# Gaussian Process Product Models for Nonparametric Nonstationarity

**Ryan Prescott Adams** RPA23@CAM.AC.UK
**Oliver Stegle** OS252@CAM.AC.UK
Cavendish Laboratory, University of Cambridge, Cambridge CB3 0HE, UK

## Abstract

Stationarity is often an unrealistic prior assumption for Gaussian process regression. One solution is to predefine an explicit nonstationary covariance function, but such covariance functions can be difficult to specify and require detailed prior knowledge of the nonstationarity. We propose the Gaussian process product model (GPPM) which models data as the pointwise product of two latent Gaussian processes to nonparametrically infer nonstationary variations of amplitude. This approach differs from other nonparametric approaches to covariance function inference in that it operates on the outputs rather than the inputs, resulting in a significant reduction in computational cost and required data for inference. We present an approximate inference scheme using Expectation Propagation. This variational approximation yields convenient GP hyperparameter selection and compact approximate predictive distributions.

## 1. Introduction

The Gaussian process (Rasmussen & Williams, 2006) is a useful and popular prior for nonlinear regression. It can be used to construct a distribution over scalar functions via a prior on smoothness. This prior is specified through a positive-definite kernel, which determines the covariance between two outputs as a function of their corresponding inputs. Often, this covariance function is taken to be stationary, i.e., a function only of the distance between the input points. Stationary covariance functions are appealing due to their intuitive interpretation and their relative ease of construction via Bochner's Theorem (Gibbs, 1997).

Appearing in *Proceedings of the 25$^{th}$ International Conference on Machine Learning*, Helsinki, Finland, 2008. 

Unfortunately, stationarity is often an unrealistic assumption. We expect many problems of interest to have nontrivial nonstationarity in the form of input-dependent noise, length scale or amplitude. While input-dependent noise and length-scale have been well-studied in the literature, nonstationarity in the form of varying amplitude has received relatively little attention.

One approach to modeling such data is to directly specify a covariance function with nonstationary properties (Gibbs, 1997; Higdon et al., 1999). In machine learning, however, we find it undesirable to need to specify the covariance nonstationarity *a priori*; rather we wish to infer it. Moreover, as the objective with Gaussian process regression is to perform nonparametric inference, we would prefer a representation of the nonstationarity which is also nonparametric.

Several approaches have been proposed to solve the problem of learning a length scale that varies across the input space. One of the first techniques was that of Sampson and Guttorp (1992), who model a spline-based mapping to a latent input space in which the data are stationary. This approach was given a nonparametric Bayesian treatment by Schmidt and O'Hagan (2003). Recently, Paciorek and Schervish (2004) extended the work of Higdon et al. (1999) to learn nonparametric variation of the covariance kernel. Other approaches involve Gaussian process mixtures (Rasmussen, 2000), augmentation of the input space (Pfingsten et al., 2006), and weighted sums of locally-stationary processes (Nott & Dunsmuir, 2002).

A related problem is input-dependent observation noise in the Gaussian process, addressed by Goldberg et al. (1998), who model a log-noise term in the covariance function with another Gaussian process, and by Le et al. (2005) who model nonstationary noise by performing regression in the natural parameter space of the exponential family. Snelson and Ghahramani (2006) achieve nonstationary noise as a side effect of the combination of input dimensionality reduction and a sparse approximation using pseudo-data.

In this paper, we propose the Gaussian process product model (GPPM) to address smooth input-dependent changes in amplitude. The GPPM models the data as the pointwise product of two latent stationary Gaussian processes. This approach has the notable computational advantage over remappings of the input space in that high dimensional problems pose no intrinsic scalability problems. Remapping the input nonparametrically while maintaining the input dimension requires at least as many latent processes as input dimensions. In contrast, the GPPM uses only a single additional GP regardless of input dimension. We develop a quadrature-based Expectation Propagation (EP) algorithm for efficient approximate inference in the GPPM model. The EP approach allows us to use the estimated marginal likelihood of the model to learn empirical settings of the Gaussian process hyperparameters. The approximate inference procedure we describe yields uncertainty in the nonstationarity, while avoiding expensive MCMC methods that are typically required. We additionally develop useful approximations for the predictive distribution arising from the EP approximation, and discuss rapid learning of a MAP estimate of the nonstationarity when observations can be considered noise free. This model is similar to that presented by Turner and Sahani (2008), who modulate sounds with Gaussian processes, however the GPPM is intended for the general regression problem and our inference approach differs significantly.

## 2. Gaussian Process Regression

In Gaussian process regression, we find a distribution over functions of the form $f : \mathcal{X} \to \mathbb{R}$, $\mathcal{X} = \mathbb{R}^m$. For a comprehensive introduction see Rasmussen and Williams (2006). The data consist of $N$ input/output pairs $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}^N$, $\boldsymbol{x}_n \in \mathcal{X}$, $y_n \in \mathbb{R}$. A vector of output points has a Gaussian prior distribution with a mean function $\mu(\boldsymbol{x})$, which we take to be zero, and a positive-definite covariance function $C(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta})$. This construction gives an analytic Gaussian predictive distribution for an unseen output $y_\star \sim \mathcal{N}(\mu_\star, v_\star)$:

$$\mu_\star = \boldsymbol{k}_N^\mathsf{T} \boldsymbol{C}_N^{-1} \boldsymbol{y}_N, \quad v_\star = C(\boldsymbol{x}_\star, \boldsymbol{x}_\star) - \boldsymbol{k}_N^\mathsf{T} \boldsymbol{C}_N^{-1} \boldsymbol{k}_N,$$

where $\boldsymbol{k}_N = [C(\boldsymbol{x}_\star, \boldsymbol{x}_1; \boldsymbol{\theta}), \ldots, C(\boldsymbol{x}_\star, \boldsymbol{x}_N; \boldsymbol{\theta})]^\mathsf{T}$, and $\boldsymbol{C}_N$ is the covariance matrix formed from the observed data. The log evidence, or log marginal likelihood after integrating out all possible functions is

$$\mathcal{L} = -\frac{1}{2} \ln |\boldsymbol{C}_N| - \frac{1}{2} \boldsymbol{y}_N^\mathsf{T} \boldsymbol{C}_N^{-1} \boldsymbol{y}_N - \frac{N}{2} \ln 2\pi. \qquad (1)$$

Stationary covariance functions only depend on a distance measure $d$ between $\boldsymbol{x}$ and $\boldsymbol{x}'$, for example the Mahalanobis distance $d(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x} - \boldsymbol{x}')^\mathsf{T} \boldsymbol{W} (\boldsymbol{x} - \boldsymbol{x}')$ with positive definite $\boldsymbol{W}$. Covariance functions that depend only on distance are appealing due to the intuition that the outputs of the function should covary in inverse proportion to how far the inputs are from each other. The model proposed in this paper attempts to retain this intuition while providing a mechanism for the relationship between distance and covariance to vary across the input space.

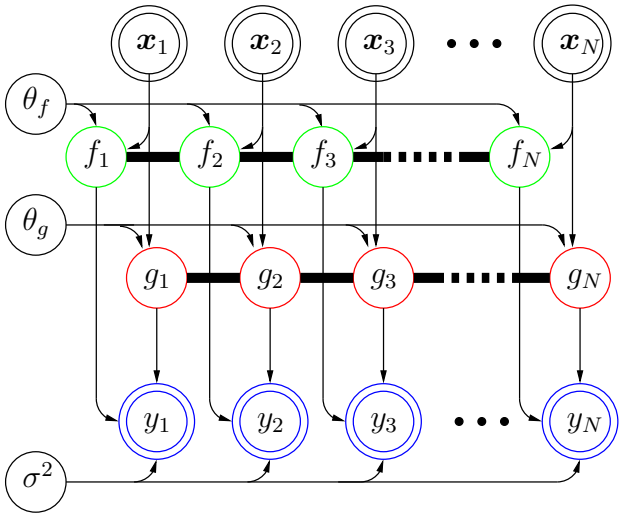



*Figure 1.* A graphical model describing the GPPM. The thick lines connecting the values of $\boldsymbol{f}$ and $\boldsymbol{g}$ represent undirected connections associated with the Gaussian process. The double-lined circles around the $\boldsymbol{y}$ values represent observables. Both $f(x)$ and $g(x)$ have the same input space.

## 3. The Gaussian Process Product Model

In the Gaussian process product model (GPPM), the observed outputs $\{y_n\}^N$ are modeled by a pointwise product of two latent functions, plus independent zero-mean Gaussian noise with variance $\sigma^2$. One latent function $f : \mathcal{X} \to \mathbb{R}$, is modulated by the other function $g : \mathcal{X} \to \mathbb{R}$ that has been exponentiated, so that

$$y_n \sim \mathcal{N}(f(\boldsymbol{x}_n) e^{g(\boldsymbol{x}_n)}, \, \sigma^2). \qquad (2)$$

We place independent zero-mean Gaussian process priors on $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$, with covariance functions $C_f(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}_f)$ and $C_g(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}_g)$, respectively. Figure 1 shows a graphical interpretation of this model. Our convention is that $f(\boldsymbol{x})$ captures local near-stationary variations in the observed function and $g(\boldsymbol{x})$ captures slowly-varying amplitude nonstationarity. The lengthscale hyperparameters of these covariance functions (and their hyperpriors) should be chosen to reflect prior beliefs about such variations. To give the flavor of this model, Figure 2 shows several samples from the GPPM.

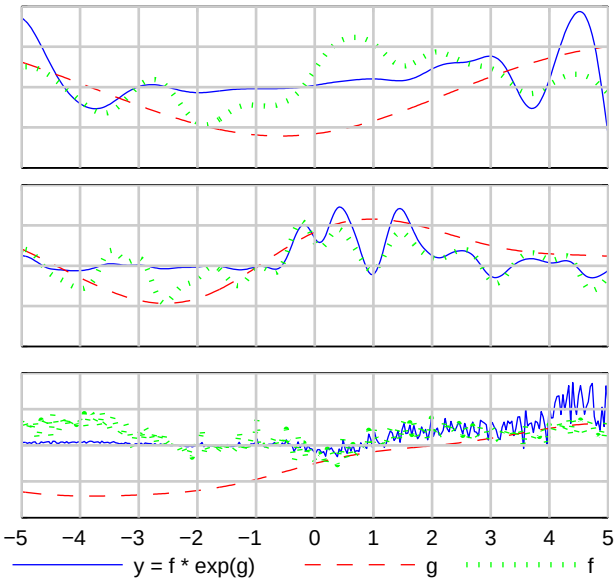



*Figure 2.* Three samples from the GPPM with different parameters. In the top plot, the length scales are $l_f = 0.5$ and $l_g = 4.0$. In the middle plot, both are shorter: $l_f = 0.25$ and $l_g = 2.0$. In the bottom plot, $l_f = 0.5$ and $l_g = 2.0$, but $f(x)$ also has additive noise.

Note that the pointwise product of a Gaussian process prior with any known function $a(\boldsymbol{x})$ results in a covariance function given by $C'(\boldsymbol{x}, \boldsymbol{x}') = a(\boldsymbol{x})C(\boldsymbol{x}, \boldsymbol{x}')a(\boldsymbol{x}')$ and that this function is guaranteed to be positive definite. In the GPPM we use an exponentiated form $a(\boldsymbol{x}) = \exp\{g(\boldsymbol{x})\}$ in order to reduce the multimodality of the posterior on the latent functions, but this is not critical for the validity of the covariance function. Without restricting the sign of one of the functions, there would be at least $2^N$ posterior modes, as each observation could be explained by the same latent function values with flipped signs.

## 4. Factor Inference in the GPPM

The basic GPPM inference task is to determine the posterior distribution over the values of the latent functions $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$ at the input locations $\{\boldsymbol{x}_n\}^N$. These latent function values will be denoted $f_n = f(\boldsymbol{x}_n)$ and $g_n = g(\boldsymbol{x}_n)$ for brevity. Additionally we will write the vectors of these latent values in bold type: $\boldsymbol{f} = [f_1, \ldots, f_N]^\mathsf{T}$ and $\boldsymbol{g} = [g_1, \ldots, g_N]^\mathsf{T}$. With this notation and with $\boldsymbol{C}_f$ and $\boldsymbol{C}_g$ representing the GP-derived covariance matrices on $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$ respectively, the posterior distribution of the latent functions is

$$p(\boldsymbol{f}, \boldsymbol{g} \,|\, \mathcal{D}, \boldsymbol{\theta}) \propto \mathcal{N}(\boldsymbol{f};\, \boldsymbol{0}, \boldsymbol{C}_f)\mathcal{N}(\boldsymbol{g};\, \boldsymbol{0}, \boldsymbol{C}_g) \times \prod_{n=1}^{N} \mathcal{N}(y_n;\, f_n e^{g_n},\, \sigma^2). \quad (3)$$

### 4.1. Approximate Inference

Approximate inference via variational methods is appealing due to its determinism and potential computational savings. In the GPPM, several properties affect our choice of approximation. First, we expect that the posterior will be approximately Gaussian, as we have strong Gaussian process priors and a near-Gaussian likelihood. Second, the likelihood factorizes to $N$ independent terms, each involving one point from the two latent functions. Third, these likelihood factors introduce nontrivial dependencies between $\boldsymbol{f}$ and $\boldsymbol{g}$ so that a factorized approximation is inappropriate. We address these properties using Expectation Propagation.

#### 4.1.1. Expectation Propagation

Expectation Propagation (Minka, 2001) makes successive *local* approximations of factors in a joint density, typically using exponential-family distributions, to yield a *global* approximation that is optimal under a divergence measure. EP is particularly well-suited for approximation of Bayesian posterior distributions with i.i.d. data as in Equation 3, as each factor only involves a few of the unknown parameters.

Our construction of the EP approximation is similar to that used by Rasmussen and Williams (2006) for binary Gaussian process classification. The prior on $\boldsymbol{f}$ and $\boldsymbol{g}$ is Gaussian with zero mean and a block covariance matrix arising from the independent Gaussian process priors. For notational convenience, we will write $\boldsymbol{\phi}$ to be the concatenation of $\boldsymbol{f}$ and $\boldsymbol{g}$ so that $\boldsymbol{\phi} = [f_1, \ldots, f_N, g_1, \ldots, g_N]^\mathsf{T}$, and $\boldsymbol{\phi}_n$ to be the $n$th pair $[f_n\ g_n]^\mathsf{T}$. The prior can now be written

$$p(\boldsymbol{\phi}) = \mathcal{N}(0, \boldsymbol{\Sigma}_{\mathsf{GP}}), \qquad \boldsymbol{\Sigma}_{\mathsf{GP}} = \begin{bmatrix} \boldsymbol{C}_f & 0 \\ 0 & \boldsymbol{C}_g \end{bmatrix}.$$

The aim of EP is to approximate the exact posterior distribution of Equation 3 with a tractable alternative

$$q(\boldsymbol{f}, \boldsymbol{g} \,|\, \mathcal{D}, \boldsymbol{\theta}) \propto \mathcal{N}(0, \boldsymbol{\Sigma}_{\mathsf{GP}}) \prod_{n=1}^{N} \tilde{t}_n(f_n, g_n). \quad (4)$$

Each of the exact likelihood terms

$$\mathcal{L}_n(f_n, g_n) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2\sigma^2}\left(f_n e^{g_n} - y_n\right)^2\right\}$$

is approximated with an unnormalized bivariate Gaussian on $f_n$ and $g_n$:

$$\tilde{t}_n(f_n, g_n) = \tilde{Z}_n \exp\left\{-\frac{1}{2}(\boldsymbol{\phi}_n - \tilde{\boldsymbol{\mu}}_n)^\mathsf{T}\tilde{\boldsymbol{\Sigma}}_n^{-1}(\boldsymbol{\phi}_n - \tilde{\boldsymbol{\mu}}_n)\right\}.$$

The product of these likelihood approximations is an unnormalized Gaussian with a block-diagonal covariance matrix.

$$\prod_{n=1}^{N} \tilde{t}_n(\boldsymbol{\phi}_n) = \exp\left\{-\frac{1}{2}(\boldsymbol{\phi} - \tilde{\boldsymbol{\mu}})^\mathsf{T}\tilde{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{\phi} - \tilde{\boldsymbol{\mu}})\right\} \prod_{n=1}^{N} \tilde{Z}_n$$

The overall approximation is Gaussian as well, as it is the product of these Gaussian likelihood approximations and the Gaussian process prior.

$$q(\boldsymbol{f}, \boldsymbol{g} \mid \mathcal{D}, \boldsymbol{\theta}) = \mathcal{N}\left(\boldsymbol{\phi} = \begin{bmatrix}\boldsymbol{f}\\ \boldsymbol{g}\end{bmatrix}; \boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \tag{5}$$
$$\boldsymbol{\Sigma} = \left(\boldsymbol{\Sigma}_{\mathsf{GP}}^{-1} + \tilde{\boldsymbol{\Sigma}}^{-1}\right)^{-1} \qquad \boldsymbol{\mu} = \boldsymbol{\Sigma}\tilde{\boldsymbol{\Sigma}}^{-1}\tilde{\boldsymbol{\mu}}$$

The Expectation Propagation algorithm proceeds by iteratively updating the parameters of the local approximations $t_n$, leaving all other approximate factors fixed. In this iterative procedure the update of the $n$th site can be understood as the minimization of the KL divergence between two approximating distributions: the product of the cavity distribution times the *exact* local likelihood, and the product of the cavity distribution times the *approximate* local likelihood. The insight of EP is that the cavity distribution "focuses" the approximation on the most relevant area.

$$\hat{\boldsymbol{\mu}}_n, \hat{\boldsymbol{\Sigma}}_n = \underset{\boldsymbol{\mu}', \boldsymbol{\Sigma}'}{\operatorname{argmin}} \operatorname{KL}\Big[\mathcal{N}(\boldsymbol{\mu}_{/n}, \boldsymbol{\Sigma}_{/n}) \times \overbrace{\mathcal{L}_n(f_n, g_n)}^{\text{exact factor}} \,\Big|\Big|\, \mathcal{N}(\boldsymbol{\mu}_{/n}, \boldsymbol{\Sigma}_{/n}) \times \underbrace{\tilde{t}_n(f_n, g_n | \boldsymbol{\mu}', \boldsymbol{\Sigma}')}_{\text{approximation}}\Big]$$

The cavity distribution for site $n$ is the product of the prior and all approximate sites excluding the $n$th. This is Gaussian with parameters

$$\boldsymbol{\Sigma}_{/n} = \left(\boldsymbol{\Sigma}_n^{-1} - \tilde{\boldsymbol{\Sigma}}_n^{-1}\right)^{-1} \tag{6}$$
$$\boldsymbol{\mu}_{/n} = \boldsymbol{\Sigma}_{/n}\left(\boldsymbol{\Sigma}_n^{-1}\boldsymbol{\mu}_n - \tilde{\boldsymbol{\Sigma}}_n^{-1}\tilde{\boldsymbol{\mu}}_n\right). \tag{7}$$

As shown by Minka (2001), the minimum of an inclusive KL divergence is achieved when the moments are equal. Thus to find the best-fitting Gaussian, it is sufficient to find the first and second moments of the product of the cavity distribution and the exact likelihood. We also find the "zeroth moment," which is the normalization constant $\hat{Z}_n$. Calculation of these moments is done numerically via Gaussian quadrature, addressed in Section 4.1.2.

Once the moments of the product have been found, we use them to recover the optimal parameters of the local approximation:

$$\begin{aligned}
\tilde{\boldsymbol{\Sigma}}_n &= \left(\hat{\boldsymbol{\Sigma}}_n^{-1} - \boldsymbol{\Sigma}_{/n}^{-1}\right)^{-1} \\
\tilde{\boldsymbol{\mu}}_n &= \tilde{\boldsymbol{\Sigma}}_n\left(\hat{\boldsymbol{\Sigma}}_n^{-1}\hat{\boldsymbol{\mu}}_n - \boldsymbol{\Sigma}_{/n}^{-1}\boldsymbol{\mu}_{/n}\right) \\
\ln \tilde{Z}_n &= \ln \hat{Z}_n - \frac{1}{2}\ln|\tilde{\boldsymbol{\Sigma}}_n| + \frac{1}{2}\ln|\boldsymbol{\Sigma}_{/n}| + \frac{1}{2}\tilde{\boldsymbol{\mu}}_n^{\mathsf{T}}\tilde{\boldsymbol{\Sigma}}_n^{-1}\tilde{\boldsymbol{\mu}}_n \\
&\quad + \frac{1}{2}\boldsymbol{\mu}_{/n}^{\mathsf{T}}\boldsymbol{\Sigma}_{/n}^{-1}\boldsymbol{\mu}_{/n} - \frac{1}{2}\hat{\boldsymbol{\mu}}_n^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}_n^{-1}\hat{\boldsymbol{\mu}}_n.
\end{aligned}$$

Taken together these equations define a fixed-point iteration scheme for approximating the posterior in Equation 3. We initialize the approximations so that the initial estimate of the mean of $\boldsymbol{f}$ is $\boldsymbol{y}$ and the mean of $\boldsymbol{g}$ is zero. We then iterate over each of the $N$ local approximations, and update the overall posterior approximation using Equation 5. To facilitate convergence of EP it is helpful to use damping to update local sites, which we implement in natural parameter space. Convergence of EP is not guaranteed, but given sufficient damping it is found to convergence for the problems we considered so far. Local approximations may not necessarily be positive definite, but as long as the overall approximation remains a valid Gaussian, this does not present a problem. Following from the treatment by Minka (2001) of negative variances, we skip the update of local approximations that would result in invalid global covariance matrices. This has not appeared to affect the accuracy of the global approximation in practice. Figure 3(b) shows the result of applying the EP procedure to a synthetic data set. Marginal error bars are shown for each function and site location.

#### 4.1.2. Gaussian Quadrature for EP

Unfortunately, the moments that minimize the KL divergence of Section 4.1.1 are not available analytically. To resolve this, we use the approach proposed by Zoeter and Heskes (2005) of approximating the moment integrals using Gaussian quadrature. When a definite integral is the product of a nonnegative "weighting function" $w(v)$ and another function $z(v)$, it can be approximated by a sum of weighted evaluations of $z(v)$

$$\int_b^a dv\; w(v)z(v) \approx \sum_{k=1}^{K} w_k z(v_k)$$

where the weights $\{w_k\}$ and abscissae $\{v_k\}$ are determined by the integration interval, the weighting function $w(v)$, and the number of evaluation points $K$. This sum is exact where $z(v)$ is a polynomial of degree $2K-1$. In the case of interest here, the weighting function is the Gaussian cavity distribution, which implies Gauss-Hermite quadrature.

One difficulty is that Gaussian quadrature is generally oriented towards univariate definite integrals and we must solve a two-dimensional integral. When the weighting function is factorizable, this is done straightforwardly by defining a lattice of abscissae and using the Cartesian product of the weights. In the GPPM, however, the cavity distribution has nonzero mean and is not generally factorizable, so we must transform

the integrand prior to to performing Gauss-Hermite quadrature. The factorizable form can be recovered by transforming the abscissae with the inverse Cholesky decomposition of the cavity covariance matrix and the cavity mean. The Gaussian parameters resulting from these moment calculations are denoted $\hat{\boldsymbol{Z}}_n$, $\hat{\boldsymbol{\mu}}_n$, and $\hat{\boldsymbol{\Sigma}}_n$ in Section 4.1.1.

### 4.2. Noise-free MAP Learning

In some applications of the GPPM, it may be that the observations can be considered noise-free. For example, one may model the noise as coming exclusively from the locally-varying function $f(x)$. The appeal of this restricted model is that proposals of the non-stationarity can now be evaluated as $O(N^2)$ rather than $O(N^3)$. This is particularly valuable for finding rapid maximum *a posteriori* (MAP) estimates of the latent modulating function $g(x)$. The computational advantage in the noise-free case comes from the deterministic coupling of the latent functions, given $\boldsymbol{y}$; we can now consider the posterior of $\boldsymbol{g}$ alone:

$$p(\boldsymbol{g} \,|\, \boldsymbol{\theta}_f, \boldsymbol{\theta}_g) \propto p(\mathcal{D} \,|\, \boldsymbol{g}, \boldsymbol{\theta}_f) p(\boldsymbol{g} \,|\, \boldsymbol{\theta}_g). \tag{8}$$

In this form, conditioning on $\boldsymbol{g}$ corresponds to a simple linear transformation of the GP prior on $\boldsymbol{f}$. Using the notational shortcut $\boldsymbol{\mathcal{G}} = \mathrm{diag}([e^{g_1}, e^{g_2}, \ldots, e^{g_N}])$, the log likelihood is

$$\ln p(\mathcal{D} \,|\, \boldsymbol{g}, \boldsymbol{\theta}_f) = -\frac{1}{2} \ln |\boldsymbol{\mathcal{G}} \boldsymbol{C}_f \boldsymbol{\mathcal{G}}| - \frac{1}{2} \boldsymbol{y}^\mathsf{T} [\boldsymbol{\mathcal{G}} \boldsymbol{C}_f \boldsymbol{\mathcal{G}}]^{-1} \boldsymbol{y} - \frac{N}{2} \ln 2\pi.$$

The log posterior over $\boldsymbol{g}$, eliminating irrelevant terms and using $\mathbf{1}$ to indicate a column vector of ones, is

$$\ln p(\boldsymbol{g} \,|\, \mathcal{D}, \boldsymbol{\theta}_f, \boldsymbol{\theta}_g) = -\boldsymbol{g}^\mathsf{T} \mathbf{1} - \frac{1}{2} \boldsymbol{y}^\mathsf{T} [\boldsymbol{\mathcal{G}} \boldsymbol{C}_f \boldsymbol{\mathcal{G}}]^{-1} \boldsymbol{y} - \frac{1}{2} \boldsymbol{g}^\mathsf{T} \boldsymbol{C}_g^{-1} \boldsymbol{g} + \text{const}$$

and the gradient in terms of $\boldsymbol{g}$ is

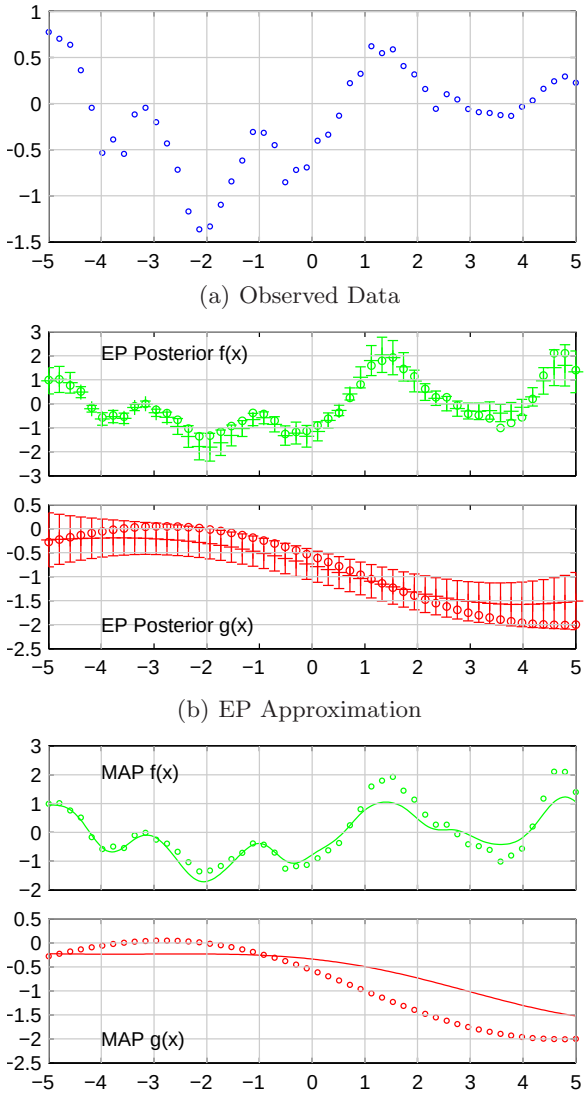
$$\frac{\partial}{\partial \boldsymbol{g}} \ln p(\boldsymbol{g} \,|\, \mathcal{D}, \boldsymbol{\theta}_f, \boldsymbol{\theta}_g) = -\mathbf{1} + \boldsymbol{Y} [\boldsymbol{\mathcal{G}} \boldsymbol{C}_f \boldsymbol{\mathcal{G}}]^{-1} \boldsymbol{y} - \boldsymbol{C}_g^{-1} \boldsymbol{g}$$

where $\boldsymbol{Y} = \mathrm{diag}(\boldsymbol{y})$. As the difficult $O(N^3)$ operations of decomposition or inversion of $\boldsymbol{C}_f$ and $\boldsymbol{C}_g$ can be done in advance, the computational complexity of taking a step in $\boldsymbol{g}$ space is $O(N^2)$. In practice, we have found the MAP estimate to be best when $f(x)$ has additive noise and $g(x)$ is smooth.

## 5. Making Predictions

As with the standard regression model, the primary task of interest is prediction at locations where data have not been observed. For the GPPM we must make predictions for both latent functions, and find the resulting distribution, integrating out the posterior distribution over the latent functions, as in

$$p(y^\star \,|\, \boldsymbol{x}^\star, \mathcal{D}, \boldsymbol{\theta}_f, \boldsymbol{\theta}_g) = \int_{\boldsymbol{f}, \boldsymbol{g}} p(y^\star \,|\, \boldsymbol{x}^\star, \boldsymbol{f}, \boldsymbol{g}) p(\boldsymbol{f}, \boldsymbol{g} \,|\, \mathcal{D}, \boldsymbol{\theta}_f, \boldsymbol{\theta}_g).$$




(a) Observed Data

(b) EP Approximation

(c) MAP Approximation

*Figure 3.* Figure 3(a) shows synthetic data generated from the GPPM with known settings and $\sigma = 0.05$. We applied the Expectation Propagation algorithm to the data and the Gaussian marginal posterior distributions over $\boldsymbol{f}$ and $\boldsymbol{g}$ are shown in Figure 3(b), along with the true $f(x)$ and $g(x)$ indicated as circles. Figure 3(c) shows the result of applying the MAP approximation to the data, despite the known observation noise. The true values are shown for comparison.

The EP scheme of Section 4.1 finds an approximate Gaussian distribution over $\boldsymbol{f}$ and $\boldsymbol{g}$, and this results in a convenient joint Gaussian distribution on $f^\star$ and $g^\star$, the values of the latent functions at $\boldsymbol{x}^\star$, with parameters

$$\boldsymbol{\mu}^\star = \boldsymbol{K}^\mathsf{T}\left(\boldsymbol{\Sigma}_{\mathsf{GP}} + \tilde{\boldsymbol{\Sigma}}\right)^{-1}\tilde{\boldsymbol{\mu}}, \quad \boldsymbol{\Sigma}^\star = \boldsymbol{\kappa} - \boldsymbol{K}^\mathsf{T}\left(\boldsymbol{\Sigma}_{\mathsf{GP}} + \tilde{\boldsymbol{\Sigma}}\right)^{-1}\boldsymbol{K},$$

where

$$\boldsymbol{K} = \begin{bmatrix} C(\boldsymbol{x}^\star, \boldsymbol{x}_1; \boldsymbol{\theta}_f) & 0 \\ C(\boldsymbol{x}^\star, \boldsymbol{x}_2; \boldsymbol{\theta}_f) & 0 \\ \vdots & \vdots \\ C(\boldsymbol{x}^\star, \boldsymbol{x}_N; \boldsymbol{\theta}_f) & 0 \\ 0 & C(\boldsymbol{x}^\star, \boldsymbol{x}_1; \boldsymbol{\theta}_g) \\ \vdots & \vdots \\ 0 & C(\boldsymbol{x}^\star, \boldsymbol{x}_N; \boldsymbol{\theta}_g) \end{bmatrix}$$

$$\boldsymbol{\kappa} = \begin{bmatrix} C(\boldsymbol{x}^\star, \boldsymbol{x}^\star; \boldsymbol{\theta}_f) & 0 \\ 0 & C(\boldsymbol{x}^\star, \boldsymbol{x}^\star; \boldsymbol{\theta}_g) \end{bmatrix}.$$

We expect that the resulting predictive distribution on $y^\star$ will be heavy-tailed and have similar properties to the noncentral Student's $t$ distribution. To approximate the true distribution's heavy tails analytically, one approach is to generate several samples from $g^\star$ and use the conditional distribution on $f^\star$ to create a mixture of Gaussians:

$$p(y^\star \mid \boldsymbol{x}^\star, \mathcal{D}, \boldsymbol{\theta}_f, \boldsymbol{\theta}_g) \approx \sum_i \mathcal{N}(y^\star; \mu^\star_{f|g_i} e^{g^\star_i}, v^\star_{f|g_i} e^{2g^\star_i}).$$

We have used $\mu^\star_{f|g_i}$ and $v^\star_{f|g_i}$ to indicate the conditional Gaussian parameters on $f^\star$ given the $i$th marginal sample from $g^\star$.

If the heavy-tailed properties are not significant for the application, and a single Gaussian distribution is preferred, then a more tractable alternative is to linearize the model around the mean $\boldsymbol{\mu}^\star$. This is a similar approach to the Extended Kalman Filter (EKF) (Haykin, 2001), which uses the first terms of the Taylor series of a nonlinear function to maintain Gaussian uncertainty in latent state estimation. The resulting approximation is

$$f^\star e^{g^\star} \underset{\boldsymbol{\mu}^\star}{\approx} \mu^\star_f e^{\mu^\star_g} + \begin{bmatrix} e^{\mu^\star_g} \\ \mu^\star_f e^{\mu^\star_g} \end{bmatrix}^\mathsf{T} \begin{bmatrix} f^\star - \mu^\star_f \\ g^\star - \mu^\star_g \end{bmatrix}$$

which transforms the Gaussian on $f^\star$ and $g^\star$ into one on $y^\star$ with parameters

$$\mu^\star_y = \mu^\star_f e^{\mu^\star_g} \qquad v^\star_y = \begin{bmatrix} e^{\mu^\star_g} \\ \mu^\star_f e^{\mu^\star_g} \end{bmatrix}^\mathsf{T} \boldsymbol{\Sigma}^\star \begin{bmatrix} e^{\mu^\star_g} \\ \mu^\star_f e^{\mu^\star_g} \end{bmatrix} + \sigma^2.$$

## 6. Hyperparameter Learning

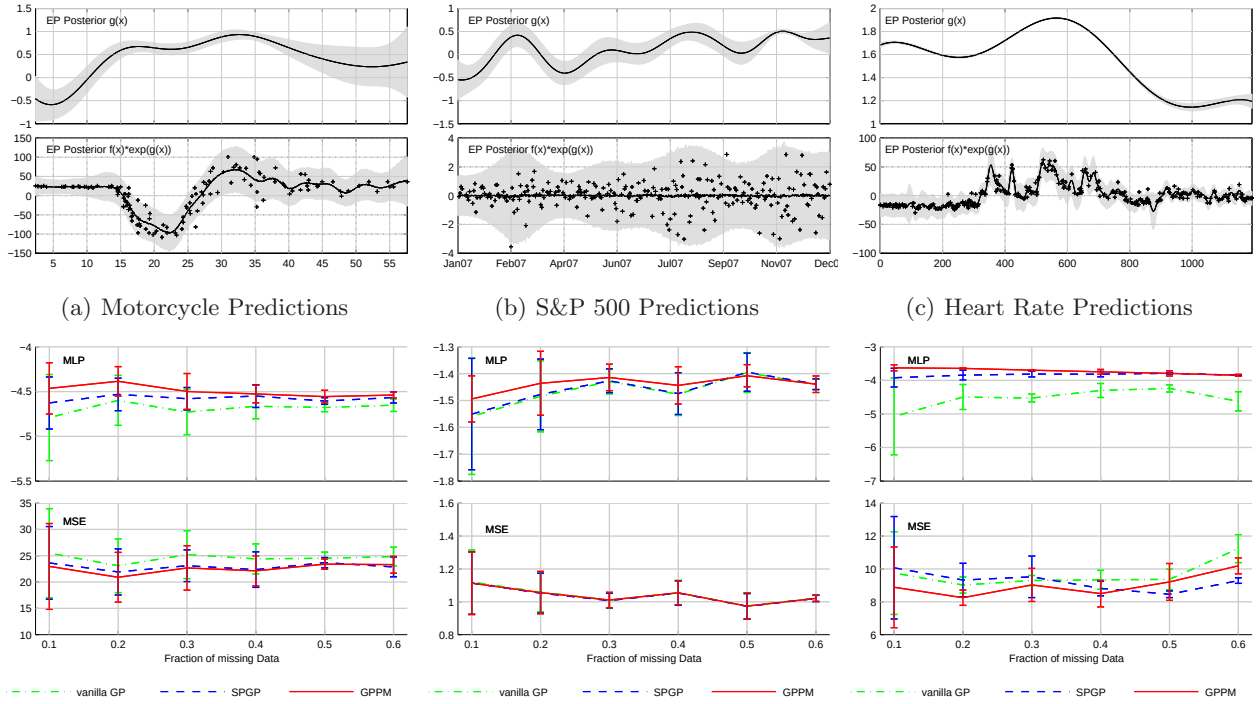
When performing Gaussian process regression, we are commonly interested in appropriate settings of the hyperparameters controlling the covariance function. These hyperparameters generally determine the length scale of correlations, the output variation (or amplitude) of the function, and the noise level. In the GPPM, we wish to find appropriate hyperparameter settings for *both* latent functions, given the data. While the vanilla Gaussian process offers the marginal likelihood analytically, it is not available directly in the GPPM. Fortunately, the EP algorithm of Section 4.1 provides a convenient estimate of the marginal likelihood, using the zeroth moments mentioned previously.

$$\ln Z_{\mathsf{EP}} = \frac{1}{2}\ln|\boldsymbol{\Sigma}| - \frac{1}{2}\ln|\boldsymbol{\Sigma}_{\mathsf{GP}}| - \frac{1}{2}\tilde{\boldsymbol{\mu}}^\mathsf{T}\tilde{\boldsymbol{\Sigma}}^{-1}\tilde{\boldsymbol{\mu}} + \frac{1}{2}\boldsymbol{\mu}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \sum_{n=1}^{N}\ln\tilde{Z}_n$$

In principle it is also possible to evaluate the gradients of $\ln Z_{\mathsf{EP}}$ with respect to hyperparameters following for instance (Seeger, 2005). In practice however, the quadrature-based moment calculation is numerically not stable enough to provide precise gradients. We hence reverted to gradient-free optimization methods to determine hyper parameter settings. We suggest setting hyperpriors to reflect the intuition described in Section 3 of $f(x)$ capturing local near-stationary variations and $g(x)$ capturing slowly varying nonstationarity on a larger lengthscale.

## 7. Results

We evaluated the GPPM model on three data sets. First, we examined the motorcycle data set (Parker & Rice, 1985), a well-studied example of a nonstationary regression task. The data are acceleration force in g's on a helmet during impact, as a function of time in milliseconds. Figure 4(a) in the upper plot shows the EP approximation found for the latent $g(x)$ function, and in the lower plot shows the Gaussian approximation to the predictive distribution, overlaid with the true data. The GPPM finds a good fit in most regions except where the $g(x)$ function becomes quite small. In these regions the uncertainty in the modulating function creates unrealistically large prediction error bars. We evaluated the accuracy of predictions using a fill-in test, where a fraction of the data are removed from the training set and compared to the model's predictions. Figure 4(d) depicts the mean log probability and the mean squared error for missing data as a function of the fraction of missing data. The GPPM outperforms both a vanilla GP and the sparse pseudo-input process (SPGP) (Snelson & Ghahramani, 2006) using either of the performance measures. We chose the SPGP for comparison to the GPPM, as it is one of the few




(a) Motorcycle Predictions (b) S&P 500 Predictions (c) Heart Rate Predictions

(d) Motorcycle Fill–in Performance (e) S&P Fill–in Performance (f) Heart Rate Fill–in Performance

*Figure 4.* Top panel: Predictive distribution of GPPM for three different data sets. The upper plot shows the EP approximation to the posterior of the log-modulating function $g(x)$ with $2\sigma$ error bars. The lower plot shows the raw data, along with the $2\sigma$ approximate predictive distribution. Lower panel: Fill-in test for corresponding data sets comparing three models. The upper plot shows the mean log probability of the missing data as a function of the fill-in rate. The lower plot shows the root mean squared error for these data. Both plots show mean values and $2\sigma$ error bars, calculated from four training/test splits.

methods capable of representing nonstationarity without requiring MCMC. Hyperparameters for the SPGP and the vanilla GP were set via ML-II optimization (Rasmussen & Williams, 2006). To set hyperparameters in the GPPM, a grid search was used, centered on the settings for the vanilla GP.

We also examined the performance of the GPPM for daily log returns of the S&P 500 stock index during 2007. We expect that these data will be well-modeled by a latent $f(x)$ comprised primarily of noise. The log modulating function $g(x)$ can be interpreted roughly as the log "volatility" of the stochastic process and is shown in the upper plot of Figure 4(b). The corresponding expected envelope is shown against the true data in the lower plot. Performance measures against the standard Gaussian process and the SPGP are shown in Figure 4(e). In this example mean predictions are equally good for three all models, but GPPM yields nonstationary uncertainty which results in an improved mean log probability.

As a last application we applied the GPPM to 23 hours of heart rate data, sampled at 5 minute intervals. Based on the physiological properties of heart rates, we expect correlations on a short time scale to be captured by $f(x)$. These local correlations will be modulated by an activity profile over a daily time scale. Figure 4(c) illustrates that these amplitude modulations are picked up by the latent $g(x)$ leading to improved predictive performance compared to the vanilla GP and SPGP, as shown in Figure 4(f).

## 8. Discussion

We have introduced the Gaussian process product model for modeling nonstationary amplitude in regression. We have presented an approximate inference algorithm using Expectation Propagation to infer the latent functions in this model and have exploited this approximation to make tractable predictions and enable hyperparameter learning. When examined on real-world data, the GPPM has yielded promising results, outperforming the vanilla Gaussian process. It has also outperformed an alternative approach to nonstationary regression in the SPGP, although it should be noted that the SPGP's focus is purely on efficient regression and not on modeling nonstationarity *per se*.

Computationally, the model we have presented, combined with the EP implementation has two appealing properties. First, as we expect the number of EP iterations to be independent of the number of data (Minka, 2001), and each local calculation is a $O(N^2)$ rank-one update of the inverse, the overall algorithm is $O(N^3)$. The GPPM is therefore only a constant multiple more expensive than performing standard Gaussian process regression. Second, in contrast to methods of modeling nonstationarity on the input side, the GPPM does not introduce additional latent spaces if the input dimensionality increases. The additional computational complexity of using the GPPM is essentially independent of input dimension.

In future work, a more comprehensive examination of inference of hyperparameters is warranted. We also expect that the basic idea of this model can be used to perform vector regression with correlation that varies across the input space.

## Acknowledgements

The authors wish to thank David MacKay for helpful comments. This work was funded by the Gates Cambridge Trust.

## References

Gibbs, M. (1997). *Bayesian Gaussian processes for regression and classification*. Doctoral dissertation, University of Cambridge, Cambridge.

Goldberg, P., Williams, C., & Bishop, C. (1998). Regression with input-dependent noise: a Gaussian process treatment. *Advances in Neural Processing Systems 10* (pp. 493–499). Cambridge, MA: MIT Press.

Haykin, S. (Ed.). (2001). *Kalman filtering and neural networks*. New York: John Wiley and Sons, Inc.

Higdon, D., Swall, J., & Kern, J. (1999). Nonstationary spatial modeling. In J. Bernardo, J. Berger, A. Dawid and A. Smith (Eds.), *Bayesian statistics 6*, 761–768. Oxford: Oxford University Press.

Le, Q., Smola, A., & Canu, S. (2005). Heteroscedastic Gaussian process regression. *Proceedings of the 22nd International Conference on Machine Learning (ICML)*.

Minka, T. P. (2001). *A family of algorithms for approximate Bayesian inference*. Doctoral dissertation, Massachusetts Institute of Technology, Cambridge, MA.

Nott, D., & Dunsmuir, W. (2002). Estimation of nonstationary spatial covariance structure. *Biometrika*, *89*, 819–829.

Paciorek, C., & Schervish, M. (2004). Nonstationary covariance functions for Gaussian process regression. *Advances in Neural Information Processing Systems 16*. Cambridge, MA: MIT Press.

Parker, R., & Rice, J. (1985). Discussion of "Some aspects of the spline smoothing approach to nonparametric curve fitting" by B.W. Silverman. *Journal of the Royal Statistical Society, Series B*, *47*, 40–42.

Pfingsten, T., Kuss, M., & Rasmussen, C. (2006). Nonstationary Gaussian process regression using a latent extension of the input space. www.kyb.mpg.de/publication.html?publ=3985.

Rasmussen, C. (2000). The infinite Gaussian mixture model. *Advances in Neural Information Processing Systems 12* (pp. 554–560). Cambridge, MA: MIT Press.

Rasmussen, C., & Williams, C. (2006). *Gaussian processes for machine learning*. Cambridge, MA: MIT Press.

Sampson, P., & Guttorp, P. (1992). Nonparametric estimation of nonstationary spatial covariance structure. *Journal of the American Statistical Association*, *87*, 108–119.

Schmidt, A. M., & O'Hagan, A. (2003). Bayesian inference for nonstationary spatial covariance structure via spatial deformations. *Journal of the Royal Statistical Society, Series B*, *65*, 745–758.

Seeger, M. (2005). *Expectation propagation for exponential families* (Technical Report). Technical report, University of California at Berkeley, 2005. See www. kyb. tuebingen. mpg. de/bs/people/seeger.

Snelson, E., & Ghahramani, Z. (2006). Variable noise and dimensionality reduction for sparse Gaussian processes. *Proceedings of the 22nd International Conference on Uncertainty in Artificial Intelligence*.

Turner, R., & Sahani, M. (2008). Modeling natural sounds with modulation cascade processes. *Advances in Neural Information Processing Systems 21*. Cambridge, MA: MIT Press.

Zoeter, O., & Heskes, T. (2005). Gaussian quadrature based expectation propagation. *Proceedings of Artificial Intelligence and Statistics 2005*.