|\, \mathbf{0}, \boldsymbol{\Sigma}_k) \, \mathrm{d}\mathbf{f}_k \, \mathrm{d}\boldsymbol{\sigma}_k. \end{aligned} \tag{19}$$

3.4. Nonlinear Kalman Smoothers

Iterated versions of nonlinear filter-smoothers have been developed to address the fact that the forward prediction, $\mathrm{N}(\mathbf{x}_k^{\text{pred}} \,|\, \mathbf{m}_k^{\text{pred}}, \mathbf{P}_k^{\text{pred}})$, may not be the optimal distribution about which to perform linearisation. It is argued that the posterior, $\mathrm{N}(\mathbf{x}_k^{\text{post}} \,|\, \mathbf{m}_k^{\text{post}}, \mathbf{P}_k^{\text{post}})$, obtained via smoothing, provides a better estimate of the region in which the likelihood affects the posterior (García-Fernández et al., 2015).

These iterated smoothers (Bell, 1994) can be seen as special cases of the algorithms described in Sec. 3.2 and Sec. 3.3, where the posterior is used to perform the linearisation in place of the cavity, *i.e.* $\alpha = 0$. The classical smoothers seek a linear approximation to the likelihood $p(\mathbf{y}_k \,|\, \mathbf{f}_k) \approx \mathrm{N}(\mathbf{y}_k \,|\, \mathbf{B}_k \mathbf{f}_k + \mathbf{b}_k, \mathbf{E}_k)$ via a Taylor expansion, Eq. (13), or SLR, Eq. (17), and then store parameters $\mathbf{B}_k$, $\mathbf{b}_k$, $\mathbf{E}_k$ to be used during the next forward pass. Instead, we

use the current posterior approximation to compute the site parameters via Eq. (16) or Eq. (18), which differs slightly from the standard presentation of these algorithms. We argue that framing the Kalman smoothers as site update rules is beneficial in that it allows for direct comparison with EP, but also that introduction of the cavity is beneficial. The cavity may be a better distribution about which to linearise than the posterior, since it does not already include the effect of the local data. However, Table 1 shows that setting $\alpha = 0$ typically provides the best performance.

3.5. Variational Inference with Natural Gradients

Variational inference (VI) is an alternative to EP, often favoured due to its convergence guarantees and ease of implementation. If VI is formulated such that the variational parameters of the approximate posterior $q(\mathbf{f})$ are the likelihood (*i.e.* site) mean and covariance, as in Eq. (5), then it can also be framed as a site update rule during Kalman smoothing (Chang et al., 2020). This parametrisation is in fact the optimal one, as discussed in Opper & Archambeau (2009), but is often avoided because the resulting optimisation problem is non-convex (instead it is common to declare a variational distribution over the full posterior, $q(\mathbf{f}) = \mathrm{N}(\mathbf{m}, \mathbf{K})$, and optimise $\mathbf{m}$, $\mathbf{K}$ with respect to the evidence lower bound. Adam et al. (2020) show how to perform *natural gradient* VI under this parametrisation).

We present here the VI site update rule, based on conjugate-computation variational inference (CVI, Khan & Lin, 2017), in order to show their similarity to EP, and to enable direct comparison between the algorithms. CVI sidesteps the issues with the optimal parametrisation by showing that natural gradient VI can be performed via local site parameter updates that avoid directly differentiating the evidence lower bound. The updates can be written,

Variational inference (with natural gradients)

$$\begin{aligned}
\tilde{\mathcal{L}}_k &= \mathbb{E}_{\mathrm{N}(\mathbf{f}_k \,|\, \boldsymbol{\mu}_k^{\text{post}}, \boldsymbol{\Sigma}_k^{\text{post}})}\big[\log p(\mathbf{y}_k \,|\, \mathbf{f}_k)\big], \\
\boldsymbol{\Sigma}_k^{\text{site}} &= -\left(\boldsymbol{\nabla}^2 \tilde{\mathcal{L}}_k\right)^{-1}, \\
\boldsymbol{\mu}_k^{\text{site}} &= \boldsymbol{\mu}_k^{\text{post}} - \left(\boldsymbol{\nabla}^2 \tilde{\mathcal{L}}_k\right)^{-1} \boldsymbol{\nabla} \tilde{\mathcal{L}}_k,
\end{aligned} \tag{20}$$

where $\boldsymbol{\nabla} \tilde{\mathcal{L}}_k \in \mathbb{R}^m$ and $\boldsymbol{\nabla}^2 \tilde{\mathcal{L}}_k \in \mathbb{R}^{m \times m}$ are the Jacobian and Hessian of $\tilde{\mathcal{L}}_k$ w.r.t. $\boldsymbol{\mu}_k^{\text{post}}$ respectively.

3.6. Spatio-Temporal Filtering and Smoothing

The methodology presented in the previous sections for temporal models directly lends itself to generalisations in spatio-temporal modelling. We consider a GP prior which is separable in the sequential (temporal) input t and the remaining (spatial) input(s) $\mathbf{r}$: $\kappa(\mathbf{r}, t; \mathbf{r}', t') = \kappa_{\mathrm{r}}(\mathbf{r}, \mathbf{r}')\, \kappa_{\mathrm{t}}(t, t')$.

Following Särkkä et al. (2013), we extend the state $\mathbf{x}(t)$ of the system via m coupled temporal processes. These processes are associated with inducing points $\{\mathbf{r}_{\mathrm{u},j}\}_{j=1}^m$ in the spatial domain. The measurement model matrix now projects the latent state at time t_k from the inducing processes in the state to function space by,

$$\mathbf{H}_k = [\mathbf{K}_{\mathrm{f}_k,\mathrm{u}}\, \mathbf{K}_{\mathrm{u,u}}^{-1}] \otimes \mathbf{H}_{\mathrm{t}}, \tag{21}$$

with Gram matrices $\mathbf{K}_{\mathrm{f}_k,\mathrm{u}} = \kappa_{\mathrm{r}}(\mathbf{r}_k, \mathbf{r}_{\mathrm{u},j})$ and $\mathbf{K}_{\mathrm{u,u}} = \kappa_{\mathrm{r}}(\mathbf{r}_{\mathrm{u},j}, \mathbf{r}_{\mathrm{u},j})$ for $j = 1, 2, \ldots, m$, where $\mathbf{H}_{\mathrm{t}}$ is the measurement model matrix for the GP prior. If the data lies on a fixed set of spatial points $\{\mathbf{r}_j\}_{j=1}^m$ (an irregular grid of m points), the above expression simplifies to $\mathbf{H}_k = \mathbf{I}_m \otimes \mathbf{H}_t$ and the models becomes exact (see, Hartikainen, 2013; Solin, 2016, for further details and discussion, also covering non-separable models).

3.7. Fast Implementation Using JAX

Sequential inference in GPs is extremely efficient, however optimising the model hyperparameters involves differentiating functions with large loops. When using automatic differentiation this typically results in a massive computational graph with large compilation overheads, memory usage and runtime. Previous approaches have avoided this issue either by using finite differences (Nickisch et al., 2018), which are slow when the number of parameters is large, or by reformulating the model to exploit linear algebra methods applicable to sparse precision matrices (Durrande et al., 2019).

We utilise the following features of the differential programming Python framework, JAX (Bradbury et al., 2018): *(i)* we avoid 'unrolling' of for-loops, *i.e.* instead of building a large graph of repeated operations, a smaller graph is recursively called, reducing the compilation overhead and memory, *(ii)* we just-in-time (JIT) compile the loops, to avoid the cost of graph retracing, *(iii)* we use accelerated linear algebra (XLA) to speed up the underlying filtering/smoothing operations. Combined, these features result in an extremely fast implementation, and based on this we provide a fully featured temporal GP framework with all models and inference methods, available at https://github.com/AaltoML/kalman-jax.

4. Empirical Analysis

Table 1 examines the performance of 17 methods from our GP framework on 7 benchmark tasks of varying data size and model complexity. Blanks (—) in the table represent scenarios where the method does not scale practically to the size of the task. First we demonstrate that state space approximate inference schemes are competitive with two state-of-the-art baseline methods on three small data sets.

Table 1. Normalised negative log predictive density (NLPD) results with 10-fold cross-validation. Smaller is better. Blank entries (—) represent scenarios where the method does not scale to the size of the task. EEP, UEP, and GHEP are the iterated smoothers with linearisation. EP(U) and EP(GH) are state space EP, where the intractable moment matching is performed via the Unscented transform or Gauss–Hermite, respectively. Linearisation performs poorly on the heteroscedastic noise task, however EEP performs well on the audio task since it is the only method capable of maintaining full site cross-covariance terms without compromising stability. The state space methods are able to match the performance of the non-sequential (batch) EP and VGP baselines.

		MOTORCYCLE	COAL	BANANA	BINARY	AUDIO	AIRLINE	RAINFOREST
	# DATA POINTS	133	333	400	10K	22K	36K	125K
	INPUT DIMENSION	1	1	2	1	1	1	2
	LIKELIHOOD	HETEROSCEDASTIC	POISSON	BERNOULLI	BERNOULLI	PRODUCT	POISSON	POISSON
LINEARISATION	EEP ($\alpha = 1$)	0.855±0.25	0.922±0.11	0.228±0.07	0.536±0.01	−0.433±0.04	0.142±0.01	0.325±0.01
	EEP ($\alpha = 0.5$)	0.855±0.25	0.922±0.11	0.228±0.07	0.536±0.01	−0.499±0.03	0.142±0.01	0.321±0.01
	EEP ($\alpha = 0$) / EKS	0.855±0.25	0.922±0.11	0.229±0.07	0.537±0.01	**−0.570±0.04**	0.142±0.01	**0.309±0.01**
	UEP ($\alpha = 1$)	0.745±0.28	0.922±0.11	0.217±0.08	0.536±0.01	−0.471±0.02	0.142±0.01	—
	UEP ($\alpha = 0.5$)	0.745±0.28	0.922±0.11	0.217±0.08	0.536±0.01	−0.474±0.02	0.142±0.01	—
	UEP ($\alpha = 0$) / UKS	0.745±0.28	0.922±0.11	0.217±0.08	0.536±0.01	−0.484±0.02	0.142±0.01	—
	GHEP ($\alpha = 1$)	0.750±0.26	0.922±0.11	0.217±0.08	0.536±0.01	—	0.142±0.01	—
	GHEP ($\alpha = 0.5$)	0.747±0.27	0.922±0.11	0.217±0.08	0.536±0.01	—	0.142±0.01	—
	GHEP($\alpha = 0$)/GHKS	0.746±0.27	0.922±0.11	0.217±0.08	0.536±0.01	—	0.142±0.01	—
MOMENT MATCH	EP(U) ($\alpha = 1$)	0.696±0.59	0.922±0.11	0.217±0.08	0.536±0.01	−0.321±0.15	0.143±0.01	—
	EP(U) ($\alpha = 0.5$)	0.479±0.30	0.922±0.11	0.217±0.08	0.536±0.01	−0.327±0.20	0.143±0.01	—
	EP(U) ($\alpha \approx 0$)	0.465±0.29	0.924±0.11	0.222±0.08	0.536±0.01	0.011±0.30	0.143±0.01	—
	EP(GH) ($\alpha = 1$)	0.569±0.41	0.922±0.11	0.217±0.08	0.536±0.01	—	0.142±0.01	—
	EP(GH) ($\alpha = 0.5$)	0.531±0.38	0.922±0.11	0.217±0.08	0.536±0.01	—	0.142±0.01	—
	EP(GH) ($\alpha \approx 0$)	**0.444±0.32**	0.922±0.11	0.217±0.08	0.536±0.01	—	0.142±0.01	—
VI	VI(U)	**0.444±0.31**	0.922±0.11	0.217±0.08	0.536±0.01	−0.204±0.02	0.142±0.01	—
	VI(GH)	0.495±0.34	0.922±0.11	0.217±0.08	0.536±0.01	—	0.142±0.01	—
BASEL.	EP(BATCH, GH)	0.441±0.30	0.922±0.11	0.216±0.10	—	—	—	—
	VGP(BATCH, GH)	0.444±0.30	0.922±0.11	0.219±0.09	—	—	—	—

We compare against batch EP (see Sec. 2.3) and a variational GP (VGP, Opper & Archambeau, 2009, with order $n + n^2$ parameters to ensure convexity of the objective, as in GPflow, Matthews et al., 2017). We then compare our methods on four large data tasks to which the baselines are not applicable. Note that sparse variants of EP and VGP are not suited to these long time series containing high-frequency behaviour (*e.g.*, Fig. 3c) that cannot be summarised by a few thousand inducing points (see Sec. 1 for discussion).

We evaluate all methods via negative log predictive density (NLPD) using 10-fold cross-validation, with each method run for 250 iterations (baselines are run until convergence). Gauss–Hermite integration uses 20^q cubature points, whereas the Unscented transform uses $2q^2 + 1$ (we use the symmetric 5$^{\text{th}}$ order cubature rule, *i.e.* UT5, Kokkala et al., 2016; McNamee & Stenger, 1967), where q is the dimensionality of the integral (typically the number of GPs that are nonlinearly combined in the likelihood). For standard EP, where a power of zero is not possible, we set $\alpha = 0.01$. We optimise the model hyperparameters by maximising the marginal likelihood $p(\mathbf{y} \mid \boldsymbol{\theta})$ separately for each method (see App. G), hence the results in Table 1 are affected by both training and inference, demonstrating their practical applicability. The baseline methods scale as $\mathcal{O}(n^3)$, while all the sequential schemes scale as $\mathcal{O}(ns^3)$.

Results Table 1 confirms it is possible to achieve state-of-the-art performance with sequential inference. The log-Gaussian Cox process and classification experiments return consistent results across all methods. However, Audio and Rainforest involve multidimensional sites, making them difficult tasks. In such cases, EEP performs well because it is the only method capable of maintaining full site covariance terms whilst remaining stable. Statistical linearisation suffers less from a reduction of cubature points than EP moment matching or VI updates, as shown by the performance of UEP on Audio. EP generally performed well, but EEP matches its performance sometimes whilst being the only method applicable to the Rainforest task. In other cases, cubature methods outperform linearisation, particularly on the Motorcycle task.

Motorcycle (heteroscedastic noise) The motorcycle crash data set (Silverman, 1985) contains 131 non-uniformly spaced measurements from an accelerometer placed on a motorcycle helmet during impact, over a period of 60 ms. It is a challenging benchmark (*e.g.*, Tolvanen et al., 2014), due

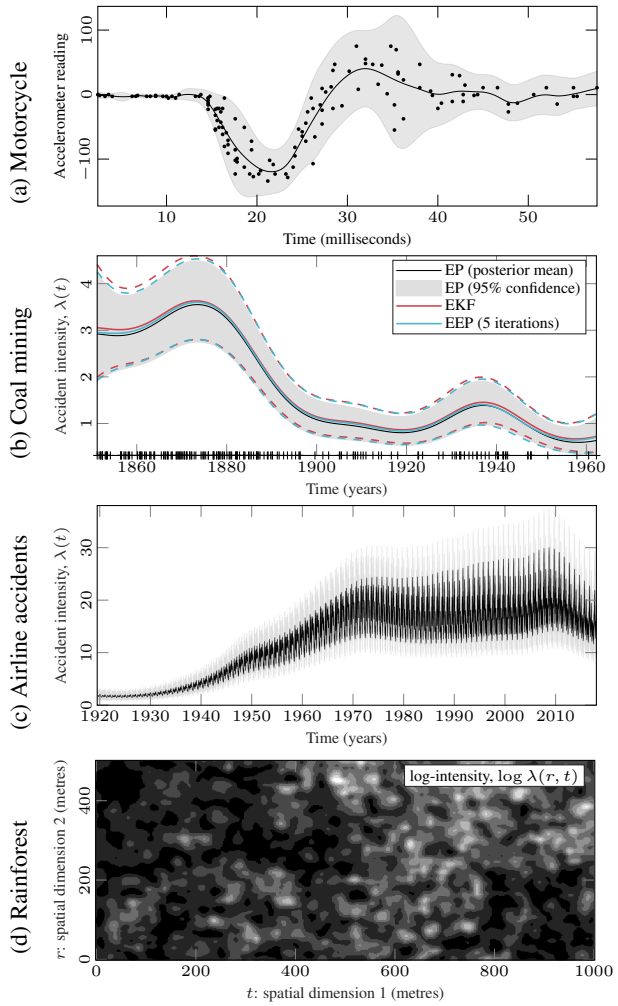


Figure 3. Examples of non-conjugate GP models. (a) In the motorcycle task (heteroscedastic noise), EP is capable of modelling the time-varying noise component. The log-Gaussian Cox processes (b)–(d) are well approximated via linearisation, and iterating improves the match to the EP posterior.

to the heteroscedastic noise variance. We model both the process itself and the measurement noise scale with independent GP priors with Matérn-$3/2$ kernels: $y_k \mid f_k^{(1)}, f_k^{(2)} \sim \mathrm{N}(y_k \mid f^{(1)}(t_k), [\phi(f^{(2)}(t_k))]^2)$, with softplus link function $\phi(f) = \log(1 + e^f)$ to ensure positive noise scale. The full EP and VGP baselines were implemented and hand-tailored for this task. For VGP, we used GPflow 2 (Matthews et al., 2017) with a custom model. VI and EP ($\alpha \approx 0$) performed well, however the linearisation-based methods failed to capture the time-varying noise (see App. I for discussion).

Coal (log-Gaussian Cox process) The coal mining disaster data set (Vanhatalo et al., 2013) contains 191 explosions that killed ten or more men in Britain between 1851–1962. We use a log-Gaussian Cox process, *i.e.* an inhomogeneous Poisson process (approximated with a Poisson likelihood for $n = 333$ equal time interval bins).

Table 2. Run times in seconds (mean across 10 runs). We report time to evaluate the marginal likelihood on the forward pass *and* perform the site updates on the smoothing pass.

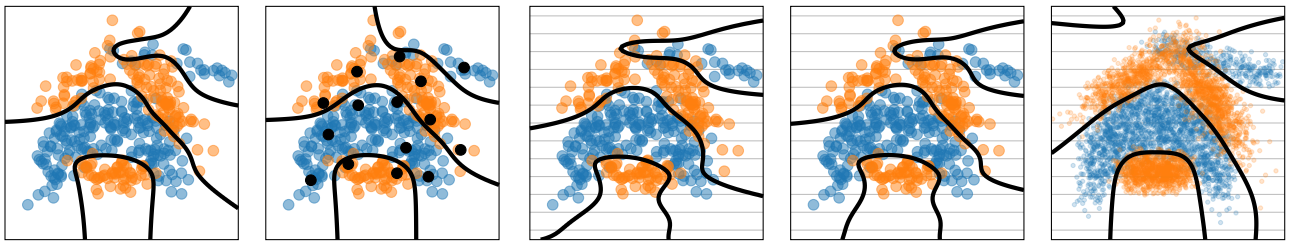
	MOTORCYCLE (Heteroscedastic)	COAL (Poisson)	BANANA (Bernoulli)	BINARY (Bernoulli)	AUDIO (Product)	AIRLINE (Poisson)	RAINFOREST (Poisson)
EEP	0.015	0.013	0.135	0.100	1.176	23.941	37.441
UEP	0.017	0.015	0.144	0.113	1.661	24.583	—
GHEP	0.020	0.016	0.146	0.120	—	23.709	—
EP(U)	0.018	0.015	0.143	0.108	1.713	23.492	—
EP(GH)	0.019	0.016	0.150	0.127	—	23.777	—
VI(U)	0.017	0.017	0.142	0.098	1.619	23.796	—
VI(GH)	0.018	0.016	0.145	0.123	—	23.611	—
Time steps	133	333	400	10000	22050	35959	500
State dim.	6	3	45	4	15	59	500

We use a Matérn-$5/2$ GP prior with likelihood $p(\mathbf{y} \mid \mathbf{f}) \approx \prod_{k=1}^{n} \mathrm{Poisson}(y_k \mid \exp(f(\hat{t}_k)))$, where $\hat{t}_k$ is the bin coordinate and y_k the number of disasters in the bin. This model reaches posterior consistency in the limit of bin width going to zero (Tokdar & Ghosh, 2007). Since linearisation requires a continuous likelihood, we approximate the discrete Poisson with a Gaussian by noticing that its first two moments are equal to the intensity $\lambda(t) = \exp(f(t))$, giving $y_k \mid f_k \overset{\text{approx.}}{\sim} \mathrm{N}(y_k \mid \lambda(\hat{t}_k), \lambda(\hat{t}_k))$. See App. I for details.

Airline (log-Gaussian Cox process) The airline accidents data (Nickisch et al., 2018) consists of 1210 dates of commercial airline accidents between 1919–2017. We use a log-Gaussian Cox process with bin width of one day, leading to $n = 35{,}959$ observations. The prior has multiple components, $\kappa(t,t') = \kappa(t,t')^{\nu=5/2}_{\text{Mat.}} + \kappa(t,t')^{1\,\text{year}}_{\text{per.}}\kappa(t,t')^{\nu=3/2}_{\text{Mat.}} + \kappa(t,t')^{1\,\text{week}}_{\text{per.}}\kappa(t,t')^{\nu=3/2}_{\text{Mat.}}$, capturing a long-term trend, time-of-year variation (with decay), and day-of-week variation (with decay). The state dimension is $s = 59$.

Binary (1D classification) As a 1D classification task, we create a long binary time series, $n = 10{,}000$, using the generating function $y(t) = \mathrm{sign}\{\frac{12\sin(4\pi t)}{0.25\pi t + 1} + \sigma_t\}$, with $\sigma_t \sim \mathrm{N}(0, 0.25^2)$. Our GP prior has a Matérn-$7/2$ kernel, $s = 4$, and the sigmoid function $\psi(f) = (1 + e^{-f})^{-1}$ maps $\mathbb{R} \mapsto [0, 1]$ (logit classification). See App. I for derivation of approximate continuous state observation model, $h(f_k, \sigma_k) = \psi(f_k) + \sqrt{\psi(f_k)(1 - \psi(f_k))}\sigma_k$.

Audio (product of GPs) We apply a simplified version of the Gaussian Time-Frequency model from Wilkinson et al. (2019) to half a second of human speech, sampled at 44.1 kHz, $n = 22{,}050$. The prior consists of 3 quasi-periodic ($\kappa_{\text{exp}}(t,t')\kappa_{\cos}(t,t')$) 'subband' GPs, and 3 smooth ($\kappa_{\text{Mat-}5/2}(t,t')$) 'amplitude' GPs. The likelihood consists of a sum of the product of these processes with additive noise and a softplus mapping $\phi(\cdot)$ for the positive amplitudes: $y_k \mid \mathbf{f}_k \sim \mathrm{N}(\sum_{i=1}^{3} f_{i,k}^{\text{sub.}}\phi(f_{i,k}^{\text{amp.}}), \sigma_k^2)$. The nonlinear inter-



(a) VGP (Full) (b) VGP (Sparse) (c) EKF (d) EEP (e) EEP

Figure 4. Inference schemes on the two-dimensional *Banana* classification task. The coloured points represent training data and the black lines are decision boundaries. (a) is the baseline variational GP method (VGP in Table 1). (b) shows the sparse variant of the VGP baseline (Generalized FITC) with 15 inducing points (black dots). In (c)–(e), the vertical dimension is treated as the 'spatial' input ($m = 15$ inducing points shown with lines) and the horizontal as the sequential ('temporal') dimension. Our formulation using the EKF (c) works well, but is further improved by the EP-like iteration in (d). In (e), the method is applied to the full data set ($n = 5400$).

action of 6 GPs ($s = 15$) in the likelihood makes this a challenging task. EEP performs best since it is capable of maintaining full site covariance terms without compromising stability. UEP outperforms EP and VI since statistical linearisation is still accurate when using few cubature points.

4.1. Spatio-Temporal Models

As presented in Sec. 3.6, the sequential inference schemes are also applicable to spatio-temporal problems. We illustrate this via two spatial problems, treating one spatial input as the sequential dimension ('time') and the other as 'space'.

Banana (2D classification) The banana data set, $n = 400$, is a common 2D classification benchmark (*e.g.*, Hensman et al., 2015). We use the logit likelihood with a separable space-time kernel: $\kappa(r, t; r', t') = \kappa(t, t')^{\nu=5/2}_{\text{Mat.}} \kappa(r, r')^{\nu=5/2}_{\text{Mat.}}$. The vertical dimension is treated as space r and the horizontal as the sequential ('temporal') dimension t. We use $m = 15$ inducing points in r (see Sec. 3.6), visualised by lines in Fig. 4(c)–(e). The state dimension is $s = 3m = 45$. Fig. 4 shows that the EKF provides a similar solution to the VGP baseline of Hensman et al. (2015), and an even closer match is obtained by EEP (3 iterations). The forward and backward passes are visualised in Fig. 1. The method is also applicable to the larger ($n = 5400$) version of the data set (Fig. 4e).

Rainforest (2D log-Gaussian Cox process) We study the density of a single tree species, *Trichilia tuberculata*, from a 1000 m $\times$ 500 m region of a rainforest in Panama (Condit, 1998; Hubbell et al., 1999; 2005). We segment the space into 4 m^2 bins, giving a 500×250 grid with 125,000 data points ($n = 500$ time steps), and use a log-Gaussian Cox process (Fig. 3d). The space-time GP has a separable Matérn-$3/2$ kernel. We *do not* use a sparse approximation in r, instead we have $m = 250$ temporal processes, so $s = 2m = 500$.

Run Times Table 2 compares time taken for all methods to make a single training step on a MacBook Pro with 2.3 GHz Intel Core i5 and 16 GB RAM using JAX. For tasks with one-dimensional sites all methods are similar, however EEP is faster than the cubature methods for the Audio task which involves 6-dimensional sites. The gridded data of the Rainforest task requires a 250-dimensional site parameter update which is impractical for most methods. Conversely, in the Banana task data points are handled one by one, such that only one-dimensional updates are required.

5. Discussion and Conclusions

We argue that development of methods capable of naturally handling sequential data is crucial to extend the applicability of GPs beyond short time series. EP was originally inspired by, and derived from, Kalman filtering and here we make the case that a return to sequential methods is desirable for large spatio-temporal problems. We present a flexible and efficient framework for sequential learning that encompasses many state-of-the-art approximate inference schemes, whilst also illuminating the connections between modern day inference methods and traditional filtering approaches.

Our theoretical contributions confirm that using linearisation in place of EP moment matching results in iterated algorithms that exactly match the classical nonlinear Kalman filters on the first pass, and also generalise the classical smoothers by refining the linearisations via multiple passes through the data. These algorithms are fast and scale to high-dimensional spatio-temporal problems more effectively than EP and VI. The methods based on Taylor series approximations only require one evaluation of the likelihood (and its Jacobian) for each data point, as opposed to cubature methods, and these algorithms make it particularly straightforward to prototype and implement new likelihood models.

We provide a detailed examination of the different properties of all these methods on five time series and two spatial tasks, showing that the state space framework for GPs is applicable beyond one-dimensional problems. We have also highlighted the scenarios in which such methods might fail: linearisation is a poor approximation when the cavities are diffuse (high variance) and the likelihood is highly nonlinear, but cubature methods do not scale well to high dimensions.

Acknowledgements

We acknowledge funding from the Academy of Finland (grant numbers 308640 and 324345) and from Innovation Fund Denmark (grant number 8057-00036A). Our results were obtained using computational resources provided by the Aalto Science-IT project. Thanks to S. T. John for help implementing the VGP baseline for the heteroscedastic noise model.

References

Adam, V., Eleftheriadis, S., Artemev, A., Durrande, N., and Hensman, J. Doubly sparse variational gaussian processes. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2874–2884. PMLR, 2020.

Bar-Shalom, Y., Li, X.-R., and Kirubarajan, T. *Estimation with Applications to Tracking and Navigation*. Wiley-Interscience, 2001.

Bell, B. M. The iterated Kalman smoother as a Gauss–Newton method. *SIAM Journal on Optimization*, 4(3): 626–636, 1994.

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., and Wanderman-Milne, S. JAX: Composable transformations of Python+NumPy programs, 2018. http://github.com/google/jax.

Bui, T. D., Yan, J., and Turner, R. E. A unifying framework for Gaussian process pseudo-point approximations using power expectation propagation. *Journal of Machine Learning Research (JMLR)*, 18(1):3649–3720, 2017.

Chang, P. E., Wilkinson, W. J., Khan, M. E., and Solin, A. Fast variational learning in state-space Gaussian process models. In *International Workshop on Machine Learning for Signal Processing (MLSP)*, 2020.

Condit, R. *Tropical Forest Census Plots*. Springer-Verlag and R. G. Landes Company, Berlin, Germany, and Georgetown, Texas, 1998.

Deisenroth, M. and Mohamed, S. Expectation propagation in Gaussian process dynamical systems. In *Advances in Neural Information Processing Systems (NIPS)*, volume 25, pp. 2609–2617. Curran Associates, Inc., 2012.

Durrande, N., Adam, V., Bordeaux, L., Eleftheriadis, S., and Hensman, J. Banded matrix operators for Gaussian Markov models in the automatic differentiation era. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 89 of *Proceedings of Machine Learning Research*, pp. 2780–2789. PMLR, 2019.

García-Fernández, Á. F., Svensson, L., Morelande, M. R., and Särkkä, S. Posterior linearization filter: Principles and implementation using sigma points. *IEEE Transactions on Signal Processing*, 63(20):5561–5573, 2015.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. *Bayesian Data Analysis*. Chapman & Hall/CRC Press, third edition, 2013.

Hartikainen, J. *Sequential Inference for Latent Temporal Gaussian Process Models*. Doctoral dissertation, Aalto University, Helsinki, Finland, 2013.

Hartikainen, J. and Särkkä, S. Kalman filtering and smoothing solutions to temporal Gaussian process regression models. In *International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2010.

Hensman, J., Fusi, N., and Lawrence, N. D. Gaussian processes for big data. In *Proceedings of the 29th Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 282–290. AUAI Press, 2013.

Hensman, J., Matthews, A., and Ghahramani, Z. Scalable variational Gaussian process classification. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 38 of *Proceedings of Machine Learning Research*, pp. 351–360. PMLR, 2015.

Hubbell, S., Foster, R., O'Brien, S., Harms, K., Condit, R., Wechsler, B., Wright, S., and Loo de Lao, S. Light-gap disturbances, recruitment limitation, and tree diversity in a neotropical forest. *Science*, 283:554–557, 1999.

Hubbell, S., Condit, R., and Foster, R. Barro Colorado forest census plot data. URL: http://ctfs.si.edu/webatlas/datasets/bci/, 2005.

Ito, K. and Xiong, K. Gaussian filters for nonlinear filtering problems. *IEEE Transactions on Automatic Control*, 45 (5):910–927, 2000.

Julier, S. J., Uhlmann, J. K., and Durrant-Whyte, H. F. A new approach for filtering nonlinear systems. In *Proceedings of the American Control Conference*, volume 3, pp. 1628–1632, 1995.

Julier, S. J., Uhlmann, J. K., and Durrant-Whyte, H. F. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Transactions on Automatic Control*, 45(3):477–482, 2000.

Jylänki, P., Vanhatalo, J., and Vehtari, A. Robust Gaussian process regression with a Student-t likelihood. *Journal of Machine Learning Research (JMLR)*, 12:3227–3257, 2011.

Khan, M. and Lin, W. Conjugate-computation variational inference: Converting variational inference in non-conjugate models to inferences in conjugate models. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 878–887. PMLR, 2017.

Kokkala, J., Solin, A., and Särkkä, S. Sigma-point filtering and smoothing based parameter estimation in nonlinear dynamic systems. *Journal of Advances in Information Fusion*, 11(1):15–30, 2016.

Kuss, M. and Rasmussen, C. E. Assessing approximations for Gaussian process classification. In *Advances in Neural Information Processing Systems (NIPS)*, volume 18, pp. 699–706. MIT Press, 2006.

Matthews, A. G. d. G., van der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrá, P., Ghahramani, Z., and Hensman, J. GPflow: A Gaussian process library using TensorFlow. *Journal of Machine Learning Research (JMLR)*, 18(40):1–6, 2017.

McNamee, J. and Stenger, F. Construction of fully symmetric numerical integration formulas. *Numerische Mathematik*, 10(4):327–344, 1967.

Minka, T. Divergence measures and message passing. Technical report, 2005.

Minka, T. P. Expectation propagation for approximate Bayesian inference. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 17, pp. 362–369. AUAI Press, 2001.

Minka, T. P. Power EP. Technical report, 2004.

Nickisch, H., Solin, A., and Grigorievskiy, A. State space Gaussian processes with non-Gaussian likelihood. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3789–3798. PMLR, 2018.

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

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

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

Rauch, H. E., Tung, F., and Striebel, C. T. Maximum likelihood estimates of linear dynamic systems. *AIAA Journal*, 3(8):1445–1450, 1965.

Reece, S. and Roberts, S. An introduction to Gaussian processes for the Kalman filter expert. In *Proceedings of the 13th Conference on Information Fusion (FUSION)*, pp. 1–9. IEEE, 2010.

Salimbeni, H. and Deisenroth, M. Doubly stochastic variational inference for deep Gaussian processes. In *Advances in Neural Information Processing Systems (NIPS)*, volume 32, pp. 4588–4599. Curran Associates, Inc., 2017.

Särkkä, S. *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.

Särkkä, S. and Solin, A. *Applied Stochastic Differential Equations*. Cambridge University Press, 2019.

Särkkä, S., Solin, A., and Hartikainen, J. Spatiotemporal learning via infinite-dimensional Bayesian filtering and smoothing. *IEEE Signal Processing Magazine*, 30(4): 51–61, 2013.

Seeger, M. Expectation propagation for exponential families. Technical report, 2005.

Silverman, B. W. Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society: Series B (Methodological)*, 47(1):1–21, 1985.

Šimandl, M. and Duník, J. Derivative-free estimation methods: New results and performance analysis. *Automatica*, 45(7):1749–1757, 2009.

Solin, A. *Stochastic Differential Equation Methods for Spatio-Temporal Gaussian Process Regression*. Doctoral dissertation, Aalto University, Helsinki, Finland, 2016.

Titsias, M. K. Variational learning of inducing variables in sparse Gaussian processes. In *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 5 of *Proceedings of Machine Learning Research*, pp. 567–574. PMLR, 2009.

Tobar, F. Band-limited gaussian processes: The sinc kernel. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 12728–12738. Curran Associates, Inc., 2019.

Tokdar, S. T. and Ghosh, J. K. Posterior consistency of logistic Gaussian process priors in density estimation. *Journal of Statistical Planning and Inference*, 137(1): 34–42, 2007.

Tolvanen, V., Jylänki, P., and Vehtari, A. Expectation propagation for nonstationary heteroscedastic Gaussian process regression. In *International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6. IEEE, 2014.

Vanhatalo, J., Riihimäki, J., Hartikainen, J., Jylänki, P., Tolvanen, V., and Vehtari, A. GPstuff: Bayesian modeling with Gaussian processes. *Journal of Machine Learning Research (JMLR)*, 14(Apr):1175–1179, 2013.

Wainwright, M. J. and Jordan, M. I. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine. Learning*, 1(1-2):1–305, 2008.

Wang, K., Pleiss, G., Gardner, J., Tyree, S., Weinberger, K. Q., and Wilson, A. G. Exact Gaussian processes on a million data points. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pp. 14622–14632. Curran Associates, Inc., 2019.

Wilkinson, W. J., Andersen, M. R., Reiss, J. D., Stowell, D., and Solin, A. End-to-end probabilistic inference for nonstationary audio analysis. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pp. 6776–6785. PMLR, 2019.

Wu, Y., Hu, D., Wu, M., and Hu, X. Unscented Kalman filtering for additive noise case: Augmented vs. non-augmented. In *Proceedings of the American Control Conference*, volume 6, pp. 4051–4055. IEEE, 2005.

Wu, Y., Hu, D., Wu, M., and Hu, X. A numerical-integration perspective on Gaussian filters. *IEEE Transactions on Signal Processing*, 54(8):2910–2921, 2006.