# Efficient Variational Inference for Gaussian Process Regression Networks

**Trung V. Nguyen**
Australian National University & NICTA

**Edwin V. Bonilla**
NICTA & Australian National University

## Abstract

In multi-output regression applications the correlations between the response variables may vary with the input space and can be highly non-linear. Gaussian process regression networks (GPRNs) are flexible and effective models to represent such complex adaptive output dependencies. However, inference in GPRNs is intractable. In this paper we propose two efficient variational inference methods for GPRNs. The first method, GPRN-MF, adopts a mean-field approach with full Gaussians over the GPRN's parameters as its factorizing distributions. The second method, GPRN-NPV, uses a nonparametric variational inference approach. We derive analytical forms for the evidence lower bound on both methods, which we use to learn the variational parameters and the hyper-parameters of the GPRN model. We obtain closed-form updates for the parameters of GPRN-MF and show that, while having relatively complex approximate posterior distributions, our approximate methods require the estimation of $\mathcal{O}(N)$ variational parameters rather than $\mathcal{O}(N^2)$ for the parameters' covariances. Our experiments on real data sets show that GPRN-NPV may give a better approximation to the posterior distribution compared to GPRN-MF, in terms of both predictive performance and stability.

## 1 Introduction

Regression with multiple outputs is an important problem in machine learning and has received considerable attention in the last few years (Bonilla et al., 2008; Teh et al., 2005; Boyle and Frean, 2005; Alvarez and Lawrence, 2009; Wilson et al., 2012). The challenge here is to develop flexible models able to capture the dependencies between the outputs, while having efficient inference methods that allow us to generalize well on unseen data.

The applications of multi-output regression are widespread including geostatistics, biology and financial applications. For example, in geostatistics, it has been shown that it is possible to leverage the relationships between different metal concentrations in a particular region in order to predict the concentration of another metal, for which only very sparse observations are available (see e.g. Goovaerts, 1997).

While various non-probabilistic approaches have been developed to address structured prediction problems, in many of these applications it is crucial to have full posterior probabilities over the predicted outputs, for example in order to use Bayesian decision-theoretic criteria for risk minimization or for active sampling.

Within the Bayesian formalism for regression problems, Gaussian processes have proved very effective tools for single and multiple output scenarios (Rasmussen and Williams, 2006). However, most popular GP-based methods to multiple output regression problems assume that the dependencies between the outputs are fixed, i.e. they are independent of the input space (see e.g. Bonilla et al., 2008; Teh et al., 2005). For problems such as the metal concentration prediction mentioned above, such an assumption may be too strong as the correlations between the different metals can vary according to their spatial locations. For example, they may be different due to distinct rock types or due to the presence of a geological fault.

Wilson et al. (2012) have recently proposed a general framework for multi-output regression where the correlations between the outputs can be spatially *adaptive*. Their method is called Gaussian process regression networks (GPRNs) and it is fundamentally an adaptive linear combination of latent Gaussian processes,

Appearing in Proceedings of the $16^{th}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2013, Scottsdale, AZ, USA. Volume 31 of JMLR: W&CP 31. Copyright 2013 by the authors.

where the weights of the linear combination are also Gaussian processes.

This paper proposes efficient approximate inference methods for GPRNs. These methods are underpinned by variational inference principles, and differ on their approximating variational distributions. The first method is a simple mean-field approach where we use a factorized distribution over the parameters of the GPRN. Each of the factor distributions is a full Gaussian. For this method we show that: (a) we can obtain an analytical expression for the evidence lower bound and closed-form updates for the variational parameters; and (b) we can parametrize the corresponding covariances with only $\mathcal{O}(N)$ parameters, instead of $\mathcal{O}(N(N+1)/2)$, where $N$ is the number of data-points. We refer to this method as the mean-field GPRN.

The second method exploits recent advances in variational inference. In particular, it builds upon the nonparametric variational inference of Gershman et al. (2012) to approximate the posterior distribution of the GPRN's parameters with a mixture of isotropic Gaussians. The advantage of this method over mean-field approaches is that it can approximate relatively complex distributions, which are not necessarily well approximated by a single Gaussian. As each covariance is isotropic, it only needs $\mathcal{O}(N)$ variational parameters. For this method we obtain an analytical solution for the expected log joint, which leads to a tight bound for the evidence lower bound. We note that, the original method of Gershman et al. (2012), uses second-order approximations at the expense of not having a proper evidence lower bound. This is not the case in our approach to non-parametric variational inference for GPRNs.

The remainder of this paper is organized as follows. In section 2 we describe the Gaussian process regression networks and its main inference task. We then introduce two efficient variational inference methods for approximating the posterior distribution of the GPRN model. In section 4 we assess the predictive performances and computational behaviors of the proposed methods on a geostatistic dataset and a high-performance concrete dataset. Finally, related work is discussed in section 5.

## 2 Gaussian Process Regression Networks

Here we describe the Gaussian process regression network (GPRN) model of Wilson et al. (2012) and explain the main inference task in GPRNs, mainly posterior inference over the parameters of the model.

Let $\mathbf{y}(\mathbf{x}) \in \mathbb{R}^P$ be a vector-valued function of $\mathbf{x} \in \mathbb{R}^D$, where $P$ and $D$ are the dimensionality of the output and input spaces respectively. In the GPRN model, our observations $\mathbf{y}(\mathbf{x})$ are assumed to be linear combinations of $Q$ noisy latent functions, $\mathbf{f}(\mathbf{x}) \in \mathbb{R}^Q$, corrupted by Gaussian noise. The distinctive feature of the GPRN model is that the coefficients $\mathbf{W}(\mathbf{x}) \in \mathbb{R}^P \times \mathbb{R}^Q$ of the linear combination of the latent functions are stochastic and so are the latent functions $\mathbf{f}(\mathbf{x})$. Thus, the GPRN model is defined as follows:

$$\mathbf{y}(\mathbf{x}) = \mathbf{W}(\mathbf{x})[\mathbf{f}(\mathbf{x}) + \sigma_f \boldsymbol{\epsilon}] + \sigma_y \mathbf{z}, \tag{1}$$

$$f_j(\mathbf{x}) \sim \mathcal{GP}(0, \kappa_f), \quad j = 1 \ldots Q, \tag{2}$$

$$W_{ij}(\mathbf{x}) \sim \mathcal{GP}(0, \kappa_w), \quad i = 1, \ldots, P; j = 1, \ldots Q, \tag{3}$$

$$\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\epsilon}; \mathbf{0}, \mathbf{I}_Q), \tag{4}$$

$$\mathbf{z} \sim \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I}_P), \tag{5}$$

where each $f_j$ is independently sampled from a Gaussian process (GP) with covariance function $\kappa_f$ and each $W_{ij}$ is also independently sampled from a GP with covariance $\kappa_w$. Here $\mathbf{I}_P$ and $\mathbf{I}_Q$ denote the identity matrices of dimension $P$ and $Q$ respectively. Although not necessary in a general GPRN model, we have constrained all latent function values to share the covariance function (and its parameters) and similarly for the weights.

One of the main advantages of the GPRN model is that the dependencies of the outputs $\mathbf{y}$ are (efficiently) induced via the latent functions $\mathbf{f}$. More importantly, as the mixing coefficients $\mathbf{W}(\mathbf{x})$ explicitly depend on the input $\mathbf{x}$, these correlations are spatially *adaptive*.

Let $\mathcal{X} = \{(\mathbf{x}_i)\}_{i=1}^N$ be the set of observed inputs and $\mathcal{D} = \{(\mathbf{y}_i)\}_{i=1}^N$ be the set of observed outputs. We denote $\mathbf{u}$ as the concatenation of the latent function parameters and the weights, i.e. $\mathbf{u} = (\hat{\mathbf{f}}, \mathbf{w})$, evaluated at all training points $\mathbf{x} \in \mathcal{X}$. Here $\hat{\mathbf{f}}$ is the noisy version of the latent function values, i.e. $\hat{\mathbf{f}} = \mathbf{f} + \sigma_f \boldsymbol{\epsilon}$, and $\mathbf{w} = \operatorname{vec} \mathbf{W}$, with vec being the Vec operator. Let us refer to $\boldsymbol{\theta} = \{\boldsymbol{\theta}_f, \boldsymbol{\theta}_w, \sigma_f, \sigma_y\}$ as the hyper-parameters of the GPRN, where $\boldsymbol{\theta}_f$ and $\boldsymbol{\theta}_w$ are the parameters of the covariances $\kappa_f$ and $\kappa_w$ respectively.

As defined by equations (2), (3) and (4), the prior over $\mathbf{u}$ evaluated at the training points is a $NQ(P+1)$-dimensional Gaussian with block diagonal covariance:

$$p(\mathbf{u}|\boldsymbol{\theta}_f, \boldsymbol{\theta}_w, \sigma_f) = \mathcal{N}(\mathbf{u}; \mathbf{0}, \mathbf{C}_u), \tag{6}$$

where the first $Q$ blocks of $\mathbf{C}_u$ are induced by the covariance function $\kappa_f$ and the last $PQ$ blocks are induced by $\kappa_w$. To keep the notation simple, we omit the training inputs $\mathcal{X}$ from the conditioning sets in the above equation and in the rest of this paper.

Given the parameters $\mathbf{u}$ and the hyper-parameters $\boldsymbol{\theta}$, the conditional likelihood given by Equations (1) and

(5) evaluated at the targets $\mathcal{D}$ can be written as:

$$p(\mathcal{D}|\mathbf{u},\boldsymbol{\theta}) = \prod_{n=1}^{N} \mathcal{N}\left(\mathbf{y}(\mathbf{x}_n); \mathbf{W}(\mathbf{x}_n)\hat{\mathbf{f}}(\mathbf{x}_n), \sigma_y^2 \mathbf{I}_P\right). \quad (7)$$

Hence, our main inference task in GPRN is to compute the posterior:

$$p(\mathbf{u}|\mathcal{D},\boldsymbol{\theta}) \propto p(\mathbf{u}|\boldsymbol{\theta}_f, \boldsymbol{\theta}_w, \sigma_f) p(\mathcal{D}|\mathbf{u}, \sigma_y), \quad (8)$$

which is intractable in general. In the next section we propose methods that approximate this posterior using variational inference.

## 3 Variational Inference for GPRNs

This section describes our inference methods to approximate the posterior $p(\mathbf{u}|\mathcal{D},\boldsymbol{\theta})$ using variational inference (Jordan et al., 1999). Our goal is to find an approximating distribution $q(\mathbf{u})$ from a restricted family of distributions that is closest to the true posterior with respect to the KL divergence:

$$\mathrm{KL}\big(q(\mathbf{u}) \parallel p(\mathbf{u}|\mathcal{D})\big) = \mathbb{E}_q\left[\log \frac{q(\mathbf{u})}{p(\mathbf{u}|\mathcal{D})}\right], \quad (9)$$

where, for simplicity in the notation, we have omitted the dependency of the posterior on the hyper-parameters $\boldsymbol{\theta}$. However, as we shall see later, the hyper-parameters of the GPRN model can be learned under the same variational framework.

Minimizing the KL divergence in Equation (9) is equivalent to maximizing the evidence lower bound, which for the GPRN is given by:

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(\mathcal{D}|\mathbf{f},\mathbf{w})] + \mathbb{E}_q[\log p(\mathbf{f},\mathbf{w})] + \mathcal{H}_q[q(\mathbf{f},\mathbf{w})], \quad (10)$$

where $\mathcal{H}_q$ is the entropy of $q(\mathbf{f},\mathbf{w})$.

In the following sections, we propose two approximating distributions for GPRN posterior inference: a factorized distribution and a mixture distribution. They give rise to the mean-field method and to the nonparametric variational inference method respectively.

We will show that for the mean-field method: (a) we can obtain an analytical expression for the evidence lower bound and closed-form updates for the variational parameters; and (b) we can parametrize the corresponding covariances with only $\mathcal{O}(N)$ parameters, instead of $\mathcal{O}(N(N+1)/2)$, where $N$ is the number of data-points. We refer to this method as the mean-field GPRN.

Additionally, for the nonparametric variational method we will show that we can obtain an analytical solution for the expected log joint, which leads to a proper evidence lower bound and that, as the mean-field approach, it only requires $\mathcal{O}(N)$ variational parameters for each corresponding GPRN parameter.

### 3.1 Mean-Field Inference for GPRN

In mean-field inference we use a family of factorized distributions:

$$q(\mathbf{f},\mathbf{w}) = \prod_{j=1}^{Q} q(\mathbf{f}_j) \prod_{i=1}^{P} q(\mathbf{w}_{ij}), \text{ with:} \quad (11)$$

$$q(\mathbf{f}_j) = \mathcal{N}(\mathbf{f}_j; \boldsymbol{\mu}_{\mathrm{f_j}}, \boldsymbol{\Sigma}_{\mathrm{f_j}}), \text{ and} \quad (12)$$

$$q(\mathbf{w}_{ij}) = \mathcal{N}(\mathbf{w}_{ij}; \boldsymbol{\mu}_{\mathrm{w_{ij}}}, \boldsymbol{\Sigma}_{\mathrm{w_{ij}}}). \quad (13)$$

where $\mathbf{f}_j = [\mathbf{f}_j(\mathbf{x}_1), \ldots, \mathbf{f}_j(\mathbf{x}_N)]^T$ and $\mathbf{w}_{ij} = [\mathbf{w}_{ij}(\mathbf{x}_1), \ldots, \mathbf{w}_{ij}(\mathbf{x}_N)]^T$. Here the approximating distributions in Equations (12) and (13) being full Gaussians is handy for exploiting the fact that the priors $p(\mathbf{f}_j)$ and $p(\mathbf{w}_{ij})$ are generated by GPs. However, as we shall see below, we only need $\mathcal{O}(N)$ parameters to characterize these full Gaussian distributions.

#### 3.1.1 Closed-Form Evidence Lower Bound

For the full Gaussian mean-field approximation, we can compute the evidence lower bound in Equation (10) in closed form. The expectation of the conditional likelihood wrt $q$ (first term in Equation (10)) is

$$\begin{aligned}
\mathbb{E}_q[\log p(\mathcal{D}|\mathbf{f},\mathbf{w})] &= -\frac{NP}{2}\log(2\pi\sigma_y^2) \\
&- \frac{1}{2\sigma_y^2}\sum_{n=1}^{N}(\mathbf{Y}_{\cdot n}^T - \boldsymbol{\Omega}_{\mathrm{w_n}}\boldsymbol{\nu}_{\mathrm{f_n}})^T(\mathbf{Y}_{\cdot n}^T - \boldsymbol{\Omega}_{\mathrm{w_n}}\boldsymbol{\nu}_{\mathrm{f_n}}) \\
&- \frac{1}{2\sigma_y^2}\sum_{i=1}^{P}\sum_{j=1}^{Q}\big[\mathrm{diag}(\boldsymbol{\Sigma}_{\mathrm{f_j}})^T(\boldsymbol{\mu}_{\mathrm{w_{ij}}} \bullet \boldsymbol{\mu}_{\mathrm{w_{ij}}}) \\
&\qquad + \mathrm{diag}(\boldsymbol{\Sigma}_{\mathrm{w_{ij}}})^T(\boldsymbol{\mu}_{\mathrm{f_j}} \bullet \boldsymbol{\mu}_{\mathrm{f_j}})\big], \quad (14)
\end{aligned}$$

where the subscript $n$ corresponds to the $n$th observation; $\mathbf{Y}_{\cdot n}^T$ is the $P$-dimensional vector of training targets corresponding to observation $n$; $\boldsymbol{\Omega}_{\mathrm{w_n}}$ is the $(P \times Q)$-dimensional matrix containing the means for the weight parameters; $\boldsymbol{\nu}_{\mathrm{f_n}}$ is the $Q$-dimensional vector of means for the latent function parameters; $\mathrm{diag}(\cdot)$ turns the diagonal elements of a matrix into a vector (or vice versa); and $\bullet$ denotes the Hadamard product.

The expectation of the log prior wrt $q(\mathbf{f},\mathbf{w})$ (second term in Equation (10)) is given by:

$$\mathbb{E}_q[\log p(\mathbf{f},\mathbf{w})] = \\ -\frac{1}{2}\sum_{j=1}^{Q}\left(\log|\mathbf{K}_f| + \boldsymbol{\mu}_{\mathrm{f_j}}^T\mathbf{K}_f^{-1}\boldsymbol{\mu}_{\mathrm{f_j}} + \text{ tr }(\mathbf{K}_f^{-1}\boldsymbol{\Sigma}_{\mathrm{f_j}})\right) \\ -\frac{1}{2}\sum_{i,j}\left(\log|\mathbf{K}_w| + \boldsymbol{\mu}_{\mathrm{w_{ij}}}\mathbf{K}_w^{-1}\boldsymbol{\mu}_{\mathrm{w_{ij}}} + \text{ tr }(\mathbf{K}_w^{-1}\boldsymbol{\Sigma}_{\mathrm{w_{ij}}})\right), \tag{15}$$

where $\mathbf{K}_f$ and $\mathbf{K}_w$ are the covariance matrices obtained by evaluating the covariance functions $\kappa_f$ and $\kappa_w$ on the training data respectively.

Finally, the entropy of the approximating distribution (third term in Equation (10)) is:

$$\mathcal{H}[q(\mathbf{f},\mathbf{w})] = \frac{1}{2}\sum_{j=1}^{Q}\log|\boldsymbol{\Sigma}_{\mathrm{f_j}}| + \frac{1}{2}\sum_{i,j}\log|\boldsymbol{\Sigma}_{\mathrm{w_{ij}}}| + \text{const.} \tag{16}$$

#### 3.1.2 Efficient Closed-form Updates for Variational Parameters

Using standard mean-field theory we obtain the following closed-form updates for the parameters of the variational distribution $q(\mathbf{f}_j)$:

$$\boldsymbol{\mu}_{\mathrm{f_j}} = \frac{1}{\sigma_y^2}\boldsymbol{\Sigma}_{\mathrm{f_j}}\sum_{i=1}^{P}\left(\mathbf{Y}_{\cdot i} - \sum_{k\neq j}\boldsymbol{\mu}_{\mathrm{w_{ik}}}\bullet\boldsymbol{\mu}_{\mathrm{f_k}}\right)\bullet\boldsymbol{\mu}_{\mathrm{w_{ij}}} \tag{17}$$

$$\boldsymbol{\Sigma}_{\mathrm{f_j}} = \left(\mathbf{K}_f^{-1} + \frac{1}{\sigma_y^2}\sum_{i=1}^{P}\mathrm{diag}(\boldsymbol{\mu}_{\mathrm{w_{ij}}}\bullet\boldsymbol{\mu}_{\mathrm{w_{ij}}} + \mathrm{Var}(\mathbf{w}_{ij}))\right)^{-1}, \tag{18}$$

where $\mathbf{Y}_{\cdot i}$ is the $N$-dimensional vector of observations corresponding to output $i$ and $\mathrm{Var}(\mathbf{w}_{ij}) = \mathrm{diag}(\boldsymbol{\Sigma}_{\mathrm{w_{ij}}})$.

Similarly for the parameters of $q(\mathbf{w}_{ij})$ we have:

$$\boldsymbol{\mu}_{\mathrm{w_{ij}}} = \frac{1}{\sigma_y^2}\boldsymbol{\Sigma}_{\mathrm{w_{ij}}}\left(\mathbf{Y}_{\cdot i} - \sum_{k\neq j}\boldsymbol{\mu}_{\mathrm{f_k}}\bullet\boldsymbol{\mu}_{\mathrm{w_{ik}}}\right)\bullet\boldsymbol{\mu}_{\mathrm{f_j}} \tag{19}$$

$$\boldsymbol{\Sigma}_{\mathrm{w_{ij}}} = \left(\mathbf{K}_w^{-1} + \frac{1}{\sigma_y^2}\mathrm{diag}(\boldsymbol{\mu}_{\mathrm{f_j}}\bullet\boldsymbol{\mu}_{\mathrm{f_j}} + \mathrm{Var}(\mathbf{f}_j))\right)^{-1}, \tag{20}$$

where $\mathrm{Var}(\mathbf{f}_j) = \mathrm{diag}(\boldsymbol{\Sigma}_{\mathrm{f_j}})$. We turn our attention to the analysis of Equations (18) and (20). We see that the estimated covariances are in terms of $\mathbf{K}_f$ and $\mathbf{K}_w$ plus a diagonal term. Hence, we approximate the posterior of the GPRN as a product of *full* covariances with a parameterization that requires only $\mathcal{O}(N)$ parameters. This result is similar to that obtained by Opper and Archambeau (2009), albeit more general, since we consider a larger class of likelihood models as given by the GPRN framework. We refer to the above updates as statistically *efficient*.

#### 3.1.3 Hyper-parameters Learning

We learn the hyper-parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_f, \boldsymbol{\theta}_w, \sigma_f, \sigma_y\}$ of the GPRN model by gradient-based optimization of the evidence lower bound in Equation (10), which can be computed for the mean-field approximation using Equations (14), (15), and (16). Note that using the point-estimates for hyper-parameters means we are assuming the posterior distribution of the hyper-parameters to be sharply peaked at one point. This assumption works well for GP models in practice (see e.g. Rasmussen and Williams, 2006, for a more thorough discussion). Detailed derivations of the gradients are given in the supplementary material.

### 3.2 Nonparametric Variational Inference for GPRN

Here we build upon the nonparametric variational inference framework of Gershman et al. (2012)[1]. We approximate the posterior distribution of the GPRN model using a mixture of $K$ isotropic Gaussian distributions:

$$q(\mathbf{u}) = \frac{1}{K}\sum_{k=1}^{K}q^{(k)}(\mathbf{u}) = \frac{1}{K}\sum_{k=1}^{K}\mathcal{N}(\mathbf{u};\boldsymbol{\mu}^{(k)},\sigma_k^2\mathbf{I}). \tag{21}$$

where $\boldsymbol{\mu}^{(k)}$ denotes the mean parameters and $\sigma_k^2$ denotes the variance of the mixture component $k$. The advantage of this approach is that we can approximate relatively complex posterior distributions efficiently using only $\mathcal{O}(KN)$ parameters for each factor. In practice $K$ is very small (typically less than 5) so the complexity is essentially $\mathcal{O}(N)$.

#### 3.2.1 Closed-form Evidence Lower Bound

In general, the expectations in equation (10) when using Equation (21) cannot be computed analytically and one needs to resort to approximations. Here we show that for the GPRN likelihood and prior model we can obtain exact analytical expressions for the first two terms in Equation (10). The main trick here is to realize that under the likelihood model of the GPRN and the isotropic nature of the covariances of the approximating distributions, the expectations decompose and we can apply our results from mean-field theory in the previous section.

In particular, we have that the expectation of the conditional likelihood wrt $q$ (first term in Equation (10))

[1] We follow the name used in the original paper but note that this is a parametric inference method.

is given by:

$$
\begin{aligned}
&\mathbb{E}_q[\log p(\mathcal{D}|\mathbf{f},\mathbf{w})] = \\
&- \frac{1}{2K\sigma_y^2}\sum_k\sum_n(\mathbf{Y}_{\cdot n}^T - \boldsymbol{\Omega}_{\mathrm{w_n}}^{(k)}\boldsymbol{\nu}_{\mathrm{f_n}}^{(k)})^T(\mathbf{Y}_{\cdot n}^T - \boldsymbol{\Omega}_{\mathrm{w_n}}^{(k)}\boldsymbol{\nu}_{\mathrm{f_n}}^{(k)}) \\
&- \frac{1}{2K}\Big(\sum_{k,j}\frac{P\sigma_k^2}{\sigma_y^2}{\boldsymbol{\mu}_{\mathrm{f_j}}^{(k)}}^T\boldsymbol{\mu}_{\mathrm{f_j}}^{(k)} + \sum_{k,i,j}\frac{P\sigma_k^2}{\sigma_y^2}{\boldsymbol{\mu}_{\mathrm{w_{ij}}}^{(k)}}^T\boldsymbol{\mu}_{\mathrm{w_{ij}}}^{(k)}\Big) \\
&- \frac{1}{2K}\Big(\sum_k\frac{\sigma_k^4}{\sigma_y^2}NPQ + NP\log(2\pi\sigma_y^2)\Big),
\end{aligned}
\tag{22}
$$

where $\boldsymbol{\mu}_{\mathrm{f_j}}^{(k)}$, $\boldsymbol{\mu}_{\mathrm{w_{ij}}}^{(k)}$ denote the mean parameters for the latent functions and weights, respectively, for component $k$. Here $\boldsymbol{\Omega}_{\mathrm{w_n}}^{(k)}$ ($P \times Q$ matrix) and $\boldsymbol{\nu}_{\mathrm{f_n}}^{(k)}$ ($Q$-dimensional vector) merely select the weight and latent mean parameters of the $n$-th sample – they are not additional parameters. The sums are over $k = 1, \ldots K$, $n = 1, \ldots N$, $i = 1, \ldots, P$ and $j = 1, \ldots, Q$.

Similarly, the expectation of the log prior wrt to the mixture distribution $q(\mathbf{u})$ in equation (21) (second term in Equation (10)) is given by:

$$
\begin{aligned}
\mathbb{E}_q[\log p(\mathbf{f},\mathbf{w})] &= -\frac{1}{2}\Big(Q\log|\mathbf{K}_f| + PQ\log|\mathbf{K}_w|\Big) \\
&- \frac{1}{2K}\Big[\sum_{k,j}{\boldsymbol{\mu}_{\mathrm{f_j}}^{(k)}}^T\mathbf{K}_f^{-1}\boldsymbol{\mu}_{\mathrm{f_j}}^{(k)} + \sigma_k^2 \text{ tr } (\mathbf{K}_f^{-1}) \\
&\qquad + \sum_{k,i,j}{\boldsymbol{\mu}_{\mathrm{w_{ij}}}^{(k)}}^T\mathbf{K}_w^{-1}\boldsymbol{\mu}_{\mathrm{w_{ij}}}^{(k)} + \sigma_k^2 \text{ tr } (\mathbf{K}_w^{-1})\Big].
\end{aligned}
\tag{23}
$$

The only remaining term in the evidence lower bound in Equation (10) is the entropy $\mathcal{H}_q[q(\mathbf{u})]$. Here we use the result in Huber et al. (2008) to provide a lower bound for this term:

$$
\mathcal{H}_q[q(\mathbf{u})] \geq -\frac{1}{K}\sum_{k=1}^{K}\log\frac{1}{K}\sum_{j=1}^{K}\mathcal{N}(\boldsymbol{\mu}^{(k)};\boldsymbol{\mu}^{(j)},(\sigma_k^2+\sigma_j^2)\mathbf{I}).
\tag{24}
$$

Simulation results from Huber et al. (2008) showed that this lower bound is closer to the true entropy value compared to the previously well-known single Gaussian bound. Hence, Equations (22), (23) and (24) define a tight analytical lower bound of the evidence lower bound in the nonparametric variational inference method for GPRNs.

### 3.2.2 Optimization of Variational Parameters and Hyper-parameters

We carry out optimization of the variational parameters $\{\boldsymbol{\mu}_{\mathrm{f_j}}^{(k)}, \boldsymbol{\mu}_{\mathrm{w_{ij}}}^{(k)}\}$ and hyper-parameters $\boldsymbol{\theta}$ by maximization of the evidence lower bound in Equation (10) using Equations (22), (23) and (24) and gradient-based optimization. Unlike the original method of Gershman et al. (2012), our algorithm naturally guarantees optimization of the evidence lower bound in the model. Detailed derivations of the gradients of the lower bound w.r.t all parameters are given in the supplementary material.

### 3.3 Predictive Distributions

For a new input location $\mathbf{x}^*$ we can use the approximate posterior to predict its *noiseless* outputs $\mathbf{y}^*$. For both approximations, the predictive distributions are analytically intractable due to the non-Gaussian likelihood wrt to the parameters $\mathbf{f}$ and $\mathbf{w}$ of the GPRN models. However their predicted means can be computed, which for mean-field approximation is

$$
\mathbb{E}[\mathbf{y}^*|\mathbf{x}^*,\mathcal{D}] = \mathbf{K}_w^*\mathbf{K}_w^{-1}\boldsymbol{\mu}_{\mathbf{w}}\mathbf{K}_f^*\mathbf{K}_f^{-1}\boldsymbol{\mu}_{\mathbf{f}}
\tag{25}
$$

where $\mathbf{W}^* = \mathbf{W}(\mathbf{x}^*)$ and $\mathbf{f}^* = \mathbf{f}(\mathbf{x}^*)$. Here $\mathbf{K}_w^*$ and $\mathbf{K}_f^*$ are the covariance matrices corresponding to the covariance functions $\kappa_w$ and $\kappa_f$ evaluated on the test point $\mathbf{x}^*$ wrt all of the training data; $\boldsymbol{\mu}_{\mathbf{w}}$ and $\boldsymbol{\mu}_{\mathbf{f}}$ are the mean of the latent and weight functions, respectively. Intuitively, the predictive mean is a linear combination of the predictive latent means with predictive weight means as the mixing coefficients.

For nonparametric variational inference, the predictive mean turns out to be the average of the predictions made by all components:

$$
\mathbb{E}[\mathbf{y}^*|\mathbf{x}^*,\mathcal{D}] = \frac{1}{K}\sum_{k=1}^{K}\mathbf{K}_w^*\mathbf{K}_w^{-1}\boldsymbol{\mu}_{\mathbf{w}}^{(k)}\mathbf{K}_f^*\mathbf{K}_f^{-1}\boldsymbol{\mu}_{\mathbf{f}}^{(k)}
\tag{26}
$$

where the notations are the same as in Equation (25). The subscript $k$ denotes the variational parameters of the $k$-th component in the approximate mixture posterior. Detailed derivations are in the supplementary material.

## 4 Experiments

We compare the performance of nonparametric variational inference (GPRN-NPV) and mean-field (GPRN-MF) using two real datasets. We use an independent GP model (IGP) as a reference method. However, we emphasize that our goal here is not to evaluate the GPRN as a multi-output regression model. Instead we aim at assessing the quality of our inference methods. The same preprocessing of data is done for all methods.

We use the squared exponential covariance functions with automatic relevance determination (ARD) for the two priors $p(\mathbf{w})$ and $p(\mathbf{f})$. We learn their hyper-parameters by optimizing the evidence lower bound while holding the variational parameters fixed. This procedure is similar to variational EM and is guaranteed to converge. L-BFGS is used as the optimization method. We found that it is effective to fix the signal variance of the latent processes and let the weight processes adapt to the scale of the response variables.

For all methods we repeat the experiments ten times with different random initializations of the hyperparameters and variational parameters (where applicable). All experiments are executed on a Intel(R) Core(TM) i7-2600 3.40GHz CPU with Matlab R2012a.

## 4.1 Description of the Datasets

We use two real world datasets. In the first dataset we are interested in estimating the concentrations of a metal using observations from other metals, which is a very common setting in geostatistics. In the second dataset we are interested in predicting for three quality measurements of concrete simultaneously. It is expected a priori that the outputs in both datasets have complex dependencies.

### 4.1.1 Jura Geostatistics

This dataset consists of measurements of concentrations of heavy metals in a 14.5 $km^2$ region of the Swiss Jura. Following previous work (see e.g. Goovaerts, 1997; Alvarez and Lawrence, 2009; Wilson et al., 2012), the task here is to predict the concentrations of cadmium at 100 locations represented by two-dimensional spatial coordinates. We use for training the concentrations of cadmium, nickel, and zinc at 259 nearby locations in conjunction with the measurements of nickel and zinc at the 100 locations where we want to make prediction for cadmium. This setting is important as we can collect samples from related, more accessible metals and enhance prediction for less accessible ones based on the learned correlations of metal concentrations. We standardize each input dimension to have zero mean and unit variance and log-transform the outputs before normalizing them.

### 4.1.2 Concrete Quality

Concrete has been used extensively in construction yet modelling its behavior is still a very difficult task due to its complex composite structure. A concrete mixture composes primarily of ingredients such as cement, water content, chemical and mineral admixtures. Different combinations of the constituents produce varying properties of concrete. For example, one important property of high-performance concrete is slump flow, which partially indicates the consistency in concrete workability. It increases with the level of water but decreases slightly after the water content passes a certain threshold. It also increases with the amount of the mineral admixture fly ash but decreases rapidly after the fly ash content reaches a certain level. All other concrete materials can similarly influence the final outcome of a concrete mixture, which is not only determined by the slump flow but also other quality indicators such as compressive strength.

The original dataset contains 103 samples with 7 input variables corresponding to the 7 concrete mixing ingredients. The 3 output variables (slump, flow, and compressive strength) are concrete quality measures. For a detailed description of the dataset the reader is referred to Yeh (2007). We randomly split it into a training set of 80 points and a test set of 23 points. We use the water, fly ash, and superplasticizer content as the input features as they have been shown to have interesting correlations with the quality of concrete (Yeh, 2007). All input and output dimensions are normalized to have zero mean and unit variance for training.

## 4.2 Results

In this section we first present experimental results on the two data sets. For exploratory purposes, we report GPRN-NPV where the approximate posterior is a mixture of 1, 2 and 3 components, which we denote as NPV1, NPV2, NPV3, respectively. From our experience using 3 modes is enough to capture major aspects of the true posterior. However harder problems may require higher multimodal approximations. An analysis of the computation and convergence aspect of the methods is also discussed at the end of this section.

### 4.2.1 Results for Jura Geostatistics

We assess the performance of all methods using Mean-absolute-error (MAE) as done in previous work (Alvarez and Lawrence, 2009; Wilson et al., 2012). The average MAE and two standard errors across 10 runs are shown in Figure 1. The GPRN-NPV method with multimodality (NPV2 and NPV3) has better predictive performance compared to the unimodal counterpart (NPV1 and GPRN-MF). They also exhibit less variability in different runs. This perhaps can be attributed to the posterior having multiple modes, which was indeed our main motivation for using GPRN-NPV from the beginning. GPRN-MF and NPV1 has only a single mode and hence may converge to a bad local minimum. NPV2 and NPV3, on ther other hand, can fit multiple modes in the posterior and is thus more robust against the extreme local minima. However we note that the performances of NPV1 and GPRN-MF are still superior compared to IGP's. This shows that mean-field variational inference for GPRN can still be a good method,

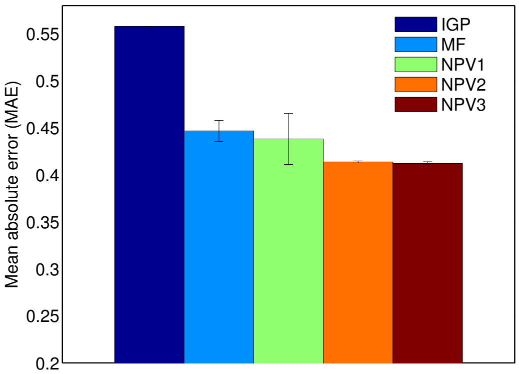


Figure 1: Mean absolute error for the Jura geostatistics dataset of IGP, GPRN-MF, and GPRN-NPV with 1,2 and 3 modes. The mean value is averaged across 10 runs and the error bars show 2 standard errors.

particularly in cases where the true posterior has a few modes that represent the data equally well.

### 4.2.2 Results for Concrete Quality

For this dataset we use the standardized mean squared error (SMSE, as in Rasmussen and Williams, 2006, Sec. 2.5). SMSE is a reasonable measure of predictive performance when there is large variation in the values of the test outputs, which is what we have in this case. The SMSE and two standard errors of the 3 outputs averaged across 10 runs are shown in Figure 2.

The non-parametric method with 1, 2 and 3 modes outperforms IGP for all 3 outputs. They also outperform GPRN-MF with respect to the prediction of slump and flow but mean-field performs better for the compressive strength output. This is an interesting result as it suggests the multimodality of the posterior distribution and the power of GPRN-NPV to match this multimodality. Here GPRN-MF seems to always converge to a local minium that explains the concrete compressive strength well. However such over-representation diminishes the information from the two remaining outputs and is likely to lead to an underfit for these outputs. GPRN-NPV on the other hand does not place all of its mass on any particular single mode in the posterior distribution. Therefore it tries to fit the data from all outputs, leading to better prediction in general. In fact, average SMSEs across all outputs for GPRN-MF, NPV1, NPV2 and NPV3 are $0.8717, 0.8256, 0.8206$ and $0.8240$ respectively.

### 4.2.3 Computational Cost and Convergence of Parameter Optimization

We now present the computation and convergence behavior of the GPRN-MF and GPRN-NPV methods. In

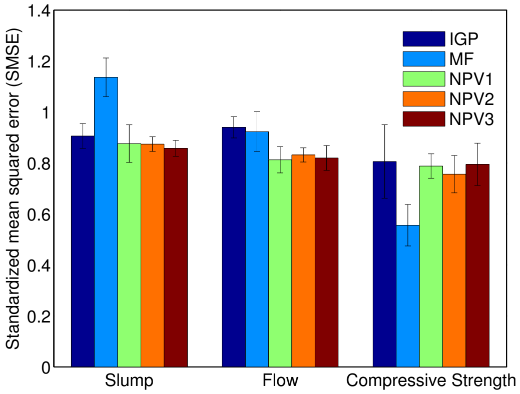
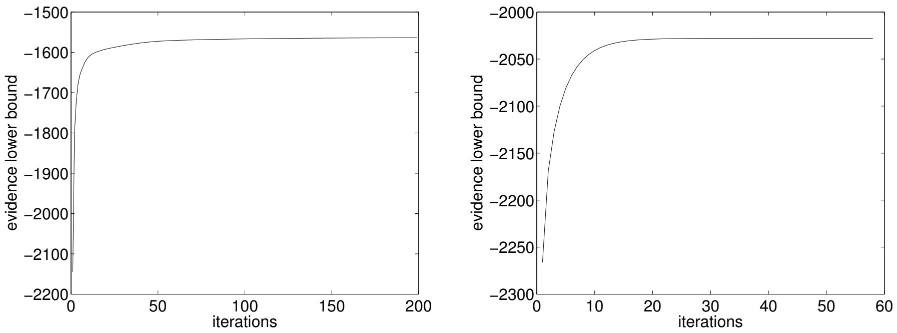


Figure 2: Standardized mean squared error for the Concrete dataset of IGP, GPRN-MF, and GPRN-NPV with 1,2 and 3 modes. The mean is averaged across 10 runs and the error bars show 2 standard errors.

Table 1 we show the average training time per optimization iteration on both data sets (one iteration updates all variational parameters and hyperparameters in the model). Recall that the variational parameters in GPRN-NPV scales linearly with the number of mixture components. This is reflected in the average training times per iteration where we see the training time indeed scales linearly with the number of modes. In theory GPRN-MF should be faster than GPRN-NPV with one mode as updates for the variational parameters in mean-field are done with closed-formed in contrast to GPRN-NPV where parameter updates are done via gradient-based search. However GPRN-NPV exhibits better convergence property (i.e., it converges at a much faster rate), and hence the average training time per iteration can be lower as seen in Table 1 for the Jura dataset. A more illustrating view on the convergence of both GPRN-MF and GPRN-NPV can be found in Figure 3 where we show two plots of the evidence lower bound in a sample run of GPRN-MF and NPV2. Both methods get close to a good value of the evidence lower bound very quickly but GPRN-MF moves slowly towards the maximum (see the long tail until the 200th iteration) whereas NPV2 achieves convergence after only 60 iterations. Our final observation concerns the optimization of the hyper-parameters for both methods. When the number of variational parameters is small, the main work of one optimization

Table 1: Average training time per iteration (seconds) for each of the variational inference algorithms.

| | GPRN-MF | NPV1 | NPV2 | NPV3 |
|---|---|---|---|---|
| **Jura** | 3.32 | 2.86 | 5.30 | 9.07 |
| **Concrete** | 0.29 | 1.27 | 2.54 | 3.80 |



(a) GPRN-MF

(b) NPV2

Figure 3: Convergence of the evidence lower bound for GPRN-MF and NPV2 from 2 example runs. The evidence lower bound is shown as a function of the number of iterations.

iteration is dominated by the cost of updating the hyperparameters. This requires taking inverses of the $N \times N$ covariance matrix which has the computation complexity of $\mathcal{O}(N^3)$, where $N$ is the maximum number of observations of all outputs. In our experience, especially for GPRN-NPV, the hyper-parameters converge to an optimum relatively quickly after a dozen iterations. This means the optimization procedure does not perform many searches in the later iterations and the cost of inverting a matrix decreases as the number of iterations increases.

## 5 Related Work

Various multi-task models have been proposed in the machine learning literature. Here we briefly describe how these models relate to the GPRN. The closest model to GPRNs is the semi-parametric latent factor model (SLFM, Teh et al., 2005), where the correlations between the $P$ outputs are also induced through the linear combination of $Q$ latent Gaussian processes. However, unlike the GPRN, these correlations are not spatially adaptive, as the weight matrix does not depend on the input $\mathbf{x}$. The SLFM is a generalization of the multi-task model of Bonilla et al. (2008), with $P = Q$ and all the latent processes share the same covariance function. The convolved GP (Alvarez and Lawrence, 2009) is somewhat a generalization of the SLFM, and consequently of the multi-task GP, where each output is a combination of latent GPs across the whole input domain. This yields a useful "smoothing" effect but it implies that the output dependencies are global and do not vary as a function of $\mathbf{x}$.

As highlighted by Wilson et al. (2012), the GPRN has the following advantages over previous multi-task models: (a) the dependencies between the outputs are spatially adaptive; (b) the noise correlations also depend on the input $\mathbf{x}$; and (c) inference only scales linearly with the number of outputs. These are the main reasons why this paper focuses on efficient inference for this model.

Wilson et al. (2012) propose an MCMC-based method and a variational-message passing technique for inference. Their experiments show that the variational method can be as accurate as the MCMC method but is more efficient. In the limit, the solution of their variational method should tend to our GPRN-MF approach, and hence we have considered the GPRN-MF as a reasonable baseline. However, as highlighted throughout this paper, we derive closed-form updates for the variational parameters of this model and and we obtain an efficient parameterization of the full Gaussians used as the approximating factorizing distributions.

**Discussion** We have shown that our GPRN-NPV method is superior to the mean-field approximation in both accuracy and stability. Furthermore, we have derived a proper evidence lower bound for this model, which we use for the optimization of the variational parameters and hyper-parameters of latent Gaussian processes. In future work we aim to extend our inference methods to other types of likelihoods, such as those used in classification or preference learning. We will also incorporate sparse approaches into our variational methods so that they scale to large datasets.

## 6 Acknowledgements

EVB gratefully acknowledges funding for the project "Data Fusion and Machine Learning for Geothermal Target Exploration and Characterization" by the Australian Renewable Energy Agency (ARENA) and National ICT Australia (NICTA)". NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

## References

Alvarez, M. and Lawrence, N. D. (2009). Sparse convolved gaussian processes for multi-output regression. In *NIPS*, pages 57–64.

Bonilla, E. V., Chai, K. M. A., and Williams, C. K. I. (2008). Multi-task Gaussian process prediction. In *NIPS*.

Boyle, P. and Frean, M. (2005). Dependent gaussian processes. In *NIPS*.

Gershman, S. J., Hoffman, M. D., and Blei, D. M. (2012). Nonparametric variational inference. In *ICML*.

Goovaerts, P. (1997). *Geostatistics for Natural Resources Evaluation*. Oxford Univ. Press.

Huber, M. F., Bailey, T., Durrant-Whyte, H., and Hanebeck, U. D. (2008). On entropy approximation for Gaussian mixture random vectors. In *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*.

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

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

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

Teh, Y. W., Seeger, M., and Jordan, M. I. (2005). Semiparametric latent factor models. In *AISTATS*.

Wilson, A. G., Knowles, D. A., and Ghahramani, Z. (2012). Gaussian process regression networks. In *ICML*.

Yeh, I.-C. (2007). Modeling slump flow of concrete using second-order regressions and artificial neural networks. *Cement and Concrete Composites*, 29(3):474–480.