Manuscript version: Author’s Accepted Manuscript
The version presented in WRAP is the author’s accepted manuscript and may differ from the published version or Version of Record.

Persistent WRAP URL:
http://wrap.warwick.ac.uk/125505

How to cite:
Please refer to published version for the most recent bibliographic citation information. If a published version is known of, the repository item page linked to above, will contain details on accessing it.

Copyright and reuse:
The Warwick Research Archive Portal (WRAP) makes this work by researchers of the University of Warwick available open access under the following conditions.

Copyright © and all moral rights to the version of the paper presented here belong to the individual author(s) and/or other copyright owners. To the extent reasonable and practicable the material made available in WRAP has been checked for eligibility before being made available.

Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

Publisher’s statement:
Please refer to the repository item page, publisher’s statement section, for further information.

For more information, please contact the WRAP Team at: wrap@warwick.ac.uk.
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 initiatives 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
dependency between observation processes through composite likelihoods and extends the Gaussian aggregation model of Law et al. [15], 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 [23] 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 2 and 3 introduce our shallow GP mixtures and deep GP constructions respectively. In Section 6 we demonstrate the empirical advantages of our framework versus 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 $Y_a$ with varying resolutions $R_a$ and sampling periods $S_a$. The observation process $Y_{a'}$ with the highest sampling rate per unit measure and hence the smallest sampling period has the base sampling period $S_{a'} = B$. Typically the observation process of interest, denoted $Y_B$, is at the sampling period $B$ but our formulation and framework applies generally. We normalize the sampling periods with respect to the base one such that $S_{a} := 1$ and hence $S_{a} = S_{a'} \in \mathbb{R}_{\geq 1}$. For example if the base period is hourly and we have a process $a$ with $N_a$ daily observations then its sampling period is $S_a = 24$ and $Y_a \in \mathbb{R}^{N_a \times 1}$ and $X_1 \in \mathbb{R}^{N_a \times 24 \times 1}$. We construct $\mathcal{A}$ datasets $\{(X_{a}, Y_{a})\}_{a=1}^{\mathcal{A}}$ across the set of all tasks $P = \{p\}$ with $X_a \in \mathbb{R}^{N_a \times S_a \times D_a}$ and $Y_a \in \mathbb{R}^{N_a \times |P_a|}$ 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 $f_q$ that are mixed through further $|P|Q \in \mathbb{N}$ latent GP functions $W_{p,q}$. More formally, $f_q \sim \mathcal{GP}(0, K_q)$ and $W_{p,q} \sim \mathcal{GP}(0, K_{p,q})$. For task $p$ the observations $Y_p = W_p f + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$ such that $Y_p$ 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 $p(W, f) = \prod_{p=1}^{\mathcal{A}} \prod_{q=1}^{|P|Q} \mathcal{N}(W_{p,q}|0, \mathbf{K}_{p,q}) \prod_{q=1}^{Q} \mathcal{N}(f_q|0, \mathbf{K}_q)$. 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 [15] [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:
V = \phi \{ U \}

\text{For computational efficiency we introduce inducing points to bring these distributions as close together as possible. By setting } N \text{ following Lyddon et al. [15] and Ribatet [24] we write down the asymptotic posterior of } H(\theta) \text{ and } J(\theta) \text{ 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 Bartlett’s second identity [30].}

\text{Following Lyddon et al. [15] and Ribatet [24] we write down the asymptotic posterior of MR-GPRN as } H(\theta, n^{-1}\phi^{-1}H(\theta)) \text{. Asymptotically one would expect the contribution of the prior to vanish causing the asymptotic posterior to match the limiting MLE. The composite weights } \phi \text{ can be used to bring these distributions as close together as possible. By setting } \phi^{-1}H(\hat{\theta}) = H(\theta_0)J(\theta_0)^{-1}H(\theta_0) \text{, and rearranging to find } \phi \text{ 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{\mid \hat{\theta} \mid}{\text{Tr}[H(\hat{\theta})^{-1}J(\hat{\theta})]} \quad \phi_{\text{Lyddon}} = \frac{\text{Tr}[H(\hat{\theta})J(\hat{\theta})^{-1}H(\hat{\theta})]}{\text{Tr}[H(\hat{\theta})]}.

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

$$
\mathcal{L}_a = \sum_{n}^{N} \sum_{i}^{|p_i|} Y_{a,i,n}^T Y_{a,i,n} + \sum_{a}^{S_a} \sum_{n}^{S_n} \text{Tr}[\Sigma_{W_n}^a \Sigma_{f_n^a}] + \mu_{f_n^a}^T \Sigma_{W_n}^a \mu_{f_n^a} + \mu_{W_n}^T \Sigma_{f_n^a} \Sigma_{W_n}^a \\
+ \frac{2}{S_a} \sum_{a}^{S_a} \mu_{W_n}^T \mu_{f_n^a} + \frac{1}{S_a} \sum_{b}^{S_b} \mu_{W_n}^T \mu_{w_n^b} \mu_{f_n^b}. 
$$

where $\mu_{W_n}^a, \Sigma_{W_n}^a$ and $\mu_{f_n^a}, \Sigma_{f_n^a}$ are respectively the mean and variance of $q(W)$ and $q(f)$ at input $x_{a,n}$. To infer the composite weights we first obtain the MLE estimate of $\theta$ by maximizing the likelihood in Eq. [1]. The weights can then be calculated and the variational lowerbound optimised as in Alg. [1] with $O(E \cdot (|P| Q + Q) NM^2)$ 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 $\mathbb{E}[Y_{a,p}] = \sum_k^{K_\pi} \pi_k \mathbb{E}_k \left[ \left[ W_p \right]^a \mathbb{E}_k \left[ \hat{f} \right] \right]$ and the variance is: $\mathbb{V}[Y_{a,p}] = \sum_k^{K_\pi} \pi_k (\sigma_f^2 I + \text{Tr} (\mathbb{V}_k \hat{f} \mathbb{V}_k [W_p] \mathbb{E}_k \left[ \hat{f} \right])[W_p]) + \mathbb{E}_k \left[ W_p \right] \mathbb{V}_k \left[ \hat{f} \right] \mathbb{E}_k \left[ W_p \right] + \text{Tr} \left( \mathbb{E}_k \left[ \hat{f} \right] \mathbb{E}_k \left[ \hat{f} \right]^T \mathbb{V}_k \left[ W_p \right] \right) + \mathbb{E}_k \left[ W_p \right] \mathbb{E}_k \left[ \hat{f} \right]^T \mathbb{E}_k \left[ W_p \right]^T \mathbb{E}_k \left[ \hat{f} \right] - \mathbb{E}[Y_{a,p}] \mathbb{E}[Y_{a,p}]^T$. Where $K$ is the number of components in the mixture of Gaussians variational posterior and $\pi_k$ is the $k$'th weight. Full derivations are in the Supp. together with corresponding ones for the positively-restricted GPRN form $Y = \sum_{p=1}^{|P|} \exp(W_p^a)f + \epsilon$ 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. 5. 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 $\{(X_a, Y_a)\}_{a=1}^{A}$ and the latent functions $\{f^{(k)}(x)\}_{k=1}^{K_a}$ that are linearly mixed to model each observation process $a$ following a mixture of experts approach [19, 22, 34]: $m_k(x) = \sum_{a=1}^{A} \pi_k w_k^{(a)} f^{(k)}(x)$. Every $f^{(k)}$ is a deep DGP of length $L^{(k)}$ and is targeting the same response $Y_a$. 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
The integral is again approximated by discretizing the sampling region with a uniform grid. Our variances of the latent functions; taking $V(x)^k$ to be the normalized predictive variance of the $k$’th function then $u^k_a = (1 - V^k_a) \sum_{i=1}^{N} y^i_a$. Formally $f_a^k \sim \mathcal{GP}(0, K_a^k(\{P(x_a), P(1^{(1)}(X_a)\})$ and $P_a^{(1)}$ denotes the parent function of $f_a^k$. When $k = 0$ we define $P_{a,0}^{(1)}$ to be the identity function, and when $k > 1$ then $P_{a,k}^{(1)}$ denotes the prediction from a specific $m_a$. For example in Fig. 3 the parent function of $f_2^{(2)}$, denoted by $P_{2,2}^{(1)}$, is $m_1$. Following Section 3 we link each of the functions to their target resolution through the likelihood. The posterior, $p(\{f_a^k\}_{a=1}^A | Y, X)$, is now proportional to:

$$p(\{f_a^k\}_{a=1}^A | Y, X) \propto \prod_{a=1}^A \prod_{k=1}^{K_a} \prod_{n=1}^{N_a} \mathcal{N}(Y_{a,n} | \int_{S_a} f_a^k(X_{a,n}, s) dX_{a,n,s}, \sigma_{a,k}^2 \cdot I) \cdot \prod_{a=1}^A \prod_{k=1}^{K_a} p(f_a^k | P_{a,k}^{(1)} X_a^k)$$ (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 $m_a$ is modelling a specific observation process.

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

$$p(\{Y_a, f_a, u_a\}_{a=1}^A) = \prod_{a=1}^A \prod_{k=1}^{K_a} p(Y_a | X_a, f_a^{(k)}) p(f_a^{(k)} | X_a, P_{a,k}^{(1)} (X_a), u_a^{(k)}) p(u_a^{(k)})$$ (5)

where $p(u_a^{(k)}) = \mathcal{N}(u_a^{(k)} | 0, K_a)$ and $p(f_a^{(k)} | X_a, P_{a,k}^{(1)} (X_a), u_a^{(k)})$ is a Gaussian with mean $f_a | K_a P_{a,k}^{(1)} (X_a), P_{a,k}^{(1)} (X_a), u_a^{(k)}$ and variance $K_a (P_{a,k}^{(1)} (X_a), P_{a,k}^{(1)} (X_a)) - K_a (P_{a,k}^{(1)} (X_a), Z_a^{(k)}, K_a^{-1} (Z_a^{(k)}, Z_a^{(k)})) K_a (Z_a^{(k)}, P_{a,k}^{(1)} (X_a))$ which is the standard noise-free GP predictive distribution.

**Inference.** Following \cite{[25]} we construct an approximate posterior $q(\{f_a, u_a\}_{a=1}^A)$ that maintains the dependency structure between the latent functions:

$$q(\{f_a, u_a\}_{a=1}^A) = \prod_{a=1}^A \prod_{k=1}^{K_a} p(f_a^{(k)} | X_a^k, P(f_a^{(k)}), u_a^{(k)}) q(u_a^{(k)})$$ (6)
Algorithm 2 Inference procedure for MR-DGP

\textbf{Input}: \(S\) multi-resolution datasets \(\{ (\mathbf{X}_a, \mathbf{Y}_a) \}_{a=1}^{S}\), initial parameters \(\theta_0\).

\textbf{procedure} \text{MARGINAL}(\mathbf{f}_a^{(k)}, \mathbf{X}, l) \\
\text{if } l = \mathcal{L}_a^{(k)} \text{ then} \\
\text{return } q(\mathbf{f}_a^{(k)}|\mathbf{X}) \\
\text{end if} \\
q(\mathcal{P}(\mathbf{f}_a^{(k)}|\mathbf{X})) \leftarrow \text{MARGINAL}(\mathcal{P}(\mathbf{f}_a^{(k)}), \mathbf{X}, l + 1) \\
\text{return } \frac{1}{S} \sum_{s=1}^{S} p(\mathbf{f}_a^{(k)}(s), \mathbf{X}) \text{ where } \mathbf{f}(s) \sim q(\mathcal{P}(\mathbf{f}_a^{(k)}|\mathbf{X})) \\
\text{end procedure} \\

\theta_1 \leftarrow \arg \min_{\theta} \left[ \sum_{a=1}^{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{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:

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

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:

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

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 reparametrization trick to draw samples from the variational posteriors [10]. The inference procedure is given in Alg. 2 and has an epoch time complexity of \(\mathcal{O}(\sum_a^{A} \sum_k^{\mathcal{K}} N_a M_a^2 \mathcal{L}_a^{(k)})\).

\textbf{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:

\[
q(\mathbf{m}_a^*) = \int q(\mathbf{m}_a^*) \prod_{l=1}^{\mathcal{K}_a} \mathcal{P}(\mathbf{f}_a^{(l)}|\mathcal{P}(\mathbf{f}_a^{(l)})) \, d\mathcal{P}(\mathbf{f}_a^{(1)}) \cdots d\mathcal{P}(\mathbf{f}_a^{(\mathcal{K}_a)}) \approx \frac{1}{S} \sum_{s=1}^{S} q(\mathbf{m}_a^*|\mathbf{f}_a^{(1)}(s), \ldots, \mathbf{f}_a^{(\mathcal{K}_a)(s)})
\]

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

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 [5] 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 \( \text{NO}_2 \) 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 \( B \)) 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 \( y = s \cdot 5 \sin(x)^2 + 0.1 \epsilon \) where \( \epsilon \sim \mathcal{N}(0, 1) \). The first \( X_1, Y_1 \) is at resolution \( S_1 = 1 \) in the range \( x=[7,12] \) with a scale \( s=1 \). The second is at resolution of \( S_2 = 5 \) between \( x=[-10, 10] \) with a scale \( s=0.5 \) and lastly the third is at resolution of \( S_3 = 5 \) \( x=[10, 20] \) with a scale \( s=0.3 \). The aim is to predict \( y \) 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. \( f_2^{(2)} \) 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. VBA-GG-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.

<table>
<thead>
<tr>
<th>Model</th>
<th>2 Hours</th>
<th>5 Hours</th>
<th>10 Hours</th>
<th>24 Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>CENTER-POINT</td>
<td>12.72 ± 1.29</td>
<td>13.56 ± 1.88</td>
<td>14.99 ± 3.63</td>
<td>16.21 ± 2.54</td>
</tr>
<tr>
<td>MR-DGP</td>
<td>13.22 ± 0.89</td>
<td>13.24 ± 1.6</td>
<td>14.41 ± 1.42</td>
<td>14.59 ± 1.18</td>
</tr>
<tr>
<td>MR-GPRN</td>
<td><strong>10.54 ± 1.15</strong></td>
<td><strong>10.21 ± 0.92</strong></td>
<td><strong>10.11 ± 1.13</strong></td>
<td><strong>10.17 ± 0.68</strong></td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>Model</th>
<th>RMSE</th>
<th>MAPE</th>
<th>Model</th>
<th>RMSE</th>
<th>MAPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CENTER-POINT</td>
<td>18.74 ± 12.65</td>
<td>0.65 ± 0.21</td>
<td>MR-CASCADE</td>
<td>1.61</td>
<td>5.87</td>
</tr>
<tr>
<td>VBAGG-NORMAL</td>
<td>15.98 ± 9.5</td>
<td>0.72 ± 0.47</td>
<td>VBAGG-NORMAL</td>
<td>1.8</td>
<td>2.53</td>
</tr>
<tr>
<td>MR-GPRN</td>
<td>12.95 ± 7.78</td>
<td>0.51 ± 0.33</td>
<td>MR-GPRN</td>
<td>1.68</td>
<td>2.96</td>
</tr>
<tr>
<td>MR-DGP</td>
<td><strong>8.7 ± 5.51</strong></td>
<td><strong>0.36 ± 0.17</strong></td>
<td>MR-DGP</td>
<td><strong>0.3</strong></td>
<td><strong>3.83</strong></td>
</tr>
</tbody>
</table>

**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 × 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


A Definitions

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

$$E[\exp(W_{ni})] = \int \mathcal{N}(W_{ni}|\mu_{W_{ni}}, \Sigma_{W_{ni}}) \exp(W_{ni}) dW_{ni}$$

$$= C_1 \cdot \int \exp\left(-\frac{1}{2}(W_{ni} - \mu_{W_{ni}})^T \Sigma_{W_{ni}}^{-1} (W_{ni} - \mu_{W_{ni}}) + W_{ni}\right) dW_{ni}$$

$$= C_1 \cdot \int \exp\left(-\frac{1}{2}(W_{ni} - (\mu_{W_{ni}} + \Sigma_{W_{ni}}))^T \Sigma_{W_{ni}}^{-1} (W_{ni} - (\mu_{W_{ni}} + \Sigma_{W_{ni}}))\right)$$

$$\cdot \exp(\mu_{W_{ni}} + \frac{1}{2} \Sigma_{W_{ni}}) dW_{ni}$$

$$= \exp(\mu_{W_{ni}} + \frac{1}{2} \Sigma_{W_{ni}})$$

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

Similarly we can derive the expectation of the square forms:

$$E[\exp(W_{ni})^T \exp(W_{ni})] = \exp(2 \cdot (\mu_{W_{ni}} + \Sigma_{W_{ni}}))$$

and:

$$E[\exp(W_{ni})^T \Sigma \exp(W_{ni})] = \int \mathcal{N}(W_{ni}|\mu_{W_{ni}}, \Sigma_{W_{ni}}) \exp(W_{ni})^T \Sigma \exp(W_{ni}) dW_{ni}$$

$$= \exp(\mu_{W_{ni}} + \frac{1}{2} \Sigma_{W_{ni}})^T \Sigma \exp(\mu_{W_{ni}} + \frac{1}{2} \Sigma_{W_{ni}})$$

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

$$q(\{f_a, u_a\}_{a=1}^A) = \prod_{a=1}^A p(f_a|\{X_a^{(k)}, r_a^{(k)}, u_a^{(k)}\}_{k=1}^{K_a}) q(\{u_a^{(k)}\}_{k=1}^{K_a})$$

(12)

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

$$\mathcal{L} = E_{q(\{f_a, u_a\}_{a=1}^A)} \left[ \log \frac{p(Y_a, f_a, u_a | X_a, f_a^{(k)}, u_a^{(k)})}{q(\{f_a, u_a\}_{a=1}^A)} \right]$$

$$= E \left[ \log \left( \prod_{a=1}^A \prod_{k=1}^{K_a} p(Y_a|X_a, f_a^{(k)}) \prod_{a=1}^A (p(f_a|\{X_a^{(k)}, r_a^{(k)}, u_a^{(k)}\}_{k=1}^{K_a}) p(\{u_a^{(k)}\}_{k=1}^{K_a})) \right) q(\{u_a^{(k)}\}_{k=1}^{K_a}) \right].$$

(13)
We now derive the closed form expected log likelihood (ELL) from Eq. 14. The ELL is:

\[ \mathcal{L}_{MR-DGP} = \mathbb{E}_q(\{\mathbf{r}_a, \mathbf{u}_a\})_{a=1}^A \left[ \log \left( \prod_{a=1}^A \prod_{k=1}^{K_a} p(\mathbf{Y}_a|\mathbf{X}_a, \mathbf{r}_a^{(k)}) \cdot \prod_{a=1}^A p(\{\mathbf{u}_a^{(k)}\})_{k=1}^K \right) \right] \]

\[ = \mathbb{E} \left[ \log \prod_{a=1}^A \prod_{k=1}^{K_a} p(\mathbf{Y}_a|\mathbf{X}_a, \mathbf{r}_a^{(k)}) \right] + \mathbb{E} \left[ \log \prod_{a=1}^A p(\{\mathbf{u}_a^{(k)}\})_{k=1}^K \right] \]

\[ = \sum_{a=1}^A \mathbb{E}_q(\mathbf{r}_a) \left[ \log p(\mathbf{Y}_a|\mathbf{X}_a, \mathbf{r}_a) \right] + \sum_{a=1}^A \mathbb{E}_q(\mathbf{u}_a) \left[ \log \frac{p(\mathbf{u}_a)}{q(\mathbf{u}_a)} \right] - KL(q(\mathbf{U})||p(\mathbf{U})) \]

\[ C \quad \text{MR-GPRN: Variational Lower Bound} \]

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

Following [9] we keep \( \mathbf{U}, \mathbf{V} \) explicit in our approximate posterior. The goal is now to learn the augmented posterior \( p(\mathbf{f}, \mathbf{W}, \mathbf{U}, \mathbf{V}, \theta) \). 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:

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

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

\[ C.1 \quad \text{MR-GPRN: Closed Form Expected Log Likelihood} \]

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

\[ \mathcal{L}_{ell} = \sum_{a=1}^A \alpha_a \mathbb{E}_q \left[ \log \mathcal{N}(\mathbf{Y}_a|\mathbf{1}_{R_a} \sum_{r=1}^{R_a} (\mathbf{Wf}(r), \sigma_a^2) \right] \quad (15) \]

where each of the components can now be dealt with separately. Dealing with component \( a \):
We now deal with each of the expectations separately.

C.1.1 ELL: 1st Term

The first expectation does not contain \( f \) or \( W \) and so the expectations can be dropped:

\[
E_q \left[ (Y_{ni}^{(a)})^T Y_{ni}^{(a)} \right] = Y_{ni}^{(a)} Y_{ni}^{(a)} \tag{16}
\]

C.1.2 ELL: 2nd and 3rd Term

In the 2nd and 3rd terms the expectation is brought inside the sum and applied to \( f \) and \( W \) separately:

\[
E_q \left[ \frac{1}{R_a} \sum_{r} W(r)_{ni} f(r)_{n}^T Y_{ni}^{(a)} \right] = \frac{1}{R_a} \sum_{r} E_q(W(r)_{ni}) \left[ E_q[f(r)_{n}]^T Y_{ni}^{(a)} \right] \tag{17}
\]

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 \( f \) and \( W \) will appear in square forms, when \( a \neq b \) the expectation can be treated as in Sec. C.1.2

\[
E_q \left[ (\sum_r W(r)_{ni} f(r)_{n})^T (\sum_r W(r)_{ni} f(r)_{n}) \right]
\]

\[
= \frac{1}{R_a^2} \sum_a \sum_b E_q(f_a) q(W_{ni}) [f(a)_{n}^T W(a)_{ni} f(b)_{n}] \tag{18}
\]

case \( a = b \):

\[
\int q(f_a) q(W_{ni}) f(a)_{n}^T W(a)_{ni} f(a)_{n} dW_a dW_{ni}
\]

\[
= \int q(f_a) q(W_{ni}^{a}) f(a)^T W(a)_{ni} f(a) dW_a^{a} dW_{ni}^{a} \tag{19}
\]

\[
= Tr[\Sigma_{n} \Sigma_{f_{n}^a}] + \mu_{W_{ni}^a} \mu_{f_{n}^a} + \mu_{W_{ni}^a} \Sigma_{f_{n}^a} \mu_{W_{ni}^a} + \mu_{F_{n}^a} \Sigma_{F_{n}^a} \mu_{W_{ni}^a} + \mu_{F_{n}^a} \Sigma_{F_{n}^a} \mu_{W_{ni}^a} \mu_{f_{n}^a}
\]
case $a \neq b$:

$$
\int q(f_n) f(a)^T_n \int q(W_{ni}) W(a)^T_{ni} W(b)_{ni} dW_{ni} f(b)_n \ df_n \\
= \int q(f_n) f(a)^T_n \int q(W_{ni}) \int q(W^a_{ni}) W(a)^T_{ni} W(b)_{ni} dW^a_{ni} dW^b_{ni} dW_{ni} f(b)_n \ df_n \\
= \int q(f_n) f(a)^T_n \mu_{W_{ni}}^T \mu_{W_{ni}} f(b) \ df_n \\
= \mu^T_{E_a} \mu_{W_{ni}}^T \mu_{W_{ni}} \\
$$

(20)

Because we are summing over $a, b$ we can write

$$
\mathbb{E}_q \left[ \left( \sum_{r=a}^{R_a} W(r)_{ni} f(r)_n \right)^T \left( \sum_{r=a}^{R_a} W(r)_{ni} f(r)_n \right) \right] = \left( \frac{1}{R_a} \sum_{b=a}^{R_a} \mu_{W_{ni}}^T \mu_{W_{ni}} \right) \\
+ \left( \frac{1}{R_a} \sum_{b=a}^{R_a} T \Sigma W_{ni} \Sigma W_{ni} \right) + \mu_{E_a} \Sigma W_{ni} \Sigma W_{ni} + \mu_{E_a} \Sigma W_{ni} \Sigma W_{ni} \\
$$

(21)

C.2 MR-GPRN: Closed Form Expected Log Likelihood ($W \rightarrow \exp(W)$)

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

$$
\mathcal{L}_{ell} = \sum_{a=1}^{A} \alpha \mathbb{E}_q \left[ \log N(Y_a \mid \frac{1}{R_a} \sum_{r=1}^{R_a} (\exp(W) f(r), \sigma_a^2) \right] \\
$$

(22)

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

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

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:

$$
\mathbb{E}_q \left[ (Y_{ni}^{(a)})^T Y_{ni}^{(a)} \right] = Y_{ni}^{(a)^T} Y_{ni}^{(a)} \\
$$

(23)

C.2.2 ELL: 2nd and 3rd Term

As in Sec. C.1.2 the expectation is applied to $f$ and $W$ separately. To evaluate the expectation of $\exp W$ we use the result from Eq. 20.
\[ \mathbb{E}_q \left[ \frac{1}{R_a} \sum_r \exp(\mathbf{W}(r)_{ni}) \mathbf{f}(r)_n^T \mathbf{Y}_{ni}^{(a)} \right] = \left( \frac{1}{R_a} \sum_r \mathbb{E}_q(\mathbf{W}_{ni}) \left[ \exp(\mathbf{W}(r)_{ni}) \right] \mathbb{E}_q(\mathbf{f}_n) \left[ \mathbf{f}(r)_n^T \right] \mathbf{Y}_{ni}^{(a)} \right) \]

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

(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 f and \( \mathbf{W} \) will appear in square forms, when \( a \neq b \) the expectation can be treated as in Sec. C.2.2

\[ \mathbb{E}_q \left[ \sum_r \exp(\mathbf{W}(r)_{ni}) \mathbf{f}(r)_n^T \sum_r \exp(\mathbf{W}(r)_{ni}) \mathbf{f}(r)_n \right] \]

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

(25)

case \( a = b \):

\[ \int q(\mathbf{f}_n)q(\mathbf{W}_{ni}^a) \mathbf{f}(a)_n^T \exp(\mathbf{W}(a)_{ni})^T \exp(\mathbf{W}(a)_{ni}) \mathbf{f}(a)_n^T \exp(\mathbf{W}(a)_{ni}) \mathbf{f}(a)_n^T \mathbf{d}\mathbf{f}_n \mathbf{d}\mathbf{W}_{ni}^a \]

\[ = \int q(\mathbf{f}_n)^T \mathbf{f}(a)_n^T \exp(2 \cdot (\mathbb{E}_q(\mathbf{W}_{ni}^a) [\mathbf{W}_{ni}^a] + \mathbb{V}_q(\mathbf{W}_{ni}^a) [\mathbf{W}_{ni}^a])) \mathbf{f}(a)_n \mathbf{d}\mathbf{f}_n \]

\[ = \mu_{\mathbf{f}_n}^a \exp(2 \cdot (\mathbb{E}_q(\mathbf{W}_{ni}^a) [\mathbf{W}_{ni}^a] + \mathbb{V}_q(\mathbf{W}_{ni}^a) [\mathbf{W}_{ni}^a])) \mu_{\mathbf{f}_n}^a \]

\[ + Tr \left[ \exp(2 \cdot (\mathbb{E}_q(\mathbf{W}_{ni}^a) [\mathbf{W}_{ni}^a] + \mathbb{V}_q(\mathbf{W}_{ni}^a) [\mathbf{W}_{ni}^a])) \Sigma_{\mathbf{f}_n}^a \right] \]

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

In the case of \( a \neq b \):

\[ \int q(\mathbf{f}_n)^T \int q(\mathbf{W}_{ni}^c) \int q(\mathbf{W}_{ni}^b) \exp(\mathbf{W}(a)_{ni})^T \exp(\mathbf{W}(b)_{ni}) \mathbf{d}\mathbf{W}_{ni}^a \mathbf{d}\mathbf{W}_{ni}^b \mathbf{d}\mathbf{W}_{ni}^c \mathbf{f}(b)_n \mathbf{d}\mathbf{f}_n \]

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

\[ = \mu_{\mathbf{f}_n}^a \exp(\mu_{\mathbf{W}_{ni}^a} + \Sigma_{\mathbf{f}_n}^a)^T \exp(\mu_{\mathbf{W}_{ni}^b} + \Sigma_{\mathbf{f}_n}^b) \mu_{\mathbf{f}_n}^b \]

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:

\[ y_i^{(1)} = f(x_i) + \epsilon_1 \]
\[ y_j^{(2)} = \frac{1}{2} \sum_{k=2j-1}^{2j} y_k^{(1)} + \epsilon_2 \]  

(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 \( Y_1 = \{y_i^{(1)}\}_{i=1}^{2N} \), \( Y_2 = \{y_j^{(2)}\}_{j=1}^{N} \) is given by:

\[ L(Y_1, Y_2) = p(Y_1|f(x), \sigma_1^2)p(Y_2|Y_1, \sigma_2^2) \]  

(27)

When the data from \( Y_1 \) has the same support as the observation process \( Y_2 \), the evidence from \( Y_2 \) will not affect parameter estimation in the probability function \( p(Y_1|f(x), \sigma_1) \). However, when \( Y_2 \) has different support from the observed \( Y_1 \), the additional evidence should impact parameter inference. As \( Y_2 \) 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 \( Y_1 \) and \( Y_2 \) through a non-parametric prior over the latent function \( f(x) \).

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:

\[ f(x) \sim \mathcal{GP}(0, k(x, x')) \]  

(28)

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

\[ k(x, x') = A \exp\left(-\frac{(x - x')^2}{l}\right) \]  

(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 \( Y_1 \) and \( Y_2 \) as:

\[ p(Y_1, Y_2, f(x)) = p(f(x)|\theta)p(Y_1|f(x), \sigma_1^2)p(Y_2|Y_1, \sigma_2^2) \]  

(30)

Where \( \theta \) is the hyper-parameters for the Gaussian process. We can write down the distribution of \( Y_1 \) and \( Y_2 \) by marginalizing out the latent function:

\[ p(Y_1, Y_2) = \int p(f(x)|\theta)p(Y_1|f(x), \sigma_1^2)p(Y_2|Y_1, \sigma_2^2)df(x) \]  

(31)

As \( Y_1 \) has all the information of \( f(x) \), this integral is tractable and we can write \( y^{(1)} \sim \mathcal{GP}(0, k(x, x') + \sigma_1) \). But when the aggregation function \( Y_2 \) has additional information about the latent function, i.e. \( Y_1 \) and \( Y_2 \) only partially overlapping, bringing additional information from \( Y_2 \) requires the prediction of the missing values of the corresponding \( Y_1 \) process. This can be done in Markov Chain Monte Carlo (MCMC) setting by treating the unobserved value of \( Y_1 \) as extra parameters. However, this increase a lot of computational complexity for the MCMC sampler. One option is to make an independence assumption for \( Y_1 \) and \( Y_2 \). Thus, the information in \( Y_2 \) can affect the latent function \( f \) 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 $Y_1$ and $Y_2$, we have:

$$p(Y_1, Y_2) = \int p(f(x)|\theta)p(Y_1|f(x), \sigma_1) p(Y_2|f(x), \sigma_2) df(x)$$ (32)

Instead of assuming the conditional probability $p(Y_2|Y_1, \sigma_2^2)$, we are now assuming the data depends on the latent function $f(x)$ directly. However, when $Y_1$ and $Y_2$ are different resolutions under the same support, this likelihood mis-specifies 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 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:

$$L(Y_1, Y_2) = \int p(f(x)|\theta)p(Y_1|f(x), \sigma_1^2) p(Y_2|f(x), \sigma_2^2)^\alpha df(x)$$ (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:

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

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

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

$$\hat{\theta} - \theta_0 \rightarrow -\ell'(\theta_0) / \ell''(\theta_0)$$ (36)

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

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

The expected variance from the composite likelihood model is:

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

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

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

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

$$\mathbb{C}H(\hat{\theta}|Y)C^T = H^{-1}(\hat{\theta}|Y)J(\hat{\theta}|Y)H^{-1}(\hat{\theta}|Y)$$ (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. \cite{lyddon2020composite} and Ribatet \cite{ribatet2015bayesian} developed two different ways to adjust the identities of the covariance matrix:

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

(41)

Where $\alpha_{\text{Ribatet}}$ considers all the information in the covariance matrix and $\alpha_{\text{Lyddon}}$ only matches the information in the diagonal elements.

\section{D.5 MCMC Composite Likelihood Experiments}

We now construct an MCMC experiment for the synthetic data using Eq. $\ref{eq:alp}$ Instead of sampling directly from the intractable joint distribution of $L(Y_1, Y_2)$, we sample from the joint probability with the latent variable $L(Y_1, Y_2, f(x))$ via a Metropolis-Hastings within Gibbs sampler. We perform three block updates: on $\theta_0$ for the Gaussian process prior, $f(x)$ for the latent function variables and $\sigma^2 = \{\sigma_1^2, \sigma_2^2\}$ for the noise parameter.

\begin{algorithm}
\caption{Block Metropolis-Hastings within Gibbs}
\begin{algorithmic}
\Input{Observed datasets $\{(X_s, Y_s)\}_{s=1}^{S}$, initial parameters $\theta_0$,}
\For{$i$-th iteration} 
\State Update parameter block $\theta_i$
\Function{Block}{$\theta_i$}
\State 1. Sample proposed value of the Gaussian process prior $\theta_i' \sim N(\theta_i, \Delta)$
\State 2. Calculate the conditional probability distribution $p(\theta_i' | Y_1, Y_2, \sigma_{i-1}^2, f(x)|_{i-1})$
\State 3. Calculate the acceptance rejection ratio:
\State \quad $\pi = \frac{p(\theta_i' | Y_1, Y_2, \sigma_{i-1}^2, f(x)|_{i-1})}{p(\theta_i | Y_1, Y_2, \sigma_{i-1}^2, f(x)|_{i-1})}$
\State 4. Update the $i$-th value of $\theta_i$ via $\pi$
\EndFunction
\State Update the parameter block $f(x)$ via $\pi = \frac{p(f(x)| Y_1, Y_2, \sigma_{i-1}^2, \theta_i)}{p(f(x)| Y_1, Y_2, \sigma_{i-1}^2, \theta_i)}$
\State Update the parameter block $\sigma_1$ via $\pi = \frac{p(\sigma_1^2 | Y_1, Y_2, \theta_i, f(x)|_{i-1})}{p(\sigma_{i-1}^2 | Y_1, Y_2, \theta_i, f(x)|_{i-1})}$
\EndFor
\Return $\theta, f(x), \sigma^2$
\end{algorithmic}
\end{algorithm}

\section{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.

\textbf{Data Generation} We consider the case of having two dependent observation processes. We generate one process $Y_1 = 5 \cdot \sin(X)^2 + 0.1 \cdot \epsilon$ with $\epsilon \sim \mathcal{N}(0, 1)$ with 100 samples over the range $[-2, 15]$. For $Y_2$ we aggregate $Y_1$ into bins of size 3, $S_2 = 3$, so that $Y_2 \in \mathbb{R}^{33}$ and $X_2 \in \mathbb{R}^{33 \times 3}$. In Fig. 2 we only plot the range $[3, 10]$.

\textbf{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 $\sigma^2_{\text{noise}}$ 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 VBA-GG-NORMAL we run for 10000 epochs. For MR-GP for we run the MLE estimate for 10000 epochs, obtain $\alpha_{\text{Ribatet}}$ 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 $f$ to be 0.1 and the lengthscales of $W$ 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 NO₂

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) and land use (UKMap). 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 the 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 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 \( X_2, Y_2 \) corresponds to the LAQN dataset and \( X_1, Y_1 \) 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 \( f^{(1)}_1 \) and \( f^{(1)}_2 \) we place 300 inducing points and for \( f^{(2)}_2 \) 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 \( f^{(1)}_1 \) and \( f^{(1)}_2 \) and K=100 for \( f^{(2)}_2 \).

**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 \( y^n \) is the aggregate output of some bag of items \( x^a = \{ x^a_i \}_{i=1}^N_a \). The likelihood of each bag is \( y^a | x^a \sim N(y|\eta^a, \tau^a) \) where \( \eta^a = \sum_{i=1}^{N_a} w^a_i \mu(x^a_i) \) and \( \mu \) 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]: \( \mu = W f \) and each dataset \( \{ X_a, Y_a \}_{a=1}^A \) directly corresponds to the observations and bag of items defined in VBAGG-NORMAL. Let \( N_a = S_a \), and \( \tau^a = \sigma^2_a \) 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 \( w^a_i = \frac{1}{S_a} \) we obtain \( y^a \sim N(\sum_{i=1}^{N_a} w^a_i \mu(x^a_i), \tau^a) \) which is MR-GPRN in the notation of [13]. VBAGG-NORMAL is then recovered when we use only one latent function (by setting \( W \) to a constant value), by only considering the single task setting and by setting the composite weight to one.

---


[2] https://www.geoinformationgroup.co.uk/ukmap