Variational Heteroscedastic Gaussian Process Regression

Miguel Lázaro-Gredilla MIGUELLG@GTAS.DICOM.UNICAN.ES
Department of Communications Engineering, Universidad de Cantabria, Spain

Michalis K. Titsias MTITSIAS@CS.MAN.AC.UK
School of Computer Science, University of Manchester, UK

Abstract

Standard Gaussian processes (GPs) model observations' noise as constant throughout input space. This is often a too restrictive assumption, but one that is needed for GP inference to be tractable. In this work we present a non-standard variational approximation that allows accurate inference in heteroscedastic GPs (i.e., under input-dependent noise conditions). Computational cost is roughly twice that of the standard GP, and also scales as $\mathcal{O}(n^3)$. Accuracy is verified by comparing with the golden standard MCMC and its effectiveness is illustrated on several synthetic and real datasets of diverse characteristics. An application to volatility forecasting is also considered.

1. Introduction

In the regression task, we are given a dataset consisting of input-output pairs $\mathcal{D} \equiv \{\mathbf{x}_i \in \mathbb{R}^D, y_i = y(\mathbf{x}_i) \in \mathbb{R}\}_{i=1}^n$ modeled as the sum of some unknown latent function $f(\mathbf{x})$ plus independent noise

$$y(\mathbf{x}_i) = f(\mathbf{x}_i) + \varepsilon_i. \tag{1}$$

The typical setting is to assume that $\{\varepsilon_i\}_{i=1}^n$ are uncorrelated zero-mean Gaussian random variables with a global or constant variance, which in statistics is referred to as *homoscedastic* (Gaussian) regression (Silverman, 1985). A Bayesian non-parametric approach to homoscedastic regression is to place a Gaussian process (GP) prior on the latent function $f(\mathbf{x})$. The GP approach is flexible and also has the elegant property that both the predictive density (given model hyperparameters) and the marginal likelihood (useful for learning the hyperparameters) are given by analytic expressions; see e.g. (Rasmussen & Williams, 2006).

In many applications, however, the assumption of constant variance can be unrealistic and it is highly desirable to consider models with input-dependent variance. This leads to *heteroscedastic* regression, which has numerous applications in statistics, especially in econometrics and statistical finance. For instance, modeling time series with time-varying volatility and stochastic volatility models is an important research field in economics (Brooks et al., 2001; Brownlees et al., 2009; Liu, 2001). Heteroscedastic regression can also have several other applications in machine learning, such as robotics (Kersting et al., 2007), and in general is a more flexible way of doing regression that includes homoscedastic regression as a special case. However, inference in heteroscedastic GP (HGP) regression is very challenging since, unlike in the homoscedastic case, the predictive density and marginal likelihood are no longer analytically tractable.

In this work, we introduce a novel variational inference method for HGP regression that is based on variational Bayes and the Gaussian approximation (Opper & Archambeau, 2009; Nickisch & Rasmussen, 2008; Honkela et al., 2011). There is relevant previous work on approximate inference in HGP regression, particularly the Markov chain Monte Carlo (MCMC) fully Bayesian method considered in (Goldberg et al., 1998) and the maximum a posteriori (MAP) approach used in (Kersting et al., 2007; Quadrianto et al., 2009). These approaches have certain limitations: MCMC is very slow in large scale applications, whereas MAP estimation does not integrate out all latent variables and is prone to overfitting. Our variational framework overcomes these limitations, by maximizing a rigorous and analytically tractable lower bound on the exact marginal likelihood. As it will be shown in the experiments, the variational method is very fast (needing roughly twice the time a normal GP does) and at the

Appearing in *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.

same time, very accurate. The latter is validated experimentally through a comparison with the elliptical slice sampling MCMC method (Murray et al., 2010).

We will exploit some ideas regarding the maximization of variational lower bounds that simplify optimization. Specifically, we will first introduce the concept of *marginalized variational* optimization where a mean field bound with a two-factor variational distribution $q(\mathbf{f})q(\mathbf{g})$, is firstly maximized by removing analytically and optimally its dependence w.r.t. $q(\mathbf{f})$. Then, by exploiting the structure of the stationary points at the local maxima, a suitable reparametrization can be used to reduce the number of variational parameters from $n + n(n+1)/2$ to just n.

In the experiments we will validate our method against the (slow) golden standard MCMC, and compare it with the MAP approach (Quadrianto et al., 2009) and the standard GP on synthetic and real-word regression datasets. We then consider applications in stochastic volatility forecasting and compare our method with GARCH(1,1) (Hansen & Lunde, 2005). In this case, and using the stochastic volatility problem considered in (Girolami & Calderhead, 2011), we also show that our variational approximation provides remarkably close posterior predictions to the ones obtained by Riemann manifold Hamiltonian Monte Carlo.

2. The Heteroscedastic GP Model

To define the Heteroscedastic GP (HGP) model we proceed in a Bayesian nonparametric fashion and place a GP prior on the unknown function $f(\mathbf{x})$ and Gaussian priors on the noise terms ε_i (see Eq. (1)):

$$f(\mathbf{x}) \sim \mathcal{GP}(0, k_f(\mathbf{x}, \mathbf{x}')), \quad \varepsilon_i \sim \mathcal{N}(0, r(\mathbf{x}_i)).$$

Therefore, observation noise has a possibly different variance $r(\mathbf{x})$ at each input point $\mathbf{x}$. If we restrict the variance to be constant across all the input space (i.e., $r(\mathbf{x}) = \sigma^2$), the described model corresponds to the standard (homoscedastic) GP regression, for which analytical inference is possible. Here, we are interested in the case in which the unknown function $r(\mathbf{x})$ can vary with $\mathbf{x}$ and can take any form. To ensure positivity, we parametrize $r(\mathbf{x}) = e^{g(\mathbf{x})}$ and place a GP prior $g(\mathbf{x}) \sim \mathcal{GP}(\mu_0, k_g(\mathbf{x}, \mathbf{x}'))$.

Once some parametric form has been selected for covariance functions $k_f(\mathbf{x}, \mathbf{x}')$ and $k_g(\mathbf{x}, \mathbf{x}')$, the model is fully specified and depends only on the covariance hyperparameters (respectively $\boldsymbol{\theta}_f$ and $\boldsymbol{\theta}_g$) and the noise mean hyperparameter μ_0. Note that, unlike other HGP proposals, we explicitly take into account μ_0 so that we can control the *scale* of the noise process (since μ_0 is exponentiated). Assuming $\mu_0 = 0$ imposes an arbitrary noise scale preference, which is not desirable. HGPs are more flexible than standard GPs, but unfortunately they are no longer analytically tractable.

In the following we will use vectorized forms to refer to the observations and the latent functions evaluated at the training points so that $\mathbf{y} \equiv \{y_i\}_{i=1}^n$, $\mathbf{f} \equiv \{f(\mathbf{x}_i)\}_{i=1}^n$, and $\mathbf{g} \equiv \{g(\mathbf{x}_i)\}_{i=1}^n$, respectively.

3. Variational Approximation

The marginal log-likelihood (evidence) of the HGP model cannot be computed analytically. However, it is possible to lower bound it variationally with an analytically tractable expression. We will first introduce a marginalized version of the standard variational approximation, in general form, and then apply it to the HGP model. Hyperparameter learning is deferred to the end of this section. For the sake of simplicity, here we omit conditioning on inputs $\mathbf{X} \equiv \{\mathbf{x}_i\}_{i=1}^n$ or hyperparameters $\boldsymbol{\theta} \equiv \{\boldsymbol{\theta}_f, \boldsymbol{\theta}_g, \mu_0\}$. Supplementary material and code can be obtained from `http://www.tsc.uc3m.es/~miguel`.

3.1. The Marginalized Variational Bound

The standard variational approximation defines

$$F(q(\mathbf{f}), q(\mathbf{g})) = \log p(\mathbf{y}) - \mathrm{KL}(q(\mathbf{f})q(\mathbf{g}) || p(\mathbf{f}, \mathbf{g}|\mathbf{y})),$$

where it is clear that F lower bounds the evidence, i.e., $\log p(\mathbf{y}) \geq F(q(\mathbf{f}), q(\mathbf{g}))$ for any possible choice of the variational probability densities $q(\mathbf{f})$ and $q(\mathbf{g})$. Since the value of $\log p(\mathbf{y})$ is independent of the variational densities, maximizing this bound w.r.t. $q(\mathbf{f})$ and $q(\mathbf{g})$ is equivalent to minimizing $\mathrm{KL}(q(\mathbf{f})q(\mathbf{g}) || p(\mathbf{f}, \mathbf{g}|\mathbf{y}))$, i.e., obtaining the best possible factorized approximation to the posterior, in the mentioned KL sense.

As it stands, F depends on two n-dimensional variational distributions. We can obtain a simpler, tighter bound by removing its dependence on one of them. We will refer to this new bound as the Marginalized Variational (MV) bound.

In order to optimally remove the dependence of $F(q(\mathbf{f}), q(\mathbf{g}))$ on $q(\mathbf{f})$, we compute the distribution $q^*(\mathbf{f})$ that maximizes $F(q(\mathbf{f}), q(\mathbf{g}))$ for a given $q(\mathbf{g})$ and insert it back into the bound. The variational Bayesian theory gives the optimal distribution $q^*(\mathbf{f})$ as

$$q^*(\mathbf{f}) = \underset{q(\mathbf{f})}{\operatorname{argmax}}\ F = \frac{p(\mathbf{f})}{Z(q(\mathbf{g}))} e^{\int q(\mathbf{g}) \log p(\mathbf{y}|\mathbf{f}, \mathbf{g}) \mathrm{d}\mathbf{g}}, \quad (2)$$

where $Z(q(\mathbf{g}))$ is the normalizing constant needed for $q^*(\mathbf{f})$ to integrate to one, i.e., $Z(q(\mathbf{g})) =$

$\int e^{\int q(\mathbf{g}) \log p(\mathbf{y}|\mathbf{f},\mathbf{g}) \mathrm{d}\mathbf{g}} p(\mathbf{f}) \mathrm{d}\mathbf{f}$. Inserting $q^*(\mathbf{f})$ (which of course depends on $q(\mathbf{g})$) back into the bound, we obtain, after some simplifications, the MV bound:

$$F(q(\mathbf{g})) = \log Z(q(\mathbf{g})) - \mathrm{KL}(q(\mathbf{g})||p(\mathbf{g})), \qquad (3)$$

where the dependence on $q(\mathbf{f})$ has been removed. The MV bound upper bounds the standard variational bound and (since it is particular case of it) also lower bounds the evidence. Hence, $\log p(\mathbf{y}) \geq F(q(\mathbf{g})) = F(q^*(\mathbf{f}), q(\mathbf{g})) \geq F(q(\mathbf{f}), q(\mathbf{g}))$. This is a general derivation, and holds for any variational bound depending on two independent distributions.

3.2. MV Bound for the HGP Model

For the HGP likelihood and priors, the MV bound can be computed in closed form if we restrict $q(\mathbf{g}) = \mathcal{N}(\mathbf{g}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$, i.e., to be a multivariate normal distribution. Note that we do not need to impose any constraint on $q(\mathbf{f})$ because the MV bound does not depend on it. Using $\mathbf{K}_f$ and $\mathbf{K}_g$ to name the covariance matrices resulting from evaluating the corresponding covariance functions at the inputs, (3) becomes

$$F(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \log \int e^{\int \mathcal{N}(\mathbf{g}|\boldsymbol{\mu},\boldsymbol{\Sigma}) \log p(\mathbf{y}|\mathbf{f},\mathbf{g}) \mathrm{d}\mathbf{g}} \mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{K}_f) \mathrm{d}\mathbf{f} - \mathrm{KL}(\mathcal{N}(\mathbf{g}|\boldsymbol{\mu}, \boldsymbol{\Sigma})||\mathcal{N}(\mathbf{g}|\mu_0\mathbf{1}, \mathbf{K}_g)).$$

The term inside the exponential is

$$\int \mathcal{N}(\mathbf{g}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \log p(\mathbf{y}|\mathbf{f}, \mathbf{g}) \mathrm{d}\mathbf{g} = \log \mathcal{N}(\mathbf{y}|\mathbf{f}, \mathbf{R}) - \frac{1}{4} \mathrm{tr}(\boldsymbol{\Sigma}) \qquad (4)$$

where $\mathbf{R}$ is a diagonal matrix with elements $[\mathbf{R}]_{ii} = e^{[\boldsymbol{\mu}]_i - [\boldsymbol{\Sigma}]_{ii}/2}$. Further, observing that $\int \mathcal{N}(\mathbf{y}|\mathbf{f}, \mathbf{R}) \mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{K}_f) \mathrm{d}\mathbf{f} = \mathcal{N}(\mathbf{y}|\mathbf{0}, \mathbf{K}_f + \mathbf{R})$, the MV bound has the following simple expression for HGP:

$$F(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \log \mathcal{N}(\mathbf{y}|\mathbf{0}, \mathbf{K}_f + \mathbf{R}) - \frac{1}{4} \mathrm{tr}(\boldsymbol{\Sigma}) - \mathrm{KL}(\mathcal{N}(\mathbf{g}|\boldsymbol{\mu}, \boldsymbol{\Sigma})||\mathcal{N}(\mathbf{g}|\mu_0\mathbf{1}, \mathbf{K}_g)). \qquad (5)$$

Observe the correspondence between the MV bound and the evidence of a standard, homoscedastic GP.

3.3. Reparametrization and Optimization

Bound (5) depends on $n + n(n+1)/2$ free variational parameters (defining $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$), i.e., the number of free parameters is quadratic in the number of observations. In this section we explain how to obtain an equivalent bound that depends on just n parameters. This is advantageous both from a computational point of view (reduced complexity) and from an optimization point of view (the optimization problem becomes easier and the interplay between variational parameters and hyperparameters is significantly reduced when both are jointly optimized).

By following a similar derivation to the one used in the Gaussian approximation (Opper & Archambeau, 2009), the stationary equations $\frac{\partial F(\boldsymbol{\mu},\boldsymbol{\Sigma})}{\partial \boldsymbol{\mu}} = 0$ and $\frac{\partial F(\boldsymbol{\mu},\boldsymbol{\Sigma})}{\partial \boldsymbol{\Sigma}} = 0$, which must be satisfied at any local or global maximum, reduce to the pair:

$$\boldsymbol{\mu} = \mathbf{K}_g(\boldsymbol{\Lambda} - \tfrac{1}{2}\mathbf{I})\mathbf{1} + \mu_0\mathbf{1}, \qquad \boldsymbol{\Sigma}^{-1} = \mathbf{K}_g^{-1} + \boldsymbol{\Lambda}, \quad (6)$$

for some positive semidefinite diagonal matrix $\boldsymbol{\Lambda}$ (see supplementary material for proof). So the above equation tells us that, at maxima, $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ both depend on some common diagonal matrix $\boldsymbol{\Lambda}$, which depends only on n diagonal elements. Therefore, we can reparametrize $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ according to Eq. (6) and use the n positive elements of $\boldsymbol{\Lambda}$ as the only free variational parameters[1].

Finally, the lower bound $F(\boldsymbol{\mu}(\boldsymbol{\Lambda}), \boldsymbol{\Sigma}(\boldsymbol{\Lambda})) = F(\boldsymbol{\Lambda})$ needs to be maximized w.r.t. the n variational parameters in $\boldsymbol{\Lambda}$. Such an optimization by construction minimizes the KL divergence between the approximate and exact posterior distribution. Simultaneously, we can maximize F w.r.t. the model hyperparameters $\boldsymbol{\theta}$, thus implementing Type-II Maximum Likelihood (ML-II) for model selection. The whole optimization is nonlinear and gradient-based procedures, such as conjugate gradient, can be used. The derivatives w.r.t. $(\boldsymbol{\Lambda}, \boldsymbol{\theta})$ can be computed analytically and efficiently.

The variational bound and its complete gradient can be computed in $\mathcal{O}(n^3)$ time. This is the same cost of a standard GP. In practice, computing the VHGP bound and its gradient takes roughly twice the time required to compute the evidence and its derivatives in a standard homoscedastic GP.

4. Predictive Distribution

The predictive distribution for a new test output y_* (corresponding to input $\mathbf{x}_*$) given training data can be expressed as $p(y_*|\mathbf{x}_*, \mathcal{D})$. Though it cannot be computed in closed form, if we regard $q^*(\mathbf{f})q(\mathbf{g})$ as a good approximation to $p(\mathbf{f}, \mathbf{g}|\mathcal{D})$, its mean and variance are analytically tractable.

First, we need an explicit expression for $q^*(\mathbf{f})$. Inserting (4) in (2) we get $q^*(\mathbf{f}) \propto \mathcal{N}(\mathbf{y}|\mathbf{f}, \mathbf{R})p(\mathbf{f})$, so

$$q^*(\mathbf{f}) = \mathcal{N}(\mathbf{f}|\mathbf{K}_f\boldsymbol{\alpha}, \mathbf{K}_f - \mathbf{K}_f(\mathbf{K}_f + \mathbf{R})^{-1}\mathbf{K}_f).$$

[1] This result is related to (Opper & Archambeau, 2009). Their general method results in $2n$ free parameters. However, in our case the stationary equations have additional structure that allows the reduction to n free parameters.

where $\boldsymbol{\alpha} = (\mathbf{K}_f + \mathbf{R})^{-1}\mathbf{y}$. The posterior distribution for $f_* = f(\mathbf{x}_*)$ under the variational approximation is

$$q(f_*) = \int p(f_*|\mathbf{x}_*, \mathbf{X}, \mathbf{f}) q^*(\mathbf{f}) \mathrm{d}\mathbf{f} = \mathcal{N}(f_*|a_*, c_*^2),$$

with $a_* = \mathbf{k}_{f*}^\top \boldsymbol{\alpha}$ and $c_*^2 = k_{f**} - \mathbf{k}_{f*}^\top (\mathbf{K}_f + \mathbf{R})^{-1} \mathbf{k}_{f*}$.

The posterior distribution of $g_* = g(\mathbf{x}_*)$ under the variational approximation is

$$q(g_*) = \int p(g_*|\mathbf{x}_*, \mathbf{X}, \mathbf{g}) q(\mathbf{g}) \mathrm{d}\mathbf{g} = \mathcal{N}(g_*|\mu_*, \sigma_*^2),$$

with $\mu_* = \mathbf{k}_{g*}^\top (\boldsymbol{\Lambda} - \frac{1}{2}\mathbf{I})\mathbf{1} + \mu_0$ and $\sigma_*^2 = k_{g**} - \mathbf{k}_{g*}^\top (\mathbf{K}_g + \boldsymbol{\Lambda}^{-1})^{-1} \mathbf{k}_{g*}$. Based on those, the predictive distribution for a new observation y_* becomes

$$\begin{aligned} q(y_*) &= \int\int p(y_*|g_*, f_*) q(f_*) q(g_*) \mathrm{d}f_* \mathrm{d}g_* \\ &= \int \mathcal{N}(y_*|a_*, c_*^2 + e^{g_*}) \mathcal{N}(g_*|\mu_*, \sigma_*^2) \mathrm{d}g_*, \end{aligned}$$

which is not analytically tractable. However, its value can be approximated up to several digits using inexpensive Gauss-Hermite quadrature. Its mean and variance, on the other hand, can be computed analytically: $\mathbb{E}_q[y_*|\mathbf{x}_*, \mathcal{D}] = a_*$ and $\mathbb{V}_q[y_*|\mathbf{x}_*, \mathcal{D}] = c_*^2 + e^{\mu_* + \sigma_*^2/2}$. Note that the predictive density is not Gaussian.

5. Most Likely HGP (MLHGP) and Max. A Posteriori HGP (MAPHGP)

MLHGP was recently introduced in (Kersting et al., 2007). In this work, the full posterior over $\mathbf{g}$ is replaced by a point estimation and an iterative algorithm to perform that estimation is provided. As the authors point out, the algorithm is not guaranteed to converge and may instead oscillate. Furthermore, it may require many iterations (each one requiring to train two standard GPs) before stabilizing. In a related, more recent work (Quadrianto et al., 2009), these issues are addressed by choosing $\mathbf{g}$ so as to maximize a penalized likelihood, equivalent up to a constant to $p(\mathbf{g}|\mathcal{D})$, thus introducing the MAPHGP approximation. Notice that this method provides a point estimate for $\mathbf{g}$, while our method is a more fully Bayesian approach that variationally integrates out $\mathbf{g}$.

Code implementing MAPHGP was kindly provided by the authors. We compare it with VHGP in Sections 6.1 and 6.2.1, using the authors' initializations. We found that the quality of the solution was highly dependent on the number and location of the latent noise variables described in (Quadrianto et al., 2009). We set them to 10 as the authors did, since for higher values MAPHGP overfit severely. In contrast, VHGP is not vulnerable to such overfitting: The number of hyperparameters specifying the model is very small, and optimizing the n variational parameters, implies better approximating the exact posterior. In our experience, VHGP is more robust than MAPHGP and allows for smoother optimization, being less likely to get stuck in bad local minima than MAPHGP is.

6. Experiments

In this section we will assess the accuracy of the VHGP and MAPHGP approximations by comparing their posteriors with MCMC; we will also compare their predictive performance in terms of Normalized Mean Square Error (NMSE) and Negative Log-Probability Density (NLPD) with the homoscedastic GP (which will refer to simply as GP); and finally apply VHGP to the volatility prediction problem. For all problems, we assume the outputs to be zero-mean.

We will be using the Automatic Relevance Determination Squared Exponential (ARD-SE) kernel, defined as $k_{\mathrm{ARDSE}}(\mathbf{x}, \mathbf{x}') = \sigma_0^2 \exp\left(-\frac{1}{2}\sum_{d=1}^{D} \frac{([\mathbf{x}]_d - [\mathbf{x}']_d)^2}{\ell_d}\right)$.

6.1. Assessing the quality of the approximation using MCMC

In order to obtain an accurate posterior for the heteroscedastic GP regression model, we use MCMC. The recently proposed elliptical slice sampling (Murray et al., 2010) was used to draw posterior samples from $p(\mathbf{g}|\mathbf{y})$ while $\mathbf{f}$ was integrated out analytically. More specifically, the joint probability density function with $\mathbf{f}$ marginalized out is

$$p(\mathbf{y}, \mathbf{g}) = N\left(\mathbf{y}|\mathbf{0}, \mathbf{K}_f + \mathrm{diag}(e^{\mathbf{g}})\right) N(\mathbf{g}|\mu_0\mathbf{1}, \mathbf{K}_g).$$

Notice the similarity of this expression with the "marginalized" variational lower bound (5), in which distribution $q(\mathbf{f})$ had been optimally removed.

We build a one-dimensional toy dataset according to the proposed heteroscedastic model described in Section 2, generating 100 samples in the range $x \in [-1, 1]$. We select the covariance function for $f(x)$ to be the SE with parameters $\ell = 0.5$, $\sigma_0^2 = 2$, and the covariance function for $g(x)$ to be another SE with parameters $\ell = 0.5$, $\sigma_0^2 = 1$ plus noise of power $\sigma^2 = 0.25$.

Inference is then performed using MAPHGP, VHGP and MCMC, fixing the hyperparameters to the known true values. This removes the effect of hyperparameter learning, so that we can assess whether MAPHGP and VHGP predictions are close to the asymptotically unbiased MCMC estimates, for the same set of hyperparameters. We emphasize that MCMC is much

slower than variational inference: With just 100 data points, VHGP takes 4 s and MAPHGP takes 3 s, but MCMC takes 400 s. The $\mathcal{O}(n^3)$ time scaling of the learning process renders MCMC impractical even for moderate-size datasets.

As shown in Fig. 1.a, the approximate posterior provided by VHGP is very close to the exact solution provided by MCMC, both for $y(x)$ and $g(x)$. Differences between the exact and the approximate methods are of course more noticeable in the posterior over $g(x)$ (Fig. 1.b), since for $y(x)$, integration over $g(x)$ smooths out the differences to some extent. MAPHGP produces clearly worse results, as could be expected. Also, its approximation for $g(x)$ is deterministic, instead of a full distribution.

We can further analyze the behavior of VHGP by looking at the full predictive posterior. The marginal posterior at $x = 0.9$ (marked with a vertical line in Fig. 1.a) is plotted in Fig. 1.c. The exact posterior produced by MCMC is almost perfectly matched by the leptokurtic VHGP posterior in red, dashed line. Such a good match could not be achieved by a Gaussian posterior, as depicted by the matching-moments Gaussian plotted in red, dotted line. Recall that MLHGP, MAPHGP and most previous approximations use a Gaussian to approximate the posterior.

The accuracy of VHGP's hyperparameter learning is of course not reflected in these experiments; it will be discussed in Section 6.3.2.

6.2. Regression performance

We will now assess the performance of VHGP on several synthetic and real datasets. We use an ARD SE covariance function for $f(x)$ and an ARD SE covariance function plus noise of power σ^2 for $g(x)$. In order to initialize hyperparameters, we first run a homoscedastic GP with an ARD SE plus noise covariance function. For $f(x)$, we set $\{\ell_d^{\text{VHGP}} = \ell_d^{\text{GP}}\}_{d=1}^D$, $\sigma_0^{\text{VHGP}} = \sigma_0^{\text{GP}}$. For $g(x)$, we set $\{\ell_d^{\text{VHGP}} = \ell_d^{\text{GP}}\}_{d=1}^D$, $\sigma_0^{\text{VHGP}} = 1$, $\sigma^{\text{VHGP}} = 1/2$, $\mu_0 = 2\log(\sigma^{\text{GP}}) - 1/2$.

Throughout, we will use the NMSE $= \frac{\sum_{j=1}^{n_*}(y_{*j}-\hat{y}_{*j})^2}{\sum_{j=1}^{n_*}(y_{*j}-\overline{y})^2}$ and the NLPD $= -\frac{1}{n_*}\sum_{j=1}^{n_*}\log p(y_{*j}|\mathcal{D})$ as performance measures. Here, y_{*j} is the j-th observation within the test set, $\hat{y}_{*j}$ is the mean of the posterior for that observation, n_* the number of test observations and $\overline{y}$ the mean of the training observations.

6.2.1. One-dimensional datasets

We will first consider four regression datasets that have traditionally been used to assess heteroscedastic regression, see e.g. (Kersting et al., 2007):

G. The synthetic dataset from (Goldberg et al., 1998), consisting of 100 data points. Inputs are uniformly spaced in the $[0, 1]$ range and outputs are generated as $2\sin 2x$ plus Gaussian noise, with standard deviation linearly increasing from 0.5 at $x = 0$ to 1.5 at $x = 1$.

C. The synthetic dataset from (Cawley et al., 2006), consisting of 100 data points. Inputs are uniformly spaced in the $[-1, 1]$ range and outputs are the Heaviside function of the inputs plus Gaussian noise of standard deviation 0.1. This dataset is not truly heteroscedastic, but the steep change at $x = 0$ can be better modeled using a locally higher noise level.

M. The motorcycle dataset from (Silverman, 1985), consisting of 133 of accelerometer readings through time following a simulated motorcycle crash during an experiment to determine the efficacy of crash-helmets.

T. The 1D toy dataset, consisting of 100 data points, that has already been described in Section 6.1.

Each dataset is generated according to its description, then a random split is performed, using 90% of the samples for training and 10% of the samples for testing. This procedure is repeated 300 times. Table 1 shows average results for GP, MAPHGP and VHGP.

The three models perform similarly in terms of NMSE, indicating that for these datasets considering the heteroscedastic noise produces little improvement in the predictive mean. However, the heteroscedastic models produce more accurate posterior distributions and this shows as a clear improvement on the NLPD measure. VHGP is equivalent or superior to MAPHGP in all datasets, as it was expected, and incurs in a smaller standard deviation in the results. Analogous results were observed on other datasets not reported here.

6.2.2. Large, multi-dimensional datasets

In order to show how VHGP can handle large multivariate datasets[2], we tested it on: Abalone[3], Pole Telecommunications[4] and Elevators[5]. Results are compared with an homoscedastic GP in Table 2.

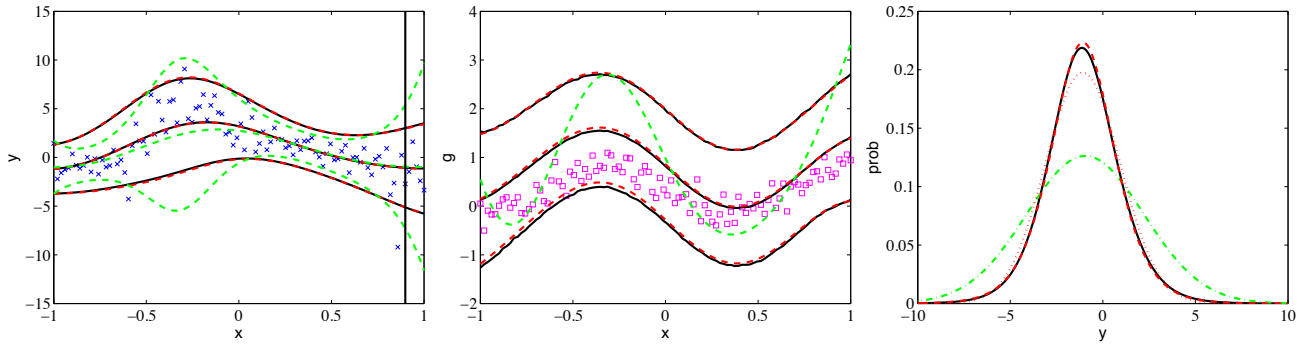
Consistent with previous results, VHGP outperforms the standard GP in terms of test NLPD, showing its superior ability to model these datasets. NMSE results are comparable between both methods, except for Pole T. For this problem, VHGP's residual distribution is

[2] Available at http://www.liaad.up.pt/~ltorgo/Regression/DataSets.html

[3] 3133 training/1044 testing samples, 8 attributes.

[4] 3000 training/12000 testing samples, 26 attributes.

[5] 3000 training/13599 testing samples, 17 attributes.



(a) Posterior over $y \pm 2$ std. deviations (b) Posterior over $g \pm 2$ std. deviations (c) Predictive distribution y at $x = 0.9$

Figure 1. Comparison of posterior distributions for MAPHGP (green, dash-dotted line), VHGP (red, dashed line) and the golden standard MCMC (black, continuous line). All methods use the same set of hyperparameters.

highly non-Gaussian, with most deviations being very small and a few being very big. If instead of the *mean* square error we computed the *median* square error, we would obtain 2.151 for the GP and a much smaller 0.269 for the VHGP.

Table 1. Average results including one standard deviation. Statistically significant differences w.r.t. GP are marked as •, whereas statistically significant differences w.r.t. MAPHGP are marked as ⋄. Significance is measured according to the t-test at the 5% significance level. Results correspond to averaging over 300 independent splits.

Problem	GP	MAPHGP	VHGP	
G. (NMSE)	0.40±0.21	0.39±0.21	0.39±0.21	
G. (NLPD)	1.51±0.28	1.53±0.44	1.45±0.28	•⋄
C. (NMSE)	0.08±0.06	0.11±0.08	• 0.10±0.07	•⋄
C. (NLPD)	-0.44±0.52	-0.44±0.61	-0.59±0.31	•⋄
M. (NMSE)	0.26±0.18	0.26±0.17	0.26±0.17	
M. (NLPD)	4.59±0.22	4.32±0.60	• 4.32±0.30	•
T. (NMSE)	0.78±0.33	0.77±0.33	0.77±0.32	
T. (NLPD)	2.22±1.16	2.10±1.15	1.91±0.97	•⋄

Table 2. Performance of GP and VHGP on several large, multidimensional problems. See text for a description.

Problem	GP	VHGP
Abalone. (NMSE)	0.4359	0.4259
Abalone (NLPD)	2.1265	2.0130
Pole T. (NMSE)	0.0237	0.0934
Pole T. (NLPD)	2.9082	1.8047
Elevators (NMSE)	0.0905	0.0939
Elevators (NLPD)	-4.7997	-4.8450

6.3. Application to volatility forecasting

Heteroscedastic GP regression is naturally suited to the problem of volatility modeling and forecasting. In financial time series, volatility is defined as the standard deviation of a return series at time instant[6] x given all the information available at time instant $x-1$. Return series can be obtained from price series $\mathrm{p}(x)$ as $y(x) = \log(\mathrm{p}(x)) - \log(\mathrm{p}(x-1))$. Then, pairs $(x, y(x))$ constitute a dataset in which the noise level (i.e., the volatility) changes over time and VHGP can be applied to estimate historical volatility or make forecasts.

To remain consistent with the existing literature, we make the usual assumption of considering the return series as a zero-mean noise-only process, i.e. set $f(x) = 0$ and also assume that time instants are discrete, integer intervals, such as days. Note, however that VHGP can seamlessly deal with variable sample rates, make forecasts for fractional intervals and even learn a non-zero $f(x)$, if a model for its covariance is available[7].

In the following, we will describe one of the volatility models to which VHGP can be applied to perform approximate inference, evaluate the quality of the inference as compared to MCMC and further make a test on real data, comparing it with the GARCH model.

6.3.1. Volatility model

It turns out that the volatility model proposed in (Liu, 2001), though stated differently, corresponds exactly to the VHGP model when the latent function is set to zero (i.e. $k_f(\mathbf{x}, \mathbf{x}') = 0$), the covariance function of $g(x)$ is $k_g(x, x') = \sigma_0^2/(1-\phi^2)\phi^{|x-x'|}$, $\mu_0 = 2\log\beta$, and x is restricted to be an integer. Thus, x represents the time instants in which the returns are observed. Note that $k_g(x, x')$ is a reparametrization of

[6] For consistency with our previous notation, we will use x to denote time.

[7] This can be used to model drifts in the mean of the return series.

the standard Ornstein-Uhlenbeck covariance function, and in the case in which x is restricted to be an integer, reduces $g(x)$ to an AR(1) process. This is a first-order Markov process, that, for ordered time instants, results in $\mathbf{K}_g^{-1}$ being tridiagonal rather than a full matrix. It is possible to exploit this fact to compute the variational bound and all its derivatives on $\mathcal{O}(n)$ time, thus rendering the whole learning process of VHGP *linear* in the number of training samples.

6.3.2. Accuracy of VHGP volatility model

In the recent work of (Girolami & Calderhead, 2011), an efficient technique to perform MCMC called Riemann Manifold Hamiltonian Monte Carlo (RMHMC) is introduced. In their experiments, inference for the precise model described in Section 6.3.1 is considered, though hyperparameters σ_0, ϕ and β are not fixed, but integrated over. Furthermore, they also exploit the tridiagonal property of $\mathbf{K}_g^{-1}$ to increase the efficiency of their algorithm, so that RMHMC can be applied to model volatility for relatively large datasets. We were kindly provided with their original code and datasets, so that we can compare the accuracy of VHGP with an asymptotically exact, fully Bayesian model.

Girolami and Calderhead generated a synthetic dataset of 2000 data points according to the mentioned model, using parameters $\sigma_0 = 0.15$, $\phi = 0.98$, $\beta = 0.65$. Using their dataset, we computed the approximate posterior over $g(x)$ using VHGP, while jointly learning the hyperparameters (i.e., using ML-II to estimate them). Those were initialized as in their work: $\sigma_0 = 0.5$, $\phi = 0.5$, $\beta = 0.5$.

Girolami and Calderhead obtain a reasonably peaked posterior for the hyperparameters, with posterior means $\mathbb{E}[\sigma_0|\mathcal{D}] = 0.1714$, $\mathbb{E}[\phi|\mathcal{D}] = 0.9771$, $\mathbb{E}[\beta|\mathcal{D}] = 0.6654$, quite close to the above mentioned ground truth values. In turn, VHGP provides the following ML-II estimates: $\sigma_0 = 0.1483$, $\phi = 0.9814$, $\beta = 0.6662$, which are also very accurate. The posterior over $g(x)$ produced by VHGP is plotted in Fig. 2 (top) together with that of RMHMC. We observe very good agreement between both, with RMHMC having a slightly higher predictive variance, due to the effect of integrating over the hyperparameters. To verify this latter claim, we also run MCMC fixing the hyperparameters to those obtained by VHGP. Results are plotted in Fig. 2 (bottom). In this case the agreement between the exact posterior and VHGP is almost exact!

Even though RMHMC was specifically tuned to exploit the tridiagonal inverse matrices involved in this model, the speed difference with VHGP was significant: RMHMC took 300 s to process the 2000 data points, whereas VHGP took less than 5 s (including hyperparameter selection). This means that VHGP is a very fast, while still very accurate, alternative to MCMC to make inference under this volatility model.

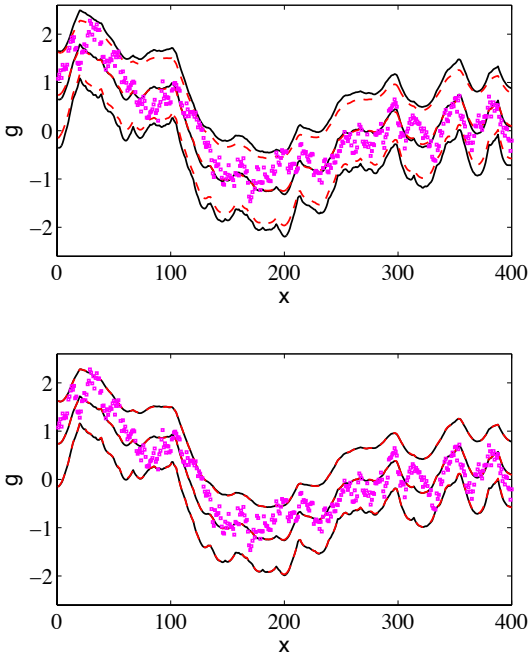


Figure 2. Posterior over g for the toy volatility dataset using VHGP (red, dashed line) vs. RMHMC (black, continuous line). RMHMC can integrate over hyperparameters (top) or fix them to VHGP ML-II values (bottom). Pink squares show ground truth values for $g(x)$, not available to the methods. Only first 400 points are shown for clarity.

6.3.3. Volatility forecasting

After validating VHGP as an accurate approximation to the exact posterior for this volatility model, in this section we will test its actual forecasting ability.

We used the return series of the daily exchange rate between the Deutschmark (DEM) and the Great Britain Pound (GBP), from Jan 1984 to Jan 1992 (totaling 1974 trading days). This series has become a standard to assess the performance of volatility prediction systems (McCullough & Renfro, 1998; Brooks et al., 2001; Wilson & Ghahramani, 2010). As a benchmark for comparison, we use GARCH(1,1), whose performance has been reported to be very competitive for this task (Hansen & Lunde, 2005). GARCH(1,1) models $y(x) \sim \mathcal{N}(y|0, r(x))$, with $r(x) = a_0 + a_1 y^2(x-1) + b_1 r(x-1)$, for some a_0, a_1 and b_1. These values are obtained by constrained maximum likelihood on the training set.

Following (Wilson & Ghahramani, 2010), we use a rolling window of the previous 120 days of returns to

make 1, 7, and 30 day ahead volatility forecasts and retrain the model every 7 days. Predictions are made for the last 1825 trading days of this series.

As a performance measure, we use the MSE between the predicted volatility and the squared returns. This is one of the few consistent ways to measure volatility, as discussed in (Brownlees et al., 2009).

Table 3. MSE for three different forecast horizons, using GARCH and VHGP.

Method	Days ahead ($\times 10^{-9}$)		
	1	7	30
GARCH(1,1)	3.092	3.312	5.043
VHGP	3.087	3.092	3.118

We see that VHGP was slightly superior to GARCH(1,1) in this time series, with an increased advantage being obtained for distant forecast horizons. This is consistent with the result reported by (Wilson & Ghahramani, 2010) for their copula processes.

7. Discussion and Future Work

In this work we have provided a theoretically well-founded approximation that enables accurate inference in heteroscedastic GPs with a comparable cost to that of standard, analytically tractable homoscedastic GPs. In order to do this, the well-known variational approximation has been used, but introducing non-standard modifications: $q(\mathbf{f})$ has been "marginalized" out from the bound and the number of variational parameters has been drastically reduced by exploiting the relations appearing in this model at the local optima.

We think this is a promising line of work, since the homoscedasticity assumption may be too strong for many real problems. It is relatively straightforward to extend this variational method to handle classification or to obtain sparse heteroscedastic GPs.

It also turns out that by selecting $k_g(\mathbf{x}, \mathbf{x}') = \sigma_0^2 \delta_{\mathbf{x}\mathbf{x}'}$, a leptokurtic i.i.d. prior is induced in noise and thus VHGP can be directly used for robust regression, a matter that deserves further investigation.

Acknowledgments

We thank the reviewers for insightful comments. MLG was supported by MICINN CONSOLIDER-INGENIO project CSD2008-00010 (COMONSENS). MKT was supported by EPSRC Grant No EP/F005687/1 "Gaussian Processes for Systems Identification with Applications in Systems Biology".

References

Brooks, C., Burke, S.P., and Persand, G. Benchmarks and the accuracy of GARCH model estimation. *International Journal of Forecasting*, 17:45–56, 2001.

Brownlees, C.T., Engle, R.F., and Kelly, B.T. A practical guide to volatility forecasting through calm and storm, 2009. URL http://ssrn.com/abstract=1502915.

Cawley, G., Talbot, N., and Chapelle, O. Estimating predictive variances with kernel ridge regression. In *Machine Learning Challenges*, pp. 56–77, 2006.

Girolami, M. and Calderhead, B. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society*, 2011. In press.

Goldberg, P., Williams, C., and Bishop, C. Regression with input-dependent noise: A Gaussian process treatment. In *Advances in NIPS*, 1998.

Hansen, P. R. and Lunde, A. A forecast comparison of volatility models: Does anything beat a GARCH(1,1). *Journal of Applied Econometrics*, 20(7):873–889, 2005.

Honkela, A., Raiko, T., Kuusela, M., Tornio, M., and Karhunen, J. Approximate Riemannian conjugate gradient learning for fixed-form variational Bayes. *Journal of Machine Learning Research*, 11:3235–3268, 2011.

Kersting, K., Plagemann, C., Pfaff, P., and Burgard, W. Most likely heteroscedastic Gaussian processes regression. In *Proc. of the ICML*, pp. 393–400, 2007.

Liu, J.S. *Monte Carlo Strategies in Scientific Computing*. New York: Springer, 2001.

McCullough, B.D. and Renfro, C.G. Benchmarks and software standards: A case study of GARCH procedures. *Journal of Economic and Social Measurement*, 25:59–71, 1998.

Murray, I., Adams, R.P., and MacKay, D.J.C. Elliptical slice sampling. In *AISTATS 13*, volume 9 of *JMLR: W&CP*, pp. 541–548, 2010.

Nickisch, H. and Rasmussen, C.E. Approximations for binary Gaussian process classification. *Journal of Machine Learning Research*, 9:2035–2078, 2008.

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

Quadrianto, N., Kersting, K., Reid, M., Caetano, T., and Buntine, W. Kernel conditional quantile estimation via reduction revisited. In *Proc. of the 9th IEEE International Conference on Data Mining*, 2009.

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

Silverman, B.W. Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society*, 47:1–52, 1985.

Wilson, A. and Ghahramani, Z. Copula processes. In *Advances in NIPS 23*, pp. 2460–2468, 2010.