Multi-resolution Multi-task Gaussian Processes

Oliver Hamelijnck
The Alan Turing Institute
Department of Computer Science
University of Warwick

Theodoros Damoulas
The Alan Turing Institute
Depts. of Computer Science & Statistics
University of Warwick

Kangrui Wang
The Alan Turing Institute

Mark Girolami
The Alan Turing Institute
University of Cambridge

Abstract

We consider evidence integration from potentially dependent observation processes under varying spatio-temporal sampling resolutions and noise levels. We develop a multi-resolution multi-task (MRGP) framework while allowing for both *inter-task* and *intra-task* multi-resolution and multi-fidelity. We develop shallow Gaussian Process (GP) mixtures that approximate the difficult to estimate joint likelihood with a composite one and deep GP constructions that naturally handle biases in the mean. By doing so, we generalize and outperform state of the art GP compositions and offer information-theoretic corrections and efficient variational approximations. We demonstrate the competitiveness of MRGPs on synthetic settings and on the challenging problem of hyper-local estimation of air pollution levels across London from multiple sensing modalities operating at disparate spatio-temporal resolutions.

1 Introduction

The increased availability of ground and remote sensor networks coupled with new sensing modalities, arising from e.g. citizen science intiatives and mobile platforms, is creating new challenges for performing formal evidence integration. These multiple observation processes and sensing modalities can be dependent, with different signal-to-noise ratios and varying sampling resolutions across space and time. In our motivating application, London authorities measure air pollution from multiple sensor networks; high-fidelity ground sensors that provide frequent multi-pollutant readings, low fidelity diffusion tubes that only provide monthly single-pollutant readings, hourly satellite-derived information at large spatial scales, and high frequency medium-fidelity multi-pollutant sensor networks. Such a multi-sensor multi-resolution multi-task evidence integration setting is becoming prevalent across any real world application of machine learning.

The current state of the art, see also Section 5, is assuming product likelihoods and unbiased observation processes as in [13], and cannot handle the challenges of real world settings that are jointly *non-stationary*, *multi-task*, *multi-fidelity*, and *multi-resolution* [1, 6, 13, 20, 21, 26, 27]. The latter challenge has recently attracted the interest of the machine learning community under the context of working with aggregate, binned observations [1, 13, 27] or the special case of natural language generation at multiple levels of abstraction [26]. The independence and unbiasedness assumptions lead to posterior contraction, degradation of performance and of uncertainty quantification.

In this paper we introduce a multi-resolution multi-task GP framework that can integrate evidence from observation processes with varying support (e.g. partially overlapping in time and space), that can be dependent and biased while allowing for both *inter-task* and *intra-task* multi-resolution and multi-fidelity. Our first contribution is a shallow GP mixture, MR-GPRN, that corrects for the

Preprint. Work in progress.

dependency between observation processes through composite likelihoods and extends the Gaussian aggregation model of Law et al. [13], the multi-task GP model of Wilson et al. [33], and the variational lower bound of Nguyen and Bonilla [18]. Our second contribution is a multi-resolution deep GP composition that can additionally handle biases in the observation processes and extends the deep GP models and variational lower bounds of Damianou and Lawrence [5] and Salimbeni and Deisenroth [25] to varying support, multi-resolution data. Lastly, we demonstrate the superiority of our models on synthetic problems and on the challenging spatio-temporal setting of predicting air pollution in London at hyper-local resolution.

Sections 3 and 4 introduce our shallow GP mixtures and deep GP constructions respectively. In Section 6 we demonstrate the empirical advantages of our framework verus the prior art followed by a additional related work in Section 5 and our concluding remarks. Further analysis and full derivations are provided in the Supplement.

2 Multi-resolution Multi-task Learning

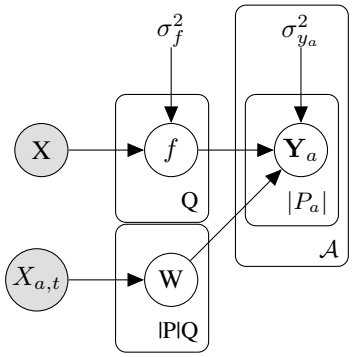
Consider $\mathcal{A} \in \mathbb{N}$ observation processes $\mathbf{Y}_a$ with varying resolutions $\mathcal{R}_a$ and *sampling periods* $\mathcal{S}_a$. The observation process $\mathbf{Y}_{a'}$ with the highest sampling rate per unit measure and hence the smallest sampling period has the *base sampling period* $\mathcal{S}_{a'} = \mathcal{B}$. Typically the observation process of interest, denoted $\mathbf{Y}_\mathcal{B}$, is at the sampling period $\mathcal{B}$ but our formulation and framework applies generally. We normalize the sampling periods with respect to the base one such that $\mathcal{S}_{a'} \coloneqq 1$ and hence $\mathcal{S}_{a \neq a'} \in \mathbb{R}_{\geq 1}$. For example if the base period is hourly and we have a process a with N_a daily observations then its sampling period is $\mathcal{S}_a$=24 and $\mathbf{Y}_a \in \mathbb{R}^{N_a \times 1}$ and $\mathbf{X}_1 \in \mathbb{R}^{N_a \times 24 \times 1}$. We construct $\mathcal{A}$ datasets $\{(\mathbf{X}_a, \mathbf{Y}_a)\}_{a=1}^{\mathcal{A}}$ across the set of all tasks $P = \{p\}$ with $\mathbf{X}_a \in \mathbb{R}^{N_a \times \mathcal{S}_a \times D_a}$ and $\mathbf{Y}_a \in \mathbb{R}^{N_a \times |P_a|}$ where $N_a \in \mathbb{N}$, $P_a \subseteq P$, $D_a \in \mathbb{N}$ are the number of observations, set of observed tasks and input dimensions, for the observation process a.

When our observation processes are also multivariate we have additional dependencies arising from this multi-task setting [3]. Multi-resolution observations can now exist both within tasks (*intra-task multi-resolution*) and across tasks (*inter-task multi-resolution*). In our motivating application, multiple sensing modalities measure multiple air pollutant levels (e.g. CO_2, NO_2, PM10, PM25) that can be highly correlated across space-time due to e.g. common emission sources and dispersion mechanisms. We are interested in flexible non-stationary non-parametric models that can scale to millions of observations while delivering improved uncertainty quantification.

3 Multi-Resolution Gaussian Process Regression Networks (MR-GPRN)

We first introduce a *shallow* instantiation of the multi-resolution multi-task framework. MR-GPRN is a shallow GP mixture, Fig. 1, that extends the Gaussian process regression network (GPRN) [33], itself a special case of the Linear Coregionalization Model (LCM) [2]. Briefly, the GPRN jointly models the set of tasks P by introducing $Q \in \mathbb{N}$ latent GP functions $\mathbf{f}_q$ that are mixed through further $|P|Q \in \mathbb{N}$ latent GP functions $\mathbf{W}_{p,q}$. More formally, $\mathbf{f}_q \sim \mathcal{GP}(0, \mathbf{K}_q^f)$ and $\mathbf{W}_{p,q} \sim \mathcal{GP}(0, \mathbf{K}_{p,q}^w)$. For task$p$ the observations $\mathbf{Y}_p = \mathbf{W}_p\mathbf{f} + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma_y^2\mathbf{I})$ such that $\mathbf{Y}_p$ is a (noisy) linear combination of product of GPs. This enables learning non-stationary random fields by varying the mixing of stationary latent GPs across input space.

Model Specification. We place a GPRN prior over the joint $p(\mathbf{W}, \mathbf{f}) = \prod_{i,j}^{|P|Q} \mathcal{N}(\mathbf{W}_{ij}|0, \mathbf{K}_{ij}) \prod_i^Q \mathcal{N}(\mathbf{f}_i|0, \mathbf{K}_i)$. Apart from the standard inter-task dependency we ideally also want to model directly the additional dependency of the observation processes but it can vary in input space quickly leading to intractability. At the other end, one can ignore this dependency and assume a product likelihood form, as in [13, 17], but this misspecification results in severe posterior contraction (see Fig. 2) when the independence assumption is violated. To circumvent these extremes we propose to approximate the full likelihood using a multi-resolution composite likelihood. This posterior over the latent functions is now:



Algorithm 1 Inference of MR-GPRN

Input: $\mathcal{A}$ multi-resolution datasets $\{(\mathbf{X}_a, \mathbf{Y}_a)\}_{a=1}^{\mathcal{A}}$, initial parameters θ,

$\hat{\theta} \leftarrow \arg\max_\theta \sum_{a=1}^{\mathcal{A}} \ell(\mathbf{Y}_a|\theta)$

$\mathbf{H} \leftarrow \sum_{a=1}^{\mathcal{A}} (\nabla \ell(\mathbf{Y}_a|\hat{\theta})(\nabla \ell(\mathbf{Y}_a|\hat{\theta}))^T$

$\mathbf{J} \leftarrow \nabla^2 \ell(\mathbf{Y}|\hat{\theta})$

$$\phi \leftarrow \begin{cases} \frac{|\hat{\theta}|}{\mathrm{Tr}[\mathbf{H}(\hat{\theta})^{-1}\mathbf{J}(\hat{\theta})]} \\ \frac{\mathrm{Tr}[\mathbf{H}(\hat{\theta})\mathbf{J}(\hat{\theta})^{-1}\mathbf{H}(\hat{\theta})]}{\mathrm{Tr}[\mathbf{H}(\hat{\theta})]} \end{cases}$$

$\theta_1 \leftarrow \arg\min_\theta \left(\sum_{a=1}^{\mathcal{A}} \phi \mathbb{E}_q \left[\ell(\mathbf{Y}_a|\theta) \right] + \mathcal{KL} \right)$

Figure 1: **Left**: Graphical model of MR-GPRN for $\mathcal{A}$ observation processes each with $|P_a|$ tasks. This allows *multi-resolution learning* between and across tasks. **Right**: Inference for MR-GPRN.

$$p(\mathbf{W}, \mathbf{f}|\mathbf{Y}) \propto \underbrace{\prod_{a=1}^{\mathcal{A}} \prod_{p=1}^{|P_a|} \prod_{n=1}^{N} \mathcal{N}(\mathbf{Y}_{a,p,n}|\frac{1}{\mathcal{S}_a} \int_{s=1}^{\mathcal{S}_a} \mathbf{W}_p(\mathbf{X}_{a,n,s})\mathbf{f}(\mathbf{X}_{a,n,s})\, d\mathbf{X}_{a,n,s}, \sigma_a^2\mathbf{I})^\phi}_{\text{MR-GPRN Composite Likelihood}} \underbrace{p(\mathbf{W}, \mathbf{f})}_{\text{GPRN Prior}}. \tag{1}$$

where $\phi \in \mathbb{R}_{>0}$ are the composite weights that are critical for inference [30]. We tie the underlying process $\mathbf{Wf}$ to each of observations processes by integrating over the sampling period of $X_{r,n}$. In general the integral is not available in closed form and so we approximate it by discretizing over a uniform grid. When we only have one task and $\mathbf{W}$ is a matrix constant functions we denote the model as MR-GP.

Composite Likelihood Weights. Under a misspecified model the asymptotic distribution of the MLE estimate converges to $\mathcal{N}(\theta_0, \frac{1}{n}\mathbf{H}(\theta_0)\mathbf{J}(\theta_0)^{-1}\mathbf{H}(\theta_0))$ where θ_0 are the true parameters and $\mathbf{H}(\theta) = \frac{1}{n}\sum_{n=1}^{N} \nabla\ell(\mathbf{Y}|\theta)\nabla\ell(\mathbf{Y}|\theta)^T$ and $\mathbf{J}(\theta) = \frac{1}{n}\sum_{n=1}^{N} \nabla^2\ell(\mathbf{Y}|\theta)$ the Hessian and Jacobian respectively. The form of the asymptotic variance is the *sandwich information matrix* and it represents the loss of information in the MLE estimate due to the failure of Bartletts second identity [30].

Following Lyddon et al. [15] and Ribatet [24] we write down the asymptotic posterior of MR-GPRN as $\mathcal{N}(\theta_0, n^{-1}\phi^{-1}\mathbf{H}(\theta))$. Asymptotically one would expect the contribution of the prior to vanish causing the asymptotic posterior to match the limiting MLE. The composite weights ϕ can be used to bring these distributions as close together as possible. By setting $\phi^{-1}\mathbf{H}(\hat{\theta}) = \mathbf{H}(\theta_0)\mathbf{J}(\theta_0)^{-1}\mathbf{H}(\theta_0)$, and rearranging to find ϕ we recover the magnitude correction of Ribatet [24]. Instead if we take traces and then rearrange we recover the correction of Lyddon et al. [15]:

$$\phi_{\text{Ribatet}} = \frac{|\hat{\theta}|}{\mathrm{Tr}[\mathbf{H}(\hat{\theta})^{-1}\mathbf{J}(\hat{\theta})]} \quad , \quad \phi_{\text{Lyddon}} = \frac{\mathrm{Tr}[\mathbf{H}(\hat{\theta})\mathbf{J}(\hat{\theta})^{-1}\mathbf{H}(\hat{\theta})]}{\mathrm{Tr}[\mathbf{H}(\hat{\theta})]}. \tag{2}$$

Inference. We derive a closed form evidence lower bound (ELBO) for MR-GPRN, see Supplementary. For computational efficiency we introduce inducing points $\mathbf{U} = \{\mathbf{u_q}\}_{q=1}^{Q=1}$ for $\mathbf{u_q} \in \mathbb{R}^M$, $\mathbf{V} = \{\mathbf{v_{p,q}}\}_{p,q=1}^{|P|,Q}$ for $\mathbf{v_{p,q}} \in \mathbb{R}^M$ at the corresponding locations $\mathbf{Z^{(u)}} = \{\mathbf{Z_q^{(u)}}\}_{q=1}^{Q}, \mathbf{Z^{(v)}} = \{\mathbf{Z_{p,q}^{(v)}}\}_{p,q=1}^{|P|,Q}$ for $\mathbf{Z}^{(\cdot)} \in \mathbb{R}^{M,D}$ for each latent GP following [29]. We construct the augmented posterior and use the approximate posterior $q(\mathbf{u}, \mathbf{v}, \mathbf{f}, \mathbf{W})p(\mathbf{f}, \mathbf{W}|\mathbf{u}, \mathbf{v})q(\mathbf{u}, \mathbf{v})$ where $q(\mathbf{u}, \mathbf{v}) = \sum_{k=1}^{K} \pi_k \prod_{q=j}^{Q} \mathcal{N}(\mathbf{m}_j^{(\mathbf{u})}, \mathbf{S}_j^{(\mathbf{u})}) \cdot \prod_{i,j=1}^{|P|,Q} \mathcal{N}(\mathbf{m}_{i,j}^{(\mathbf{v})}, \mathbf{S}_{i,j}^{(\mathbf{v})})$ is a free form mixture of Gaussians. The

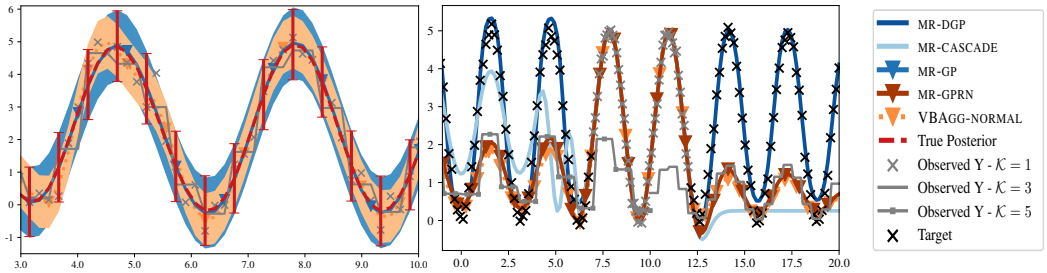


Figure 2: **Left**: MR-GP recovers the true predictive variance whereas the product likelihood assumption in VBAGG-NORMAL leads to posterior contraction. **Right**: MR-DGP recovers the true predictive mean under a multi-resolution setting with scaling biases. Both VBAGG-NORMAL and MR-GPRN fail as they propagate the bias. Grey crosses and lines denote observed values. Black crosses denote observations removed for testing.

expected log-likelihood (ELL) of each component a is available in closed form:

$$\mathcal{L}_a = \sum_{n}^{|P_i|}\sum^{N} \mathbf{Y}_{a,i,n}^T \mathbf{Y}_{a,i,n} + \sum_{a}^{\mathcal{S}_a}\sum_{b=a}^{\mathcal{S}_a} \mathrm{Tr}[\mathbf{\Sigma}_{W_{ni}^a}\mathbf{\Sigma}_{f_n^a}] + \mu_{f_a}^T \mathbf{\Sigma}_{W_{ni}^a}\mu_{f_n^a} + \mu_{W_{ni}^a}^T \mathbf{\Sigma}_{f_n^a}\mu_{W_{ni}^a} + \frac{2}{\mathcal{S}_a}\mathbf{Y}_{a,i,n}^T \sum_{a}^{\mathcal{S}_a} \mu_{W_{ni}^a}^T \mu_{f_n^a} + \frac{1}{{\mathcal{S}_a}^2}\sum_{a}^{\mathcal{S}_a}\sum_{b}^{\mathcal{S}_a} \mu_{f_n^a}^T \mu_{w_{ni}^a}^T \mu_{w_{ni}^b}\mu_{f_n^b}. \tag{3}$$

where $\mu_{W_{ni}^a}^a$, $\mathbf{\Sigma}_{W_{ni}^a}^a$ and $\mu_{f_n^a}^a$, $\mathbf{\Sigma}_{f_n^a}^a$ are respectively the mean and variance of $q(\mathbf{W})$ and $q(\mathbf{f})$ at input $\mathbf{X}_{a,n,s}$. To infer the composite weights we first obtain the MLE estimate of θ by maximizing the likelihood in Eq. 1. The weights can then be calculated and the variational lowerbound optimised as in Alg. 1 with $\mathcal{O}(E \cdot (|P|Q+Q)NM^2)$ for E optimization steps until convergence. Our closed form ELBO generalizes prior art on the GPRN when there is only one observation process a and $R_a = 1$ by allowing for a free form mixture of Gaussian variational posteriors [12, 18].

Predictive Density. Although the full predictive distribution of a specific observation process is not available in closed form, using the variational posterior we derive the predictive mean and variance, avoiding MC estimates. The mean is $\mathbb{E}[\mathbf{Y}_{a,p}^*] = \sum_k^K \pi_k E_k\left[\mathbf{W}_p^*\right]\mathbb{E}_k[\hat{\mathbf{f}}^*]$ and the variance is: $\mathbb{V}[\mathbf{Y}_{\mathbf{a},\mathbf{p}}^*] = \sum_k^K \pi_k(\sigma_y^2 I + \mathrm{Tr}(\mathbb{V}_k[\hat{\mathbf{f}}^*]\mathbb{V}_k[\mathbf{W}_p^*]) + \mathbb{E}_k\left[\mathbf{W}_p^*\right]\mathbb{V}_k[\hat{\mathbf{f}}^*]\mathbb{E}_k\left[\mathbf{W}_p^*\right] + \mathrm{Tr}(\mathbb{E}_k[\hat{\mathbf{f}}^*]\mathbb{E}_k[\hat{\mathbf{f}}^*]^{\mathrm{T}}\mathbb{V}_k[\mathbf{W}_p^*]) + \mathbb{E}_k\left[\mathbf{W}_p^*\right]\mathbb{E}_k\left[\hat{\mathbf{f}}^*\right]\mathbb{E}_k\left[\hat{\mathbf{f}}^*\right]^{\mathrm{T}}\mathbb{E}_k\left[\mathbf{W}_p^*\right]^{\mathrm{T}}) - \mathbb{E}[\mathbf{Y}_a^*]\mathbb{E}[\mathbf{Y}_a^*]^{\mathrm{T}}$. Where K is the number of components in the mixture of Gaussians variational posterior and π_k is the k'th weight. Full derivations are in the Supp. together with corresponding ones for the positively-restricted GPRN form $\mathbf{Y} = \sum_{p=1}^{|P|}\exp(\mathbf{W}_p)\mathbf{f} + \epsilon$ that improves identifiability and predictive performance.

4 Multi-Resolution Deep Gaussian Processes (MR-DGP)

We now introduce a *deep* instantiation of the multi-resolution multi-task framework. MR-DGP is a deep GP (DGP) composition that extends the model of Damianou and Lawrence [5] into a tree-structured multi-resolution composition, Fig. 3. Briefly, a DGP is a composition of GPs where the output of each layer feeds into the input of the next. This hierarchical structure allows for complex, non-stationary, processes to be modelled without the use of highly parameterized kernels.

Model Specification. Consider a multi-resolution dataset $\{(\mathbf{X}_a, \mathbf{Y}_a)\}_{a=1}^{\mathcal{A}}$ and the latent functions $\{\mathbf{f}_a^{(k)}\}_{k=1}^{\mathcal{K}_a}$ that are linearly mixed to model each observation process a following a *mixture of experts* approach [19, 22, 34]: $\mathbf{m}_k(\mathbf{x}) = \sum_k^{\mathcal{K}_a} \mathbf{w}_a^{(k)}\mathbf{f}_a^{(k)}(\mathbf{x})$. Every $\mathbf{f}_a^{(k)}$ is a deep DGP of length $\mathcal{L}_a^{(k)}$ and is targeting the same response $\mathbf{Y}_a$. Notice that the gating network is naturally defined by the multiple resolutions and intuitively we want to weigh higher the latent GPs that are closest to the base resolution of interest and that provide the most support. In prediction we achieve this through the predictive

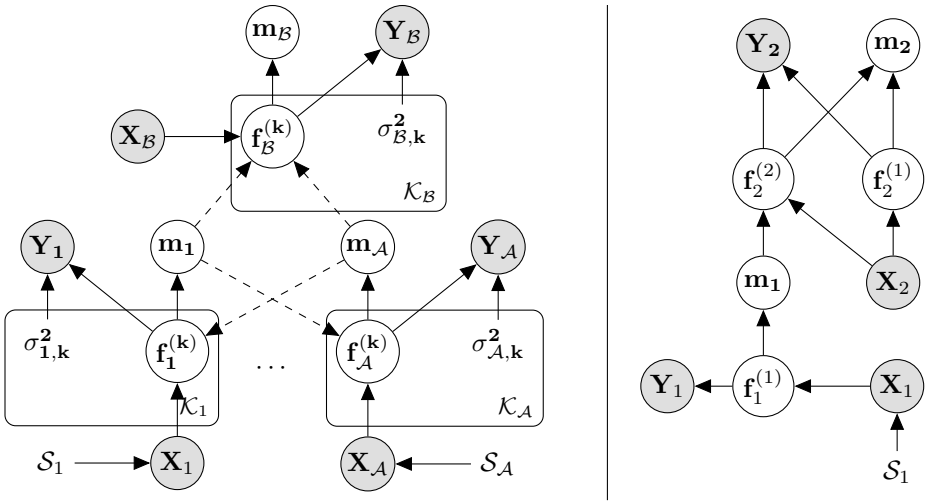


Figure 3: **Left**: General plate diagram of MR-DGP for $\mathcal{A}$ observation processes and the target observation process $Y_\mathcal{B}$ at the base resolution $\mathcal{B}$. **Right**: A specific instantiation of an MR-DGP for the case of 2 resolutions and 2 observation processes with a target process $\mathbf{Y}_2$ as in the *intra*-task multi-resolution NO_2 experiment in Section 4.

variances of the latent functions; taking $\mathbf{V}(\mathbf{x})_a^k$ to be the normalized predictive variance of the k'th function then $w_a^{(k)} = (1 - \mathbf{V}_a^{(k)}) \sum_{i=1}^{k} \mathbf{V}_a^{(i-1)}$. Formally $\mathbf{f}_a^{(k)} \sim \mathcal{GP}(0, \mathbf{K}_a^{(k)}(\mathcal{P}(\mathbf{X}_a), \mathcal{P}_{a,k}^{(1)}(\mathbf{X}_a))$ and $\mathcal{P}_{a,k}^{(1)}$ denotes the parent function of $\mathbf{f}_a^{(k)}$. When $k = 0$ we define $\mathcal{P}_{a,0}^{(\cdot)}$ to be the identity function, and when $k > 1$ then $\mathcal{P}_{a,0}^{(\cdot)}$ denotes the prediction from a specific $\mathbf{m}_a$. For example in Fig. 3 the parent function of $\mathbf{f}_2^{(2)}$, denoted by $\mathcal{P}_{2,2}^{1}$, is $\mathbf{m}_1$. Following Section 3 we link each of the functions to their target resolution through the likelihood. The posterior, $p(\{\{\mathbf{f}_a^{(k)}\}_k^{\mathcal{K}_a}\}_{a=1}^{\mathcal{A}} | \mathbf{Y}, \mathbf{X})$, is now proportional to:

$$\underbrace{\prod_{a=1}^{\mathcal{A}} \prod_{k=1}^{\mathcal{K}_a} \prod_{n=1}^{N_a} \mathcal{N}(\mathbf{Y}_{a,n} | \int_{s=1}^{\mathcal{S}_a} \mathbf{f}_a^{(k)}(\mathbf{X}_{a,n,s})\, d\mathbf{X}_{a,n,s}, \sigma_{a,k}^2 \mathbf{I})}_{\text{MR-DGP Likelihood}} \cdot \underbrace{\prod_{a=1}^{\mathcal{A}} \prod_{k=1}^{\mathcal{K}_a} p(\mathbf{f}_a^{(k)} | \mathcal{P}_{a,k}^{(1)}, \mathbf{X}_a^{(k)})}_{\text{MR-DGP Prior}} \tag{4}$$

The integral is again approximated by discretizing the sampling region with a uniform grid. Our formulation naturally handles biases between the mean of different observation processes and has an appealing and *meaningful* interpretation as each $\mathbf{m}_a$ is modelling a specific observation process.

Augmented Posterior. Due to the nonlinear form of $\mathcal{P}_a^{(1)}$ inside $p(\mathbf{f}_\mathcal{A} | \mathbf{X}_a, \mathcal{P}_a^{(1)})$ the marginal $p(\mathbf{f}_\mathcal{A} | \mathbf{X})$ is in general analytically intractable. Introducing inducing points $\mathbf{U} = \{\mathbf{u_a}\}_{a=1}^{\mathcal{A}}$ for $\mathbf{u_a} \in \mathbb{R}^M$ and locations $\mathbf{Z} = \{\mathbf{Z_a}\}_{a=1}^{\mathcal{A}}$ for $\mathbf{Z_a} \in \mathbb{R}^{M \times D}$, allows $\mathcal{P}(\mathbf{f}_a^{(1)})$ to be propagated through the non-linear kernel function of the GP [5, 29]. The augmented joint is now given by:

$$p(\{\mathbf{Y}_a, \mathbf{f}_a, \mathbf{u}_a\}_{a=1}^{\mathcal{A}}) = (\prod_{a=1}^{\mathcal{A}} \prod_{k=1}^{\mathcal{K}_a} p(\mathbf{Y}_a | \mathbf{X}_a, \mathbf{f}_a^{(k)}) p(\mathbf{f}_a^{(k)} | \mathbf{X}_a^{(k)}, \mathcal{P}_{a,k}^{(1)}, \mathbf{u}_a^{(k)}) p(\mathbf{u}_a^{(k)})) \tag{5}$$

where $p(\mathbf{u}_a^{(k)}) = \mathcal{N}(\mathbf{u}_a^{(k)} | \mathbf{0}, \mathbf{K}_a)$ and $p(\mathbf{f}_a | \mathbf{X}_a, \mathcal{P}_{a,k}^{(1)}, \mathbf{u}_a^{(k)})$ is a Gaussian with mean $\mathbf{f}_a | \mathbf{K}_a(\mathcal{P}_{a,k}^{(1)}(\mathbf{X}_a), \mathbf{Z}_a^{(k)}) \mathbf{K}_a^{-1}(\mathbf{Z}_a^{(k)}, \mathbf{Z}_a^{(k)}) \mathbf{u}_a^{(k)}$, and variance $\mathbf{K}_a(\mathcal{P}_{a,k}^{(1)}(\mathbf{X}_a), \mathcal{P}_{a,k}^{(1)}(\mathbf{X}_a))_a - \mathbf{K}_a(\mathcal{P}_{a,k}^{(1)}(\mathbf{X}_a), \mathbf{Z}_a^{(k)}) \mathbf{K}_a^{-1}(\mathbf{Z}_a^{(k)}, \mathbf{Z}_a^{(k)}) \mathbf{K}_a(\mathbf{Z}_a^{(k)}, \mathcal{P}_{a,k}^{(1)}(\mathbf{X}_a))$ which is the standard noise-free GP predictive distribution.

Inference. Following [25] we construct an approximate posterior $q(\{\mathbf{f}_a, \mathbf{u}_a\}_{a=1}^{\mathcal{A}})$ that maintains the dependency structure between the latent functions:

$$q(\{\mathbf{f}_a, \mathbf{u}_a\}_{a=1}^{\mathcal{A}}) = \prod_{a=1}^{\mathcal{A}} \prod_{k=1}^{\mathcal{K}_a} p(\mathbf{f}_a^{(k)} | \mathbf{X}_a^{(k)}, \mathcal{P}(\mathbf{f}_a^{(k)}), \mathbf{u}_a^{(k)}) q(\mathbf{u}_a^{(k)}) \tag{6}$$

Algorithm 2 Inference procedure for MR-DGP

Input: S multi-resolution datasets $\{(\mathbf{X}_s, \mathbf{Y}_s)\}_{s=1}^{S}$, initial parameters θ_0,
procedure MARGINAL($\mathbf{f}_a^{(k)}$,$\mathbf{X}$, l)
  if $l = \mathcal{L}_a^{(k)}$ **then**
    return $q(\mathbf{f}_a^{(k)}|\mathbf{X})$
  end if
  $q(\mathcal{P}(\mathbf{f}_a^{(k)})|\mathbf{X}) \leftarrow$ MARGINAL $(\mathcal{P}(\mathbf{f}_a^{(k)}), \mathbf{X}, l+1)$
  return $\frac{1}{S}\sum_{s=1}^{S} p(\mathbf{f}_a^{(k)}|\mathbf{f}^{(s)}, \mathbf{X}))$ where $\mathbf{f}^{(s)} \sim q(\mathcal{P}(\mathbf{f}_a^{(k)})|\mathbf{X})$
end procedure
$\theta_1 \leftarrow \arg\min_{\theta} \left[\sum_{a=1}^{\mathcal{A}}\sum_{k=1}^{\mathcal{K}_a} \mathbb{E}_{\text{MARGINAL}(\mathbf{f}_a^{(k)},\mathbf{X}_a,0)}\left[\log p(\mathbf{Y}_a|\mathbf{f}_a^{(k)}, \mathbf{X}_a, \theta)\right] + \mathcal{KL}(q(\mathbf{u})||p(\mathbf{u}))\right]$

where $q(\mathbf{u}_a) = \mathcal{N}(\mathbf{m}_a, \mathbf{S}_a)$ is a free-form Gaussian. The ELBO is:

$$\mathcal{L}_{\text{MR-DGP}} = \underbrace{\sum_{a=1}^{\mathcal{A}}\sum_{k=1}^{\mathcal{K}_a} \mathbb{E}_{q(\mathbf{f}_a^{(k)})}\left[\log p(\mathbf{Y}_a|\mathbf{X}_a, \mathbf{f}_a^{(k)})\right]}_{\mathcal{L}_{\text{ell}}} + \underbrace{\sum_{a=1}^{\mathcal{A}}\sum_{k=1}^{\mathcal{K}_a} \mathbb{E}_{q(\mathbf{u}_a^{(k)})}\left[\log \frac{p(\mathbf{u}_a^{(k)})}{q(\mathbf{u}_a^{(k)})}\right]}_{-\mathcal{KL}(q(\mathbf{U})||p(\mathbf{U}))}$$

where the likelihood factorizes across observation processes. For each likelihood component the marginal $q(\mathbf{f}_a^{(k)})$ is required and the posterior of a specific latent function only depends on the parent functions in the previous layer. For general $\mathbf{f}_a^{(k)}$ the posterior is:

$$q(\mathbf{f}_a^{(k)}) = \int q(\mathbf{f}_a^{(k)}|\mathcal{P}^{(1)}(\mathbf{f}_a^{(k)})) \prod_{l=1}^{\mathcal{L}-1} q(\mathcal{P}^{(l)}(\mathbf{f}_a^{(k)})|\mathcal{P}^{(l+1)}(\mathbf{f}_a^{(k)}))\, d\mathcal{P}^{(1)}(\mathbf{f}_a^{(k)}) \cdots d\mathcal{P}^{(\mathcal{L})}(\mathbf{f}_a^{(k)}) \quad (7)$$

Since $\mathcal{P}^{(\mathcal{L})}(\mathbf{f}_a^{(k)})$ is a Gaussian, sampling is straightforward and the integral is approximated by Monte Carlo and we use the reparametization trick to draw samples from the variational posteriors [10]. The inference procedure is given in Alg. 2 and has an epoch time complexity of $\mathcal{O}(\sum_a^{\mathcal{A}}\sum_k^{\mathcal{K}} N_a M_a^2 \mathcal{L}_a^{(k)})$.

Predictive Density. To predict at $\mathbf{x}^* \in \mathbb{R}^D$ for process k we approximate the predictive density $q(\mathbf{m}_a^*)$ by sampling from the variational posteriors:

$$q(\mathbf{m}_a^*) = \int q(\mathbf{m}_a^*|\mathbf{f}_a^{(1)}, \cdots, \mathbf{f}_a^{(k)}) \prod_{k}^{\mathcal{K}_a} q(\mathbf{f}_a^{(k)})\, d\mathbf{f}_a^1 \cdots d\mathbf{f}_a^{(\mathcal{K})} \approx \frac{1}{S}\sum_{s=1}^{S} q(\mathbf{m}_a^*|\mathbf{f}_a^{(1)(s)}, \cdots, \mathbf{f}_a^{(k)(s)}) \quad (8)$$

where $\mathbf{f}_\cdot^{(\cdot)(s)} \sim q(\mathbf{f}_\cdot^{(\cdot)})$.

5 Related Work

Gaussian processes (GPs) are the workhorse for spatio-temporal modelling in spatial statistics [8] and in machine learning [23] with the direct link between multi-task GPs and Linear Models of Coregionalisation (LCM) reviewed by Alvarez et al. [2]. Heteroscedastic GPs [14] and recently proposed deeper compositions of GPs for the multi-fidelity setting [4, 20, 21] assume that all observations are of the same resolution. In spatial statistics the related *change of support* problem has been approached through Markov Chain Monte Carlo approximations and domain discretizations [7, 8]. A recent exception to this is the work by Smith et al. [27] that solves the integral for squared exponential kernels but only considers observations from one resolution and cannot handle additional input features. Finally, we note that the multiresolution GP work by Fox and Dunson [6] defines a DGP construction for non-stationary models that is more akin to multi-scale modelling [32] which typically focuses on learning multiple kernel lengthscales to explain both broad and fine variations in the underlying process and hence cannot handle multi-resolution observations .

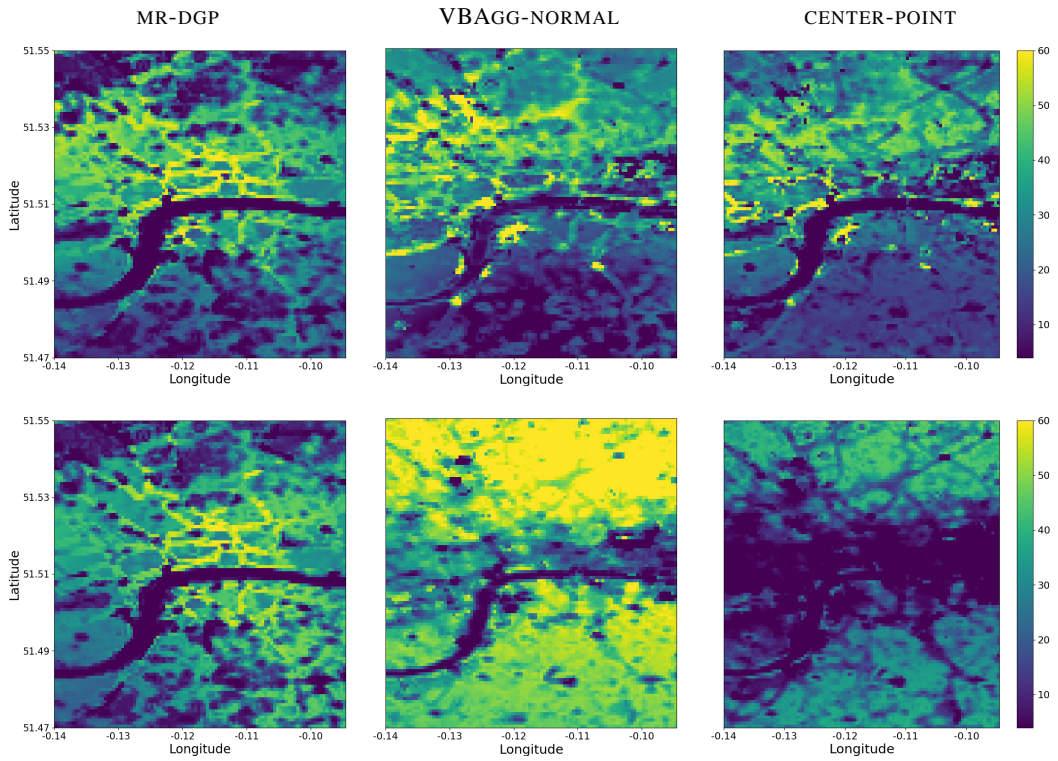


Figure 4: Spatio-temporal estimation and forecasting of NO_2 levels in London. **Top Row**: Spatial slices from MR-GPRN, VBAGG-NORMAL and CENTER-POINT respectively at 09/02/2019 11:00:00 using observations from both LAQN (base resolution $\mathcal{B}$) and the satellite model (low spatial resolution). **Bottom Row**: Spatial slices at the base resolution from the same models at 09/03/2019 17:00:00 where *only* observations from the satellite model are present.

6 Experiments

We demonstrate and evaluate the MRGPs on synthetic experiments and the challenging problem of estimating and forecasting air pollution in the city of London. We compare against VBAGG-NORMAL [13] and two additional baselines. The first, CENTER-POINT , is a GPRN modified to support multi-resolution data by taking the center point of each aggregation region as the input. The second, MR-CASCADE is a MR-DGP but instead of a tree structured DGP as in Fig. 3 we construct a cascade to illustrate the benefits of the tree composition and the mixture of experts approach of MR-DGP. Experiments are coded[1] in *TensorFlow* and we provide additional analysis and experiments in the Supp.

Dependent observation processes: We provide additional details of the dependent observation processes experiment in the left of Fig. 2 in the Supp.

Biased observation processes:. To demonstrate the ability of MR-DGP in handling biases across observation processes we construct 3 datasets from the function $\mathbf{y} = s \cdot 5\sin(\mathbf{x})^2 + 0.1\epsilon$ where $\epsilon \sim \mathcal{N}(0, 1)$. The first $\mathbf{X}_1, \mathbf{Y}_1$ is at resolution $\mathcal{S}_1 = 1$ in the range x=[7,12] with a scale $s = 1$. The second is at resolution of $\mathcal{S}_2 = 5$ between x=[-10, 10] with a scale $s = 0.5$ and lastly the third is at resolution of $\mathcal{S}_3 = 5$ x=[10, 20] with a scale $s = 0.3$. The aim is to predict $\mathbf{y}$ across the range [-10, 20] and the results are shown in Table 2 and Fig. 2. MR-DGP significantly outperforms all of the four alternative approaches as it is learning a forward *mapping* between observation processes, e.g. $\mathbf{f}_2^{(2)}$ in Fig. 3, and is not just trusting and propagating the mean.

[1]Codebase and datasets to reproduce results will be made available on publication

Table 1: *Inter*-task multi-resolution. Missing data predictive MSE on PM25 from MR-GPRN, MR-DGP and baseline CENTER-POINT for 4 different aggregation levels of PM10. VBAGG-NORMAL is inapplicable in this experiment as it is a single-task approach and can be seen as a special case of MR-GPRN without composite weight corrections.

Model	PM10 Resolution			
	2 Hours	5 Hours	10 Hours	24
CENTER-POINT	12.72 ± 1.29	13.56 ± 1.88	14.99 ± 3.63	16.21 ± 2.54
MR-DGP	13.22 ± 0.89	13.24 ± 1.6	14.41 ± 1.42	14.59 ± 1.18
MR-GPRN	$\mathbf{10.54} \pm 1.15$	$\mathbf{10.21} \pm 0.92$	$\mathbf{10.11} \pm 1.13$	$\mathbf{10.17} \pm 0.68$

Table 2: *Intra*-task multi-resolution. **Left**: Predicting NO_2 across London (Fig. 4). **Right**: Synthetic experiment results (Fig. 2) with three observations processes and scaling bias.

Model	RMSE	MAPE
CENTER-POINT	18.74 ± 12.65	0.65 ± 0.21
VBAGG-NORMAL	15.98 ± 9.5	0.72 ± 0.47
MR-GPRN	12.95 ± 7.78	0.51 ± 0.33
MR-DGP	$\mathbf{8.7} \pm 5.51$	$\mathbf{0.36} \pm 0.17$

Model	RMSE	MAPE
MR-CASCADE	1.61	5.87
VBAGG-NORMAL	1.8	2.53
MR-GPRN	1.68	2.96
MR-DGP	**0.3**	**3.83**

Training. When training both MR-GPRN and VBAGG-NORMAL we first jointly optimize the variational and hyper parameters while keeping the likelihood variances fixed and then jointly optimize all parameters together. For MR-DGP we first optimize layer by layer and then jointly optimize all parameters together, see Appendix. We find that this helps to avoid early local optima.

***Inter*-task multi-resolution: modelling of PM10 and PM25 in London**: In this experiment we consider the case of having multiple tasks with different resolutions. We jointly model PM10 and PM25 at a specific location in London. The site we consider is *RB7*, from the LAQN network, in the date range 18/06/2018 to 28/06/2018. At this location we have hourly data from both PM10 and PM25. To simulate having multiple resolutions we construct 2, 5, 10 and 24 hour aggregations of PM10 and remove a 2 day region of pm25 which is the test region. The results are shown in Table 1.

***Intra*-task multi-resolution: spatio-temporal modelling of NO_2 in London**: In this experiment we consider the case of a single task but with multiple multi-resolution observation processes. First we use observations coming from ground point sensors from the London Air Quality Network (LAQN). These sensors provide hourly readings of NO_2. Secondly we use observations arising from a global satellite model [16] that provide hourly data at a spatial resolution of 7km $\times$ 7km and provide 48 hour forecasts. We train on both the LAQN and satellite observations from 19/02/2018-20/02/2018 and just the satellite sensors from 20/02/2018-21/02/2018. We then predict at the resolution of the LAQN sensors in the latter date range. To calculate errors we predict for each LAQN sensor site, and find the average and standard deviation across all sites.

We find that MR-DGP is able to substantially outperform both VBAGG-NORMAL, MR-GPRN and the baselines, Table 2, as it is learning the forward mapping between the spatially low-resolution process and the high resolution LAQN reference grade sensors, while handling scaling biases. This is further highlighted in the bottom of Fig. 4 where MR-DGP is able to predict and retain high resolution structure based only on satellite observations whereas VBAGG-NORMAL and CENTER-BASELINE completely over-smooth.

7 Conclusion

We offer a framework for evidence integration when observation processes can have varying *inter*- and *intra-task* sampling resolutions, dependencies, and different signal to noise ratios. Our motivation comes from a challenging and impactful problem of hyper-local air quality prediction in the city of London, while the underlying multi-resolution multi-sensor problem is general and pervasive across modern spatio-temporal settings and beyond. We proposed both shallow mixtures and deep learning

models that generalise and outperform the prior art, correct for posterior contraction, and can handle biases in observation processes such as discrepancies in the mean. Further directions now open up to robustify the multi-resolution framework against outliers and against further model misspecification by exploiting ongoing advances in generalized variational inference [11]. Finally an open challenge remains on developing continuous model constructions that avoid domain discretization, as in [1], for multivariate settings.

8 Acknowledgements

Oliver Hamelijnck, Kangrui Wang and Theodoros Damoulas are funded by the Lloyds Register Foundation programme on Data Centric Engineering through the London Air Quality project. This work was furthermore supported by The Alan Turing Institute for Data Science and AI under EPSRC grant EP/N510129/1 in collaboration with the Greater London Authority. We would like to thank Libby Rogers for her help with processing the air pollution data.

References

[1] Adelsberg, M. and Schwantes, C. (2018). Binned kernels for anomaly detection in multi-timescale data using Gaussian processes. In *Proceedings of the KDD 2017: Workshop on Anomaly Detection in Finance*, Proceedings of Machine Learning Research.

[2] Alvarez, M. A., Rosasco, L., Lawrence, N. D., et al. (2012). Kernels for vector-valued functions: A review. *Foundations and Trends® in Machine Learning*, 4(3):195–266.

[3] Caruana, R. (1997). Multitask learning. *Machine learning*, 28(1):41–75.

[4] Cutajar, K., Pullin, M., Damianou, A., Lawrence, N., and González, J. (2019). Deep Gaussian Processes for Multi-fidelity Modeling. *arXiv e-prints*, page arXiv:1903.07320.

[5] Damianou, A. and Lawrence, N. (2013). Deep Gaussian processes. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*.

[6] Fox, E. B. and Dunson, D. B. (2012). Multiresolution Gaussian processes. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*.

[7] Fuentes, M. and Raftery, A. E. (2005). Model evaluation and spatial interpolation by Bayesian combination of observations with outputs from numerical models. *Biometrics*.

[8] Gelfand, A., Fuentes, M., Guttorp, P., and Diggle, P. (2010). *Handbook of Spatial Statistics*. Chapman & Hall/CRC Handbooks of Modern Statistical Methods. Taylor & Francis.

[9] Hensman, J., Fusi, N., and Lawrence, N. D. (2013). Gaussian processes for big data. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*.

[10] Kingma, D. P. and Welling, M. (2014). Auto-encoding variational bayes. In *International Conference for Learning Representations*.

[11] Knoblauch, J., Jewson, J., and Damoulas, T. (2019). Generalized Variational Inference. *arXiv e-prints*, page arXiv:1904.02063.

[12] Krauth, K., Bonilla, E. V., Cutajar, K., and Filippone, M. (2017). AutoGP: Exploring the Capabilities and Limitations of Gaussian Process Models. *In Conference on Uncertainty in Artificial Intelligence (UAI)*.

[13] Law, H. C. L., Sejdinovic, D., Cameron, E., Lucas, T. C., Flaxman, S., Battle, K., and Fukumizu, K. (2018). Variational learning on aggregate outputs with Gaussian processes. *Advances in Neural Information Processing Systems (NeurIPS)*.

[14] Lázaro-Gredilla, M. and Titsias, M. K. (2011). Variational heteroscedastic Gaussian process regression. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*.

[15] Lyddon, S. P., Holmes, C. C., and Walker, S. G. (2019). General Bayesian updating and the loss-likelihood Bootstrap. *Biometrika*.

[16] Marécal, V., Peuch, V.-H., Andersson, C., Andersson, S., Arteta, J., Beekmann, M., Benedictow, A., Bergström, R., Bessagnet, B., Cansado, A., Chéroux, F., Colette, A., Coman, A., Curier, R. L., Denier van der Gon, H. A. C., Drouin, A., Elbern, H., Emili, E., Engelen, R. J., Eskes, H. J., Foret, G., Friese, E., Gauss, M., Giannaros, C., Guth, J., Joly, M., Jaumouillé, E., Josse, B., Kadygrov, N., Kaiser, J. W., Krajsek, K., Kuenen, J., Kumar, U., Liora, N., Lopez, E., Malherbe, L., Martinez, I., Melas, D., Meleux, F., Menut, L., Moinat, P., Morales, T., Parmentier, J., Piacentini, A., Plu, M., Poupkou, A., Queguiner, S., Robertson, L., Rouïl, L., Schaap, M., Segers, A., Sofiev, M., Tarasson, L., Thomas, M., Timmermans, R., Valdebenito, A., van Velthoven, P., van Versendaal, R., Vira, J., and Ung, A. (2015). A regional air quality forecasting system over europe: the macc-ii daily ensemble production. *Geoscientific Model Development*.

[17] Moreno-Muñoz, P., Artés-Rodríguez, A., and Álvarez, M. A. (2018). Heterogeneous multi-output Gaussian process prediction. In *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*.

[18] Nguyen, T. and Bonilla, E. (2013). Efficient variational inference for Gaussian process regression networks. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*.

[19] Nguyen, T. and Bonilla, E. (2014). Fast allocation of Gaussian process experts. In *Proceedings of the 31st International Conference on Machine Learning*.

[20] Perdikaris, P., Raissi, M., Damianou, A., D. Lawrence, N., and Karniadakis, G. (2017). Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*.

[21] Perdikaris, P., Venturi, D., Royset, J. O., and Karniadakis, G. E. (2015). Multi-fidelity modelling via recursive co-kriging and Gaussian–markov random fields. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*.

[22] Rasmussen, C. E. and Ghahramani, Z. (2002). Infinite mixtures of Gaussian process experts. In *Advances in Neural Information Processing Systems 14*.

[23] Rasmussen, C. E. and Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press.

[24] Ribatet, M. (2012). Bayesian inference from composite likelihoods, with an application to spatial extremes. In *Statistica Sinica 22: 813–845*.

[25] Salimbeni, H. and Deisenroth, M. (2017). Doubly stochastic variational inference for deep Gaussian processes. In *Advances in Neural Information Processing Systems 30*.

[26] Serban, I. V., Klinger, T., Tesauro, G., Talamadupula, K., Zhou, B., Bengio, Y., and Courville, A. (2017). Multiresolution recurrent neural networks: An application to dialogue response generation. In *Thirty-First AAAI Conference on Artificial Intelligence*.

[27] Smith, M. T., Alvarez, M. A., and Lawrence, N. D. (2018). Gaussian process regression for binned data. *arXiv e-prints*.

[28] Stoehr, J. and Friel, N. (2015). Calibration of conditional composite likelihood for Bayesian inference on gibbs random fields. In *Artificial Intelligence and Statistics*.

[29] Titsias, M. (2009). Variational learning of inducing variables in sparse Gaussian processes. In *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*.

[30] Varin, C., Reid, N., and Firth, D. (2011a). An overview of composite likelihood methods. *Statist. Sinica*.

[31] Varin, C., Reid, N., and Firth, D. (2011b). An overview of composite likelihood methods. *Statistica Sinica*.

[32] Walder, C., Kim, K. I., and Schölkopf, B. (2008). Sparse multiscale Gaussian process regression. In *Proceedings of the 25th international conference on Machine learning*.

[33] Wilson, A. G., Knowles, D. A., and Ghahramani, Z. (2012). Gaussian process regression networks. In *Proceedings of the 29th International Conference on Machine Learning*.

[34] Yuan, C. and Neubauer, C. (2009). Variational mixture of Gaussian process experts. In *Advances in Neural Information Processing Systems 21*.

A Definitions

In this section we provide additional mathematical definitions needed to support Sec. C.2.

$$\begin{aligned}\mathbb{E}[\exp(\mathbf{W}_{ni})] &= \int \mathcal{N}(\mathbf{W}_{ni}|\mu_{\mathbf{W}_{ni}}, \Sigma_{\mathbf{W}_{ni}}) \exp(\mathbf{W}_{ni})\, d\mathbf{W}_{ni} \\ &= C_1 \cdot \int \exp(-\frac{1}{2}(\mathbf{W}_{ni} - \mu_{\mathbf{W}_{ni}})^T \Sigma^{-1}_{\mathbf{W}_{ni}}(\mathbf{W}_{ni} - \mu_{\mathbf{W}_{ni}}) + \mathbf{W}_{ni})\, d\mathbf{W}_{ni} \\ &= C_1 \cdot \int \exp(-\frac{1}{2}(\mathbf{W}_{ni} - (\mu_{\mathbf{W}_{ni}} + \Sigma_{\mathbf{W}_{ni}}))^T \Sigma^{-1}_{\mathbf{W}_{ni}}(\mathbf{W}_{ni} - (\mu_{\mathbf{W}_{ni}} + \Sigma_{\mathbf{W}_{ni}})))) \\ &\cdot \exp(\mu_{\mathbf{W}_{ni}} + \frac{1}{2}\Sigma_{\mathbf{W}_{ni}})\, d\mathbf{W}_{ni} \\ &= \exp(\mu_{\mathbf{W}_{ni}} + \frac{1}{2}\Sigma_{\mathbf{W}_{ni}})\end{aligned} \tag{9}$$

where $C_1 = ((2\pi)^{\frac{1}{2}}|\Sigma_{\mathbf{W}_{ni}}|^{\frac{1}{2}})^{-1}$.

Similarly we can derive the expectation of the square forms:

$$\mathbb{E}[\exp(\mathbf{W}_{ni})^T \exp(\mathbf{W}_{ni})] = \exp(2 \cdot (\mu_{\mathbf{W}_{ni}} + \Sigma_{\mathbf{W}_{ni}})) \tag{10}$$

and :

$$\begin{aligned}\mathbb{E}[\exp(\mathbf{W}_{ni})^T \Sigma \exp(\mathbf{W}_{ni})] &= \int \mathcal{N}(\mathbf{W}_{ni}|\mu_{\mathbf{W}_{ni}}, \Sigma_{\mathbf{W}_{ni}}) \exp(\mathbf{W}_{ni})^T \Sigma \exp(\mathbf{W}_{ni})\, d\mathbf{W}_{ni} \\ &= \exp(\mu_{\mathbf{W}_{ni}} + \Sigma_{\mathbf{W}_{ni}})^T \Sigma \exp(\mu_{\mathbf{W}_{ni}} + \Sigma_{\mathbf{W}_{ni}})\end{aligned} \tag{11}$$

B MR-DGP: Variational Lower Bound

In this section we provide the derivation of the variational lower bound for MR-DGP. Following [25] we construct an approximate posterior that maintains the dependency structure between layers

$$q(\{\mathbf{f}_a, \mathbf{u}_a\}_{a=1}^{\mathcal{A}}) = \prod_{a=1}^{\mathcal{A}} p(\mathbf{f}_a|\{\mathbf{X}_a^{(k)}, \mathbf{f}_a^{(k)}, \mathbf{u}_a^{(k)}\}_{k=1}^{\mathcal{K}_a}) q(\{\mathbf{u}_a^{(k)}\}_{k=1}^{\mathcal{K}}) \tag{12}$$

where $q(\mathbf{u}_a) = \mathcal{N}(\mathbf{m}_a, \mathbf{S}_a)$ is a free-form Gaussian. The evidence lowerbound (ELBO), which lower bounds the log marginal likelihood $\log p(\mathbf{Y}|\mathbf{X})$, is

$$\begin{aligned}\mathcal{L} &= \mathbb{E}_{q(\{\mathbf{f}_a, \mathbf{u}_a\}_{a=1}^{\mathcal{A}})}\left[\log \frac{p(\{\mathbf{Y}_a, \mathbf{f}_a, \mathbf{u}_a\}_{a=1}^{\mathcal{A}})}{q(\{\mathbf{f}_a, \mathbf{u}_a\}_{a=1}^{\mathcal{A}})}\right] \\ &= \mathbb{E}\left[\log \frac{(\prod_{a=1}^{\mathcal{A}} \prod_{k=1}^{\mathcal{K}_a} p(\mathbf{Y}_a|\mathbf{X}_a, \mathbf{f}_a^{(k)})) \cdot \prod_{a=1}^{\mathcal{A}} (p(\mathbf{f}_a|\{\mathbf{X}_a^{(k)}, \mathbf{f}_a^{(k)}, \mathbf{u}_a^{(k)}\}_{k=1}^{\mathcal{K}_a}) p(\{\mathbf{u}_a^{(k)}\}_{k=1}^{\mathcal{K}}))}{\prod_{a=1}^{\mathcal{A}} p(\mathbf{f}_a|\{\mathbf{X}_a^{(k)}, \mathbf{f}_a^{(k)}, \mathbf{u}_a^{(k)}\}_{k=1}^{\mathcal{K}_a}) q(\{\mathbf{u}_a^{(k)}\}_{k=1}^{\mathcal{K}})}\right].\end{aligned} \tag{13}$$

Cancelling the relevant terms inside the logarithm we get

$$\begin{aligned}\mathcal{L}_{\text{MR-DGP}} &= \mathbb{E}_{q(\{\mathbf{f}_a,\mathbf{u}_a\}_{a=1}^{\mathcal{A}})}\left[\log \frac{(\prod_{a=1}^{\mathcal{A}}\prod_{k=1}^{\mathcal{K}_a} p(\mathbf{Y}_a|\mathbf{X}_a,\mathbf{f}_a^{(k)}))\cdot\prod_{a=1}^{\mathcal{A}} p(\{\mathbf{u}_a^{(k)}\}_{k=1}^{\mathcal{K}})}{\prod_{a=1}^{\mathcal{A}} q(\{\mathbf{u}_a^{(k)}\}_{k=1}^{\mathcal{K}})}\right] \\ &= \mathbb{E}\left[\log\prod_{a=1}^{\mathcal{A}}\prod_{k=1}^{\mathcal{K}_a} p(\mathbf{Y}_a|\mathbf{X}_a,\mathbf{f}_a^{(k)})\right] + \mathbb{E}\left[\log\frac{\prod_{a=1}^{\mathcal{A}} p(\{\mathbf{u}_a^{(k)}\}_{k=1}^{\mathcal{K}})}{\prod_{a=1}^{\mathcal{A}} q(\{\mathbf{u}_a^{(k)}\}_{k=1}^{\mathcal{K}})}\right] \\ &= \underbrace{\sum_{a=1}^{\mathcal{A}}\mathbb{E}_{q(\mathbf{f}_a)}\left[\log p(\mathbf{Y}_a|\mathbf{X}_a,\mathbf{f}_a)\right]}_{\text{ELL}} + \underbrace{\sum_{a=1}^{\mathcal{A}}\mathbb{E}_{q(\mathbf{u}_a)}\left[\log\frac{p(\mathbf{u}_a)}{q(\mathbf{u}_a)}\right]}_{-\mathcal{KL}(q\mathbf{U}||p(\mathbf{u}))}\end{aligned}$$

C MR-GPRN: Variational Lower Bound

In this section we provide the full derivation of the variational lower bound of MR-GPRN. Recall that we have Q global latent processes $\mathbf{f}_q \sim \mathcal{N}(0,\mathbf{K}_\mathbf{q})$ and $P\times Q$ task specific latent functions $\mathbf{W}_{p,q} \sim \mathcal{N}(0,\mathbf{K}_{\mathbf{p},\mathbf{q}})$. To allow for computationally efficient inference we introduce inducing points [29] for all latent functions. For $\mathbf{f}$: $\mathbf{U}=\{\mathbf{u}_\mathbf{q}\}_{q=1}^{Q=1}$ where $\mathbf{u}_\mathbf{q}\in\mathbb{R}^M$ at locations $\mathbf{Z}^{(f)}=\{\mathbf{Z}_\mathbf{q}\}_{q=1}^{Q}$ for $\mathbf{Z}_\mathbf{q}\in\mathbb{R}^{M,D}$. For $\mathbf{W}$: $\mathbf{V}=\{\mathbf{v}_{\mathbf{p},\mathbf{q}}\}_{p,q=1}^{PQ=1}$ where $\mathbf{v}_{\mathbf{p},\mathbf{q}}\in\mathbb{R}^M$ at locations $\mathbf{Z}^{(w)}=\{\mathbf{Z}_{\mathbf{p},\mathbf{q}}\}_{p,q=1}^{PQ}$ for $\mathbf{Z}_{\mathbf{p},\mathbf{q}}\in\mathbb{R}^{M,D}$.

Following [9] we keep $\mathbf{U},\mathbf{V}$ explicit in our approximate posterior. The goal is now to learn the augmented posterior $p(\mathbf{f},\mathbf{W},\mathbf{UV}|\mathbf{X},\mathbf{Y},\theta)$. To do so we introduce our approximate posterior $q(\mathbf{f},\mathbf{W},\mathbf{U},\mathbf{V}) = p(\mathbf{f}|\mathbf{U})p(\mathbf{W}|\mathbf{v})q(\mathbf{U},\mathbf{V})$ where $q(\mathbf{U},\mathbf{V}) = \sum_{k=1}^{K}\pi_k\prod_{p=1}^{P}\mathcal{N}(\mathbf{m}_p^{(f)},\mathbf{S}_p^{(f)})\prod_{p,q=1}^{PQ}\mathcal{N}(\mathbf{m}_{pq}^{(w)},\mathbf{S}_{pq}^{(w)})$ which is a mixture of Gaussians and defines the variational parameters to learn. The conditionals $p(\mathbf{f}|\mathbf{U})$ and $p(\mathbf{W}|\mathbf{v})$ are given by the standard noise free GP prediction.

Variational inference turns into an optimisation problem where the objective function is the evidence lower bound (ELBO). Our ELBO is:

$$\mathcal{L}(q) = \sum_{a=1}^{\mathcal{A}}\alpha\mathbb{E}_q\left[\log p(\mathbf{Y}_a|\mathbf{f},\mathbf{W},\mathbf{X}_a)\right] + KL(q(\mathbf{U},\mathbf{V})||p(\mathbf{U},\mathbf{V})), \tag{14}$$

The subsequent sections derive the forms of both the expected log likelihood (ELL) and the KL term.

C.1 MR-GPRN: Closed Form Expected Log Likelihood

We now derive the closed form expected log likelihood (ELL) from Eq. 14. The ELL is:

$$\mathcal{L}_{ell} = \sum_{a=1}^{\mathcal{A}}\alpha_k E_q\left[\log\mathcal{N}(\mathbf{Y}_a|\frac{1}{R_a}\sum_{r=1}^{R_a}(\mathbf{Wf})(r),\sigma_a^2)\right] \tag{15}$$

where each of the components can now be dealt with separately. Dealing with component a:

$$
\begin{aligned}
\mathcal{L}_{ell_a} &= \sum_{n=1}^{N_a}\sum_{k=1}^{K}\pi_k\sum_{p=1}^{P}\int q_k(\mathbf{f}_n)q_k(\mathbf{W}_{ni})\log\mathcal{N}(\mathbf{Y}_{ni}^{(a)}|\frac{1}{R_a}\sum_{r}^{R_a}\mathbf{W}(r)_{ni}\mathbf{f}(r)_n,\sigma_a^2)\,d\mathbf{W}_{ni}\,d\mathbf{f}_n \\
&= C_1 + C_2\sum_{n=1}^{N_a}\sum_{k=1}^{K}\pi_k\sum_{p=1}^{P}\int q_k(\mathbf{f}_n)q_k(\mathbf{W}_{ni})\cdot \\
&\qquad\qquad (\mathbf{Y}_{ni}^{(a)} - \frac{1}{R_a}\sum_{r}^{R_a}\mathbf{W}(r)_{ni}\mathbf{f}(r)_n)^T(\mathbf{Y}_{ni}^{(a)} - \frac{1}{R_a}\sum_{r}^{R_a}\mathbf{W}(r)_{ni}\mathbf{f}(r)_n)\,d\mathbf{W}_{ni}\,d\mathbf{f}_n \\
&= C_1 + C_2\sum_{n=1}^{N_a}\sum_{k=1}^{K}\pi_k\sum_{p=1}^{P}\mathbb{E}_q\left[(\mathbf{Y}_{ni}^{(a)})^T\mathbf{Y}_{ni}^{(a)}\right] - \mathbb{E}_q\left[(\frac{1}{R_a}\sum_{r}^{R_a}\mathbf{W}(r)_{ni}\mathbf{f}(r)_n)^T\mathbf{Y}_{ni}^{(a)}\right] - \\
&\quad \mathbb{E}_q\left[(\mathbf{Y}_{ni}^{(a)})^T(\frac{1}{R_a}\sum_{r}^{R_a}\mathbf{W}(r)_{ni}\mathbf{f}(r)_n)\right] + \mathbb{E}_q\left[(\sum_{r}^{R_a}\mathbf{W}(r)_{ni}\mathbf{f}(r)_n)^T(\sum_{r}^{R_a}\mathbf{W}(r)_{ni}\mathbf{f}(r)_n)\right]
\end{aligned}
$$

We now deal with each of the expectations separately.

C.1.1 ELL: 1st Term

The first expectation does not contain $\mathbf{f}$ or $\mathbf{W}$ and so the expectations can be dropped:

$$
\mathbb{E}_q\left[(\mathbf{Y}_{ni}^{(a)})^T\mathbf{Y}_{ni}^{(a)}\right] = {\mathbf{Y}_{ni}^{(a)}}^T\mathbf{Y}_{ni}^{(a)} \tag{16}
$$

C.1.2 ELL: 2nd and 3rd Term

In the 2nd and 3rd terms the expectation is brought inside the sum and applied to $\mathbf{f}$ and $\mathbf{W}$ separately:

$$
\mathbb{E}_q\left[(\frac{1}{R_a}\sum_{r}^{R_a}\mathbf{W}(r)_{ni}\mathbf{f}(r)_n)^T\mathbf{Y}_{ni}^{(a)}\right] = (\frac{1}{R_a}\sum_{r}^{R_a}\mathbb{E}_{q(\mathbf{W}_{ni})}\left[\mathbf{W}(r)_{ni}\right]\mathbb{E}_{q(\mathbf{f}_n)}\left[\mathbf{f}(r)_n)^T\right]\mathbf{Y}_{ni}^{(a)} \tag{17}
$$

C.1.3 ELL: 4th Term

In the last expectation we have a product of sums that is expanded into a double sum over a and b. There is now two cases: when $a = b$ inside the sum $\mathbf{f}$ and $\mathbf{W}$ will appear in square forms, when $a \neq b$ the expectation can be treated as in Sec. C.1.2.

$$
\begin{aligned}
&\mathbb{E}_q\big[(\sum_{r}^{R_a}\mathbf{W}(r)_{ni}\mathbf{f}(r)_n)^T(\sum_{r}^{R_a}\mathbf{W}(r)_{ni}\mathbf{f}(r)_n)\big] \\
&\qquad\qquad = \frac{1}{{R_a}^2}\sum_{a}^{R_a}\sum_{b}^{R_a}\mathbb{E}_{q(\mathbf{f}_n)q(\mathbf{W}_{ni})}\left[\mathbf{f}(a)_n^T\mathbf{W}(a)_{ni}^T\mathbf{W}(b)_{ni}\mathbf{f}(b)_n\right]
\end{aligned} \tag{18}
$$

case $a = b$:

$$
\begin{aligned}
&\int q(\mathbf{f}_n)q(\mathbf{W}_{ni})\mathbf{f}(a)_n^T\mathbf{W}(a)_{ni}^T\mathbf{W}(a)_{ni}\mathbf{f}(a)_n\,d\mathbf{f}_n\,d\mathbf{W}_{ni} \\
&\qquad = \int q(\mathbf{f}_n^a)q(\mathbf{W}_{ni}^a)\mathbf{f}(a)^T\mathbf{W}(a)^T\mathbf{W}(a)_{ni}\mathbf{f}(a)\,d\mathbf{f}_a\,d\mathbf{W}_{ni}^a \\
&\qquad = Tr[\Sigma_{\mathbf{W}^a}\Sigma_{\mathbf{f}_n^a}] + \mu_{\mathbf{f}_n^a}^T\Sigma_{\mathbf{W}_{ni}^a}\mu_{\mathbf{f}_n^a} + \mu_{\mathbf{W}_{ni}^a}^T\Sigma_{\mathbf{f}_n^a}\mu_{\mathbf{W}_{ni}^a} + \mu_{\mathbf{f}_a}^T\mu_{\mathbf{W}_{ni}^a}^T\mu_{\mathbf{W}_{ni}^a}\mu_{\mathbf{f}_n^a}
\end{aligned} \tag{19}
$$

case $a \neq b$:

$$\begin{aligned}
&\int q(\mathbf{f}_n)\mathbf{f}(a)_n^T \int q(\mathbf{W}_{ni})\mathbf{W}(a)_{ni}^T\mathbf{W}(b)_{ni}\, d\mathbf{W}_{ni}\, \mathbf{f}(b)_n\, d\mathbf{f}_n \\
&= \int q(\mathbf{f}_n)\mathbf{f}(a)_n^T \int q(\mathbf{W}_{ni}^c) \int q(\mathbf{W}_{ni}^b) \int q(\mathbf{W}_{ni}^a)\mathbf{W}(a)_{ni}^T\mathbf{W}(b)_{ni}\, d\mathbf{W}_{ni}^a\, d\mathbf{W}_{ni}^b\, d\mathbf{W}_{ni}^c\, \mathbf{f}(b)_n\, d\mathbf{f}_n \\
&= \int q(\mathbf{f}_n)\mathbf{f}(a)_n^T \mu_{\mathbf{W}_{ni}^a}^T \mu_{\mathbf{W}_{ni}^b} \mathbf{f}(b)\, d\mathbf{f}_n \\
&= \mu_{\mathbf{f}_n^a}^T \mu_{\mathbf{W}_{ni}^a}^T \mu_{\mathbf{W}_{ni}^b} \mu_{\mathbf{f}_n^b}
\end{aligned} \tag{20}$$

Because we are summing over a, b we can write

$$\begin{aligned}
\mathbb{E}_q\big[(\sum_r^{R_a} \mathbf{W}(r)_{ni}\mathbf{f}(r)_n)^T(\sum_r^{R_a} \mathbf{W}(r)_{ni}\mathbf{f}(r)_n)\big] &= (\frac{1}{R_a^2}\sum_b^{R_a}\sum^{R_a} \mu_{\mathbf{f}_n^a}^T \mu_{\mathbf{W}_{ni}^a}^T \mu_{\mathbf{W}_{ni}^b} \mu_{\mathbf{f}_n^b}) \\
&+ (\frac{1}{R_a^2}\sum_a^{R_a}\sum_{b=a}^{R_a} Tr[\Sigma_{\mathbf{W}_a}\Sigma_{\mathbf{f}_n^a}] + \mu_{\mathbf{f}_n^a}^T\Sigma_{\mathbf{W}_{ni}^a}\mu_{\mathbf{f}_n^a} + \mu_{\mathbf{W}_{ni}^a}^T\Sigma_{\mathbf{f}_n^a}\mu_{\mathbf{W}_{ni}^a}
\end{aligned} \tag{21}$$

C.2 MR-GPRN: Closed Form Expected Log Likelihood ($\mathbf{W} \to \exp(\mathbf{W})$)

If $\mathbf{W}$ is passed through an exponential function to enforce positive latent weights the expected log likelihood is:

$$\mathcal{L}_{ell} = \sum_{a=1}^{\mathcal{A}} \alpha E_q \left[\log \mathcal{N}(\mathbf{Y}_a | \frac{1}{R_a}\sum_{r=1}^{R_a}(\exp(\mathbf{W})\mathbf{f})(r), \sigma_a^2)\right] \tag{22}$$

Each of the likelihood components can now be dealt with separately. Consider the general a component we have:

$$\begin{aligned}
\mathcal{L}_{ell_a} = \; & C_1 + C_2 \sum_{n=1}^{N_a}\sum_{k=1}^{K}\pi_k\sum_{p=1}^{P}\mathbb{E}_q\left[(\mathbf{Y}_{ni}^{(a)})^T\mathbf{Y}_{ni}^{(a)}\right] - \mathbb{E}_q\left[(\frac{1}{R_a}\sum_r^{R_a}\exp(\mathbf{W}(r)_{ni})\mathbf{f}(r)_n)^T\mathbf{Y}_{ni}^{(a)}\right] - \\
& \mathbb{E}_q\left[(\mathbf{Y}_{ni}^{(a)})^T(\frac{1}{R_a}\sum_r^{R_a}\exp(\mathbf{W}(r)_{ni})\mathbf{f}(r)_n)\right] + \\
& \mathbb{E}_q\left[(\sum_r^{R_a}\exp(\mathbf{W}(r)_{ni})\mathbf{f}(r)_n)^T(\sum_r^{R_a}\exp(\mathbf{W}(r)_{ni})\mathbf{f}(r)_n)\right]
\end{aligned}$$

We now deal with each of the expectations separately.

C.2.1 ELL: 1st Term

As in Sec. C.1.1 the expectation is constant:

$$\mathbb{E}_q\left[(\mathbf{Y}_{ni}^{(a)})^T\mathbf{Y}_{ni}^{(a)}\right] = {\mathbf{Y}_{ni}^{(a)}}^T\mathbf{Y}_{ni}^{(a)} \tag{23}$$

C.2.2 ELL: 2nd and 3rd Term

As in Sec. C.1.2 the expectation is applied to $\mathbf{f}$ and $\mathbf{W}$ separately. To evaluate the expectation of $\exp \mathbf{W}$ we use the result from Eq. 9:

$$\mathbb{E}_q\big[(\frac{1}{R_a}\sum_r^{R_a}\exp(\mathbf{W}(r)_{ni})\mathbf{f}(r)_n)^T\mathbf{Y}_{ni}^{(a)}\big] = (\frac{1}{R_a}\sum_r^{R_a}\mathbb{E}_{q(\mathbf{W}_{ni})}\left[\exp(\mathbf{W}(r)_{ni})\right]\mathbb{E}_{q(\mathbf{f}_n)}\left[\mathbf{f}(r)_n)^T\right]\mathbf{Y}_{ni}^{(a)}$$
$$= (\frac{1}{R_a}\sum_r^{R_a}\exp(\mathbb{E}_{q(\mathbf{W}_{ni})}\left[\mathbf{W}(r)_{ni}\right] + \mathbb{V}_{q(\mathbf{W}_{ni})}\left[\mathbf{W}_{ni}(r)\right])\cdot\mathbb{E}_{q(\mathbf{f}_n)}\left[\mathbf{f}(r)_n)^T\right]\mathbf{Y}_{ni}^{(a)} \tag{24}$$

C.2.3 ELL: 4th Term

In the last expectation we have a product of sums that is expanded into a double sum over a and b. There is now two cases: when $a = b$ inside the sum $\mathbf{f}$ and $\mathbf{W}$ will appear in square forms, when $a \neq b$ the expectation can be treated as in Sec. C.2.2.

$$\mathbb{E}_q\big[(\sum_r^{R_a}\exp(\mathbf{W}(r)_{ni})\mathbf{f}(r)_n)^T(\sum_r^{R_a}\exp(\mathbf{W}(r)_{ni})\mathbf{f}(r)_n)\big]$$
$$= \frac{1}{{R_a}^2}\sum_a^{R_a}\sum_b^{R_a}\mathbb{E}_{q(\mathbf{f}_n)q(\mathbf{W}_{ni})}\left[\mathbf{f}(a)_n^T\exp(\mathbf{W}(a)_{ni})^T\exp(\mathbf{W}(b)_{ni})\mathbf{f}(b)_n\right] \tag{25}$$

case $a = b$:

$$\int q(\mathbf{f}_n^a)q(\mathbf{W}_{ni}^a)\mathbf{f}(a)^T\exp(\mathbf{W}(a))^T\exp(\mathbf{W}(a)_{ni})\mathbf{f}(a)\,d\mathbf{f}_a\,d\mathbf{W}_{ni}^a$$
$$= \int q(\mathbf{f}_n^a)\mathbf{f}(a)^T\exp(2\cdot(\mathbb{E}_{q(\mathbf{W}_{ni}^a)}\left[\mathbf{W}_{ni}^a\right] + \mathbb{V}q(\mathbf{W}_{ni}^a)\left[\mathbf{W}_{ni}^a\right]))\mathbf{f}(a)\,d\mathbf{f}_a$$
$$= \mu_{\mathbf{f}_n^a}^T\exp(2\cdot(\mathbb{E}_{q(\mathbf{W}_{ni}^a)}\left[\mathbf{W}_{ni}^a\right] + \mathbb{V}_{q(\mathbf{W}_{ni}^a)}\left[\mathbf{W}_{ni}^a\right]))\mu_{\mathbf{f}_n^a}$$
$$+ Tr\left[\exp(2\cdot(\mathbb{E}_{q(\mathbf{W}_{ni}^a)}\left[\mathbf{W}_{ni}^a\right] + \mathbb{V}q(\mathbf{W}_{ni}^a)\left[\mathbf{W}_{ni}^a\right]))\mathbf{\Sigma}_{\mathbf{f}_n^a}\right]$$

To evaluate the expected value of the square form of $\exp(\mathbf{W})$ we apply the result of Eq. 10.

In the case of $a \neq b$:

$$\int q(\mathbf{f}_n)\mathbf{f}(a)_n^T\int q(\mathbf{W}_{ni}^c)\int q(\mathbf{W}_{ni}^b)\int q(\mathbf{W}_{ni}^a)\exp(\mathbf{W}(a)_{ni})^T\exp(\mathbf{W}(b)_{ni})\,d\mathbf{W}_{ni}^a\,d\mathbf{W}_{ni}^b\,d\mathbf{W}_{ni}^c\,\mathbf{f}(b)_n\,d\mathbf{f}_n$$
$$= \int q(\mathbf{f}_n)\mathbf{f}(a)_n^T\exp(\mu_{\mathbf{W}_{ni}^a} + \frac{1}{2}\Sigma_{ni}^a)^T\exp(\mu_{\mathbf{W}_{ni}^b} + \frac{1}{2}\Sigma_{ni}^b)\mathbf{f}(b)\,d\mathbf{f}_n$$
$$= \mu_{\mathbf{f}_n^a}^T\exp(\mu_{\mathbf{W}_{ni}^a} + \Sigma_{ni}^a)^T\exp(\mu_{\mathbf{W}_{ni}^b} + \Sigma_{ni}^b)\mu_{\mathbf{f}_n^b}$$

D Synthetic Examples

Apart from the variational experiments in Section 2 of the main paper, additional experiments using a Markov Chain Monte Carlo (MCMC) approach are conducted in this section. We show that when the dependency structure is lost through the product likelihood construction, the mean of the posterior distribution for the latent function will also deviate from the true one. We also demonstrate the posterior contraction and the effect of the different corrections.

D.1 Data Generating Process

We generate two synthetic observation processes from:

$$
\begin{aligned}
y_i^{(1)} &= f(x_i) + \epsilon_1 \\
y_j^{(2)} &= \frac{1}{2} \sum_{k=2j-1}^{2j} y_k^{(1)} + \epsilon_2
\end{aligned}
\tag{26}
$$

Where $y_i^{(1)}, i = 1, ..., 2N$ is the observed value of $f(x_i)$ with noise $\epsilon_1 \sim \mathcal{N}(0, \sigma_1^2)$ and $y_j^{(2)}, j = 1, ..., N$ is the aggregate function of $y^{(1)}$ with noise $\epsilon_2 \sim \mathcal{N}(0, \sigma_2^2)$,$\sigma_1 = 1, \sigma_2 = 0.1$. We are using a $\sin$ function to generate data $f(x_i) = 5\sin^2(x_i)$. The likelihood function with the observation processes $\mathbf{Y_1} = \{y_i^{(1)}\}_{i=1}^{2N}$, $\mathbf{Y_2} = \{y_j^{(2)}\}_{j=1}^{N}$ is given by:

$$
L(\mathbf{Y}_1, \mathbf{Y}_2) = p(\mathbf{Y}_1|\mathbf{f}(\mathbf{x}), \sigma_1^2) p(\mathbf{Y}_2|\mathbf{Y}_1, \sigma_2^2) \tag{27}
$$

When the data from $\mathbf{Y}_1$ has the same support as the observation process $\mathbf{Y}_2$, the evidence from $\mathbf{Y}_2$ will not affect parameter estimation in the probability function $p(\mathbf{Y}_1|\mathbf{f}(\mathbf{x}), \sigma_1)$. However, when $\mathbf{Y}_2$ has different support from the observed $\mathbf{Y}_1$, the additional evidence should impact parameter inference. As $\mathbf{Y}_2$ does not depend on the latent function, this evidence will be hard to pass via the likelihood function in Eq. 27. One way to correct for this is to introduce dependency between $\mathbf{Y}_1$ and $\mathbf{Y}_2$ through a non-parametric prior over the latent function $\mathbf{f}(\mathbf{x})$.

D.2 Gaussian Processes: Product Likelihood

Since the two observation processes follow the same underlying function $sin^2(x)$, we use a single Gaussian process to model the latent function $f(x)$. We assume:

$$
f(x) \sim \mathcal{GP}(0, k(x, x')) \tag{28}
$$

where $k(x, x')$ is the covariance function of $f(x)$. We are using the squared exponential kernel:

$$
k(x, x') = A \exp(-\frac{(x - x')^2}{l}) \tag{29}
$$

where A is the amplitude parameter and l is the length scale for the kernel function. Thus, we can write down the joint distribution of $\mathbf{Y}_1$ and $\mathbf{Y}_2$ as:

$$
p(\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{f}(\mathbf{x})) = p(\mathbf{f}(\mathbf{x})|\theta) p(\mathbf{Y}_1|\mathbf{f}(\mathbf{x}), \sigma_1^2) p(\mathbf{Y}_2|\mathbf{Y}_1, \sigma_2^2) \tag{30}
$$

Where θ is the hyper-parameters for the Gaussian process. We can write down the distribution of $\mathbf{Y}_1$ and $\mathbf{Y}_2$ by marginalizing out the latent function:

$$
p(\mathbf{Y}_1, \mathbf{Y}_2) = \int p(\mathbf{f}(\mathbf{x})|\theta) p(\mathbf{Y}_1|\mathbf{f}(\mathbf{x}), \sigma_1^2) p(\mathbf{Y}_2|\mathbf{Y}_1, \sigma_2^2) d\mathbf{f}(\mathbf{x}) \tag{31}
$$

As $\mathbf{Y}_1$ has all the information of $\mathbf{f}(\mathbf{x})$, this integral is tractable and we can write $y^{(1)} \sim \mathcal{GP}(0, k(x, x') + \sigma_1)$. But when the aggregation function $\mathbf{Y}_2$ has additional information about the latent function, i.e. $\mathbf{Y}_1$ and $\mathbf{Y}_2$ only partially overlapping, bringing additional information from $\mathbf{Y}_2$ requires the prediction of the missing values of the corresponding $\mathbf{Y}_1$ process. This can be done in Markov Chain Monte Carlo(MCMC) setting by treating the unobserved value of $\mathbf{Y}_1$ as extra parameters. However, this increase a lot of computational complexity for the MCMC sampler. One option is to make an independence assumption for $\mathbf{Y}_1$ and $\mathbf{Y}_2$. Thus, the information in $\mathbf{Y}_2$ can affect the latent function $\mathbf{f}$ directly.

D.3 Gaussian Processes: Composite Likelihood

Using the composite likelihoods, we assume each part of the likelihood is independent to each other. For the joint probability of $\mathbf{Y}_1$ and $\mathbf{Y}_2$, we have:

$$p(\mathbf{Y}_1, \mathbf{Y}_2) = \int p(\mathbf{f}(\mathbf{x})|\theta)p(\mathbf{Y}_1|\mathbf{f}(\mathbf{x}), \sigma_1^2)p(\mathbf{Y}_2|\mathbf{f}(\mathbf{x}), \sigma_2^2)d\mathbf{f}(\mathbf{x}) \tag{32}$$

Instead of assuming the conditional probability $p(\mathbf{Y}_2|\mathbf{Y}_1, \sigma_2^2)$, we are now assuming the data depends on the latent function $\mathbf{f}(\mathbf{x})$ directly. However, when $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are different resolutions under the same support, this likelihood misspecifies the correlation and will make the inference of $f(x)$ contract into the observed mean. While this contraction actually equals to an extra bias to the data in the overlapping zone, the misspecified dependency structure will lead to an overfitting problem. This overfitting problem of product likelihoods has been studied in the information theory [28, 31] and the simplest way is to use an exponential weight to correct the inference:

$$L(\mathbf{Y}_1, \mathbf{Y}_2) = \int p(\mathbf{f}(\mathbf{x})|\theta)p(\mathbf{Y}_1|\mathbf{f}(\mathbf{x}), \sigma_1^2)^{\alpha}p(\mathbf{Y}_2|\mathbf{f}(\mathbf{x}), \sigma_2^2)^{\alpha}d\mathbf{f}(\mathbf{x}) \tag{33}$$

where $\alpha \in \mathbb{R}_{>0}$ is composite weight for the likelihood. The problem of learning the latent function becomes learning the parameters of the likelihood function and the composite weights.

D.4 Composite Weights

The composite log likelihood function can be written as:

$$\ell_c(\hat{\theta}) = \sum_{i=1}^{k} f(\hat{\theta}_i|\mathbf{Y}) \tag{34}$$

where $f(\hat{\theta}_i|Y)$ is the likelihood function of i-th parameter θ_i and we assume each part of the likelihood function is independent to each other. $\hat{\theta}_i$ is the estimated value of θ_i. With the observed distribution of $\mathbf{Y}$, $p_0(\mathbf{Y}|\theta_0)$ and θ_0 as the true parameter value, we have:

$$\ell_c'(\theta_0) = \ell_c'(\hat{\theta}) + (\theta_0 - \hat{\theta})\ell_c''(\hat{\theta}) + o(n^{-1}) \tag{35}$$

$$\hat{\theta} - \theta_0 \to -\frac{\ell'(\theta_0)}{\ell''(\theta_0)} \tag{36}$$

Since we have $\ell'(\theta) = J(\theta)$ and $\ell''(\theta) = H(\theta)$, the variance of θ will follow the sandwich variance $H^{-1}(\theta)J(\theta)H^{-1}(\theta)$. Then, calculating the Taylor expansion for the likelihood, we have:

$$\ell_c(\theta_0) = \ell_c(\hat{\theta}) + (\theta_0 - \hat{\theta})\ell_c'(\hat{\theta}) + \frac{1}{2}(\theta_0 - \hat{\theta})\ell_c''(\hat{\theta})(\theta_0 - \hat{\theta})^T + o(n^{-1}) \tag{37}$$

The expected variance from the composite likelihood model is :

$$\mathbb{E}_\theta[Var(\hat{\theta}|\mathbf{Y})] = -H(\hat{\theta}|\mathbf{Y}) \tag{38}$$

Since $\hat{\theta} \to \theta$, we need to set the variance of the estimated parameter to the asymptotic variance. Thus, we have:

$$\mathbb{E}_\theta[\alpha Var(\hat{\theta}|\mathbf{Y})] = H^{-1}(\hat{\theta}|\mathbf{Y})J(\hat{\theta}|\mathbf{Y})H^{-1}(\hat{\theta}|\mathbf{Y}) \tag{39}$$

For a scalar variable, we can match the variance to the exact asymptotic vairance using a scalar number. But if the estimating variable θ is high dimensional, it's not easy to adjust the proper variance using a single weight. We could use a matrix ($\mathbf{C} \in \mathbb{R}^{k\times k}$) to adjust the covariance structure. In this case we would have:

$$\mathbf{C}H(\hat{\theta}|\mathbf{Y})\mathbf{C}^T = H^{-1}(\hat{\theta}|\mathbf{Y})J(\hat{\theta}|\mathbf{Y})H^{-1}(\hat{\theta}|\mathbf{Y}) \tag{40}$$

However, this increases the computational complexity substantially. One alternative way is to use a scalar weight to match the identities of the covariance matrix. Lyddon et al. [15] and Ribatet [24] developed two different ways to adjust the identities of the covariance matrix:

$$\alpha_{\text{Ribatet}} = \frac{|\hat{\theta}|}{Tr[\mathbf{H}(\hat{\theta})^{-1}\mathbf{J}(\hat{\theta})]} \quad , \quad \alpha_{\text{Lyddon}} = \frac{Tr[\mathbf{H}(\hat{\theta})\mathbf{J}(\hat{\theta})^{-1}\mathbf{H}(\hat{\theta})]}{Tr[\mathbf{H}(\hat{\theta})]}. \tag{41}$$

Where α_{Ribatet} considers all the information in the covariance matrix and α_{Lyddon} only matches the information in the diagonal elements.

D.5 MCMC Composite Likelihood Experiments

We now construct an MCMC experiment for the synthetic data using Eq. 26. Instead of sampling directly from the intractable joint distribution of $L(\mathbf{Y_1}, \mathbf{Y_2})$, we sample from the joint probability with the latent variable $L(\mathbf{Y_1}, \mathbf{Y_2}, \mathbf{f}(\mathbf{x}))$ via a Metropolis-Hastings within Gibbs sampler. We perform three block updates: on θ_0 for the Gaussian process prior, $\mathbf{f}(\mathbf{x})$ for the latent function variables and $\sigma^2 = \{\sigma_1^2, \sigma_2^2\}$ for the noise parameter.

Algorithm 3 Block Metropolis-Hastings within Gibbs

Input: Observed datasets $\{(\mathbf{X}_s, \mathbf{Y}_s)\}_{s=1}^{S}$, initial parameters θ_0,

for i-th iteration **do**
Update parameter block θ_i

function BLOCK (θ_i)

1. Sample proposed value of the Gaussian process prior $\theta_i' \sim N(\theta_{i-1}, \Delta)$
2. Calculate the conditional probability distribution $p(\theta_i'|Y_1, Y_2, \sigma_{i-1}^2, f(x)_{i-1})$
3. Calculate the acceptance rejection ratio:

$\pi = \frac{p(\theta_i'|Y_1, Y_2, \sigma_{i-1}^2, \mathbf{f}(\mathbf{x})_{i-1})}{p(\theta_{i-1}|Y_1, Y_2, \sigma_{i-1}^2, \mathbf{f}(\mathbf{x})_{i-1})}$

4. Update the i-th value of θ_i via π

end function

Update the parameter block $\mathbf{f}(\mathbf{x})_i$ via $\pi = \frac{p(\mathbf{f}(\mathbf{x})_i'|Y_1, Y_2, \sigma_{i-1}^2, \theta_i)}{p(\mathbf{f}(\mathbf{x})_{i-1}|Y_1, Y_2, \sigma_{i-1}^2, \theta_i)}$

Update the parameter block σ_i via $\pi = \frac{p(\sigma_i^2|Y_1, Y_2, \theta_i, \mathbf{f}(\mathbf{x})_i)}{p(\sigma_{i-1}^2|Y_1, Y_2, \theta_i, \mathbf{f}(\mathbf{x})_i)}$

end for
return $\theta, \mathbf{f}(\mathbf{x}), \sigma^2$

D.6 Variational Composite Likelihood Experiments

In this section we provide further details to reproduce the variational composite likelihood experiments in Sec. 2.2 and Fig. 2 of the main paper.

Data Generation We consider the case of having two dependent observation processes. We generate one process $\mathbf{Y}_1 = 5 \cdot \sin(\mathbf{X})^2 + 0.1 \cdot \epsilon$ with $\epsilon \sim \mathcal{N}(0, 1)$ with 100 samples over the range $[-2, 15]$. For $\mathbf{Y}_2$ we aggregate $\mathbf{Y}_1$ into bins of size 3, $\mathcal{S}_2 = 3$, so that $\mathbf{Y}_2 \in \mathbb{R}^{33}$ and $\mathbf{X}_2 \in \mathbb{R}^{33\times 3}$. In Fig. 2 we only plot the range $[3, 10]$.

Parameter Initialization For both MR-GP and VBAGG-NORMAL we use an SE kernels with length-scale of 0.1 and variance 1.0. We initialize the likelihood noise to 0.1.

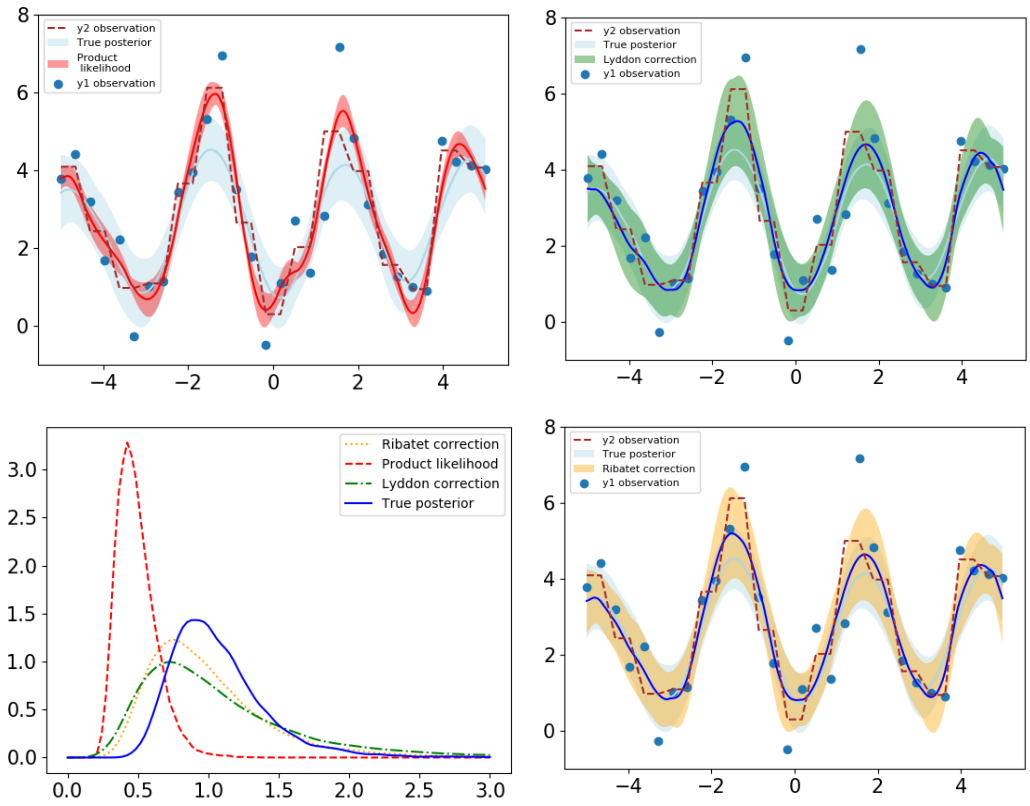


Figure 5: **Top Left:** Comparing the posterior of the latent function $f(x)$ under the product likelihood assumption and a correctly specified likelihood. The product likelihood assumption causes extreme posterior contraction which effects both the mean and variance. **Bottom Left:** Comparison of the true posterior of σ_1 noise to the posterior under the product likelihood and under the composite likelihood with different weights. The composite likelihood is able to recover the true posterior. **Top Right:** Comparing the posterior of the latent function $f(x)$ under the composite likelihood assumption with Lyddon correction and a correctly specified likelihood.**Bottom Right:** Comparing the posterior of the latent function $f(x)$ under the composite likelihood assumption with Ribatet correction and a correctly specified likelihood. The two correction have the similar results for our experiments. Although the mean of the function is not exactly match the true function, the variance of the latent function is corrected close to the true function. The misspecified part is due to the imprefect match of the asymptotic variance discussed in section D.4

Additional Training Details For VBAGG-NORMAL we run for 10000 epochs. For MR-GP for we run the MLE estimate for 10000 epochs, obtain α_{Ribatet} and then optimize the ELBO for 10000 epochs.

E Multi-resolution Air Pollution Experiments

E.1 Inter-task Multi-resolution: PM10-PM25

In this section we provide additional details for reproducing the inter-task multi-resolution experiments as described in the main paper.

Variational Parameter Initialization: For MR-DGP we initialize all likelihood noises to 0.01 and we use a Matern32 kernel for all latent functions with a lengthscale of 0.01. For both MR-GPRN and CENTER-POINT we initialize the likelihood noise to be 0.1 and use a squared exponential kernel for all latent functions. We use $Q = 1$ and set the lengthscale of $\mathbf{f}$ to be 0.1 and the lengthscales of $\mathbf{W}$ to be 3.0.

Training Details: We train MR-DGP for a total of 900 iterations. We train both MR-GPRN and CENTER-POINT for 2000 iterations each.

E.2 Intra-task Multi-resolution: Space-time NO_2

In this section we provide additional details for reproducing the *intra*-task multi-resolution experiments described in Sec. 4 of the main paper.

Data pre-processing: We extract spatial features based on the London road network (OS Highways) [2] and land use (UKMap) [3]. OS Highways is a dataset of every road in London with information of the length, road classification (A Road, B Road, etc). UKMap is a dataset of polygons where each polygon represents a physical entity, e.g a building, a river, a park, etc. UKMap provides additional information such as the height of the buildings and the area of the parks and rivers. For each input location we construct a buffer of approximately 100m (a radius 0.001 degrees in SRID:4326). Within the buffer zone we calculate the average length of the A-roads, the average ratio between the width of the roads and height of buildings on the corresponding roads, and the total area of vegetation and water. We convert all time stamps into unix epochs and we standardize all features before training. To approximate the integral in the likelihood (Eq. 4 in main text) we discretize the area of each satellite based observation input into a 10 by 10 uniform grid of lat-lon points.

MR-DGP Architecture: For MR-DGP we use the architecture described on the right subfigure of Fig. 3 in the main paper where $\mathbf{X}_2, \mathbf{Y}_2$ corresponds to the LAQN dataset and $\mathbf{X}_1, \mathbf{Y}_1$ to the satellite dataset. We give the initialization of the specific latent functions below.

Variational Parameter Initialization: For MR-GPRN and VBAGG-NORMAL we use 400 inducing points for all latent functions. Both the inducing function values and the variances are randomly initialized between 0 and 1. For MR-DGP the latent functions $\mathbf{f}_1^{(1)}$ and $\mathbf{f}_2^{(1)}$ we place 300 inducing points and for $\mathbf{f}_2^{(2)}$ we use 100. For all models we initialize the inducing points locations with K-means with K=300 on the satellite model observations input for $\mathbf{f}_1^{(1)}$ and $\mathbf{f}_2^{(1)}$ and K=100 for $\mathbf{f}_2^{(2)}$.

Model Parameter Initialization: In all models and latent function withing, MR-GPRN, VBAGG-NORMAL and MR-DGP we use SE kernels initialized with lengthscales of 0.1 and SE variance to 1.0. We initialize the likelihood noise to be 0.1.

Additional Training Details: We train MR-DGP for a total of 1200 iterations. We train both MR-GPRN, VBAGG-NORMAL and CENTER-POINT for 2000 iterations each.

F Relation to VBAgg

In this section we show that MR-GPRN is a generalisation of VBAGG-NORMAL [13] from a single GP to a GPRN. In VBAgg each observation y^a is the aggregate output of some bag of items $\mathbf{x}^a = \{\mathbf{x}_i^a\}_{i=1}^{N_a}$. The likelihood of each bag is $y^a|\mathbf{x}^a \sim \mathcal{N}(y|\eta^a, \tau^a)$ where $\eta^a = \sum_{i=1}^{N_a} w_i^a \mu(\mathbf{x}_i^a)$ and μ is the mean of the latent process f. In MR-GPRN we are modelling the underlying process with the sum of products of GPs. Rewriting MR-GPRN using the notation of [13]: $\mu = \mathbf{W}\mathbf{f}$ and each dataset $\{\mathbf{X}_a, \mathbf{Y}_a\}_{a=1}^{\mathcal{A}}$ directly corresponds to the observations and bag of items defined in VBAGG-NORMAL. Let $N_a = \mathcal{S}_a$, and $\tau^a = \sigma_a^2$ and the composite weight $\alpha = 1$. The composite weight of value 1 is implicitly included in the model of VBAGG-NORMAL through the independence assumption. We assume an simple aggregation of the bag of items, although we note that is not necessary, so setting $w_i^a = \frac{1}{\mathcal{S}_a}$ we obtain $y^a \sim \mathcal{N}(\sum_{i=1}^{N_a} w_i^a \mu(\mathbf{x}_i^a), \tau^a)$ which is MR-GPRN in the notation of [13]. VBAGG-NORMAL is then recovered when we use only one latent function (by setting $\mathbf{W}$ to a constant value), by only considering the single task setting and by setting the composite weight to one.

[2] `https://www.ordnancesurvey.co.uk/business-and-government/help-and-support/products/os-mastermap-highways-network.html`

[3] `https://www.geoinformationgroup.co.uk/ukmap`