Outlier-Robust Multi-Output Gaussian Processes

Joshua Rooijakkers
Imperial College London

Leiv Rønneberg
University of Oslo

François-Xavier Briol
University College London

Jeremias Knoblauch
University College London

Matias Altamirano
University of Oxford

Abstract

Multi-output Gaussian process (MOGP) regression allows modelling dependencies among multiple correlated response variables. Similarly to standard Gaussian processes (GPs), MOGPs are sensitive to model misspecification and outliers, which can distort predictions within individual outputs. In the multi-output case, this situation can be further exacerbated as anomalous observations in one response can adversely influence predictions for other correlated outputs. To handle this situation, we propose R-MOGP—an MOGP that is provably robust to outliers in any individual response. Our approach preserves conjugacy, achieving robustness without sacrificing computational efficiency, and integrates naturally with sparse variational inference for scalability. Empirically, we show that R-MOGP achieves performance comparable to existing state-of-the-art robust MOGPs at a fraction of their computational costs.

1 Introduction

Gaussian processes (GPs; Rasmussen and Williams, 2006) provide flexible Bayesian models for latent functions which are widely used in statistics and machine learning. While GPs are usually formulated as having a single response variable (and hence a single output), many real-world applications require simultaneous predictions of multiple interacting response variables, leading to the natural extension to *multi-output Gaussian processes* (MOGPs; Bonilla et al., 2007; Álvarez et al., 2012). By modelling these correlated variables jointly, MOGPs are well-suited for a wide variety of tasks, such as multi-task regression (Álvarez et al., 2012), causal inference (Alaa and Van Der Schaar, 2017; Dimitriou et al., 2024), spatio-temporal modelling (Genton and Kleiber, 2015), numerical integration (Xi et al., 2018; Sun et al., 2023) and multi-task optimisation (Swersky et al., 2013; Maddox et al., 2021). Similarly to standard GPs, MOGPs offer an exact Bayesian closed-form posterior under the assumption of a Gaussian likelihood (Álvarez et al., 2012). This assumption is computationally convenient, but also makes MOGPs vulnerable to outliers, model misspecification, or non-Gaussian noise (Wynne et al., 2021). As a result, predictions and uncertainty estimates may become unreliable. In the multi-output setting, however, this issue is even more pronounced: contaminated observations in one output can not only distort predictions for that output but also propagate errors to correlated outputs. This is a common problem in various applications, including in cancer research, where MOGPs are used to model how the viability of different correlated cancer cell samples respond to varying drug doses (Rønneberg et al., 2023).

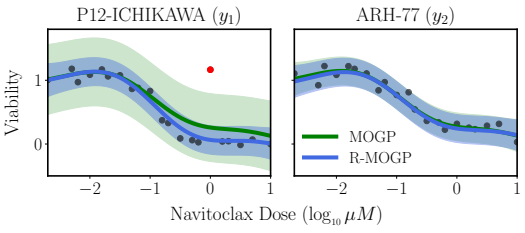


Figure 1: Navitoclax dose-response with outlier. We show the viability of two cancer cell lines exposed to varying doses of the drug Navitoclax.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Such experiments often produce extremely noisy measurements due to the heterogeneity of cell growth and technical issues with the experimental assays, such as evaporation or contamination—which in turn distorts MOGP predictions. Figure 1 illustrates this for the drug Navitoclax, where such an outlier (marked by the red dot) causes the MOGP predictions to lie above the observed dose-response values, with increased uncertainty. If unaccounted for, this can bias summary statistics commonly used to link genetic markers to drug efficacy, and ultimately lead to incorrect treatments and poor health outcomes.

To address these challenges, prior work has replaced Gaussian noise with heavier-tailed alternatives such as the Student-t, Laplace, and mixture distributions (Jylänki et al., 2011; Huang et al., 2024; Kuss, 2006; Lu et al., 2023). Alternatively, Yu et al. (2007) introduce a Student-t process prior over the latent functions. While these approaches offer greater robustness to outliers, they also break conjugacy, and therefore require approximate inference and additional non-convex optimisation routines with overall computational costs that far exceed their conjugate counterparts. This is already problematic for standard GPs, but even more consequential for MOGPs: even conjugate inference in MOGPs scales cubically in both the number of samples *and* the number of outputs.

In this paper, we introduce R-MOGP, which is *fully conjugate* for regression, avoiding the need for additional computationally expensive optimisation techniques. Unlike standard conjugate MOGPs, R-MOGP is also robust: we theoretically show that the method is provably robust against unbounded corruptions in any individual response, both within a single output and across correlated outputs. We further propose novel parameter-optimisation procedures that improve robustness by borrowing statistical strength across outputs. R-MOGPs are flexible, and accommodate *heterotopic data*—a common scenario in multi-output problems where we do not observe data for some of the output channels. They also naturally integrate with the sparse variational framework to further improve scalability. Empirically, we demonstrate that R-MOGPs achieve performance comparable to existing robust MOGPs at a fraction of their computational cost. Our contributions are summarised as follows:

1. We introduce a generalised, robust form of MOGPs and derive the associated closed forms for the posterior and predictive distributions.
2. We theoretically show that, unlike MOGPs, R-MOGP is robust to arbitrarily extreme outliers.
3. We show that R-MOGP is competitive with existing state-of-the-art robust MOGPs while requiring substantially less computation.

2 Background

GP Regression Let $\mathcal{D} = \{X, \mathbf{y}\}$ be a dataset of N observations, where $X = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^\top$ and $\mathbf{y} = [y_1, \ldots, y_N]^\top$, with $\mathbf{x}_i \in \mathcal{X} \subseteq \mathbb{R}^d$ and $y_i \in \mathbb{R}$ for all $i \in \{1, \ldots, N\}$. GP regression (Stein, 1999; Rasmussen and Williams, 2006; Garnett, 2023) models these observations as noisy evaluations of a latent function $f : \mathcal{X} \to \mathbb{R}$ with additive noise $\varepsilon_i \in \mathbb{R}$ via $y_i = f(\mathbf{x}_i) + \varepsilon_i$ for all $i \in \{1, \ldots, N\}$. Our prior beliefs about the unknown function f are expressed through a GP prior $f \sim \mathcal{GP}(m, \kappa)$ for $m : \mathcal{X} \to \mathbb{R}$ and $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ denoting *scalar-valued* mean and covariance (or kernel) functions, respectively. Here, m and κ capture our prior beliefs about properties such as smoothness and sparsity of f, and typically depend on hyperparameters $\boldsymbol{\theta} \in \Theta$ which are suppressed from notation for legibility. Under the GP prior, the corresponding latent function vector $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)]^\top$ follows an N-dimensional Gaussian distribution with density $p(\mathbf{f}|X) = \mathcal{N}(\mathbf{f}; \mathbf{m}, K)$, where $\mathbf{m} = [m(\mathbf{x}_1), \ldots, m(\mathbf{x}_N)]^\top$ and $K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$.

If $\boldsymbol{\varepsilon} = [\varepsilon_1, \ldots, \varepsilon_N]^\top \in \mathbb{R}^N$ is assumed to be Gaussian so that $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \sigma^2 I_N)$, where I_N is the N-dimensional identity matrix, the posterior predictive density for $f_\star = f(\mathbf{x}_\star)$ at $\mathbf{x}_\star \in \mathcal{X}$ has the closed form $p(f_\star|\mathbf{x}_\star, \mathbf{y}, X) = \mathcal{N}(f_\star; \mu_\star^{\text{GP}}, \Sigma_\star^{\text{GP}})$, where

$$\begin{aligned} \mu_\star^{\text{GP}} &= m_\star + k_\star^\top (K + \sigma^2 I_N)^{-1} (\mathbf{y} - \mathbf{m}), \\ \Sigma_\star^{\text{GP}} &= k_{\star\star} - k_\star^\top (K + \sigma^2 I_N)^{-1} k_\star, \end{aligned} \tag{1}$$

where $k_\star = [\kappa(\mathbf{x}_\star, \mathbf{x}_1), \ldots, \kappa(\mathbf{x}_\star, \mathbf{x}_N)]^\top$, $k_{\star\star} = \kappa(\mathbf{x}_\star, \mathbf{x}_\star)$ and $m_\star = m(\mathbf{x}_\star)$. The green colour is later used to highlight differences between GPs and RCGPs.

Robust and Conjugate GPs The closed-form predictive in (1) relies on the assumption that the observation noise $\boldsymbol{\varepsilon}$ is Gaussian. If this assumption is accurate, a Bayesian approach constitutes a

way of learning from data which is in some sense optimal (Zellner, 1988). Critically however, this optimality breaks down in the presence of model misspecification, such as in the case of heavy-tailed noise or outliers when working with GPs. To address this issue, recent work has explored generalised Bayesian (GB) inference (Grünwald, 2012; Bissiri et al., 2016; Ghosh and Basu, 2016; Miller and Dunson, 2019; Knoblauch et al., 2022; Dellaporta et al., 2022; Fong et al., 2023; Wild et al., 2023; McLatchie et al., 2025a,b; Shen et al., 2025). The generalised Bayesian posterior is given by

$$p_{\text{GB}}(\mathbf{f}|\mathbf{y}, X) \propto p(\mathbf{f}|X) \exp(-N\mathcal{L}(\mathbf{f}, \mathbf{y}, X)), \tag{2}$$

where $\mathcal{L}$ denotes an empirical loss function measuring the disagreement between the statistical model and the observed data, and different loss functions are chosen to provide robustness or computational efficiency (see e.g. Knoblauch et al., 2018; Futami et al., 2018; Boustati et al., 2020; Matsubara et al., 2022, 2024; Altamirano et al., 2023; Duran-Martin et al., 2024).

Recent work (Altamirano et al., 2024; Sinaga et al., 2025; Laplante et al., 2025; Ezzerg et al., 2026) has applied generalised Bayesian inference to GPs by using a *weighted Fisher divergence* as the loss function (Hyvärinen, 2005; Barp et al., 2019; Xu et al., 2022), given by

$$\mathcal{L}_{\mathbf{w}}(\mathbf{f}, \mathbf{y}, X) = \mathbb{E}_{\tilde{\mathbf{x}} \sim p_{0,x}} \left[\mathbb{E}_{\tilde{y} \sim p_0(\cdot|\tilde{\mathbf{x}})} \left[\| w(\tilde{\mathbf{x}}, \tilde{y})(s_{\text{model}}(\tilde{\mathbf{x}}, \tilde{y}) - s_{\text{data}}(\tilde{\mathbf{x}}, \tilde{y})) \|_2^2 \right] \right] \tag{3}$$

where s_{model} and s_{data} are the scores of the respective posited and true conditional likelihoods $p_\theta(y|\mathbf{x})$ and $p_0(y|\mathbf{x})$, i.e. $s_{\text{model}} = \nabla_y \log p_\theta(y|\mathbf{x})$. Importantly, $w : \mathcal{X} \times \mathbb{R} \to \mathbb{R}\backslash\{0\}$ is a continuously differentiable weight function that downweights extreme observations. When positing a Gaussian model, Altamirano et al. (2024) show that the GB posterior with the above loss function is conjugate, and its closed-form predictive is given by $p_w(f_\star|x_\star, \mathbf{x}, \mathbf{y}) = \mathcal{N}(f_\star; \mu_\star^{\text{RCGP}}, \Sigma_\star^{\text{RCGP}})$, where

$$\begin{aligned} \mu_\star^{\text{RCGP}} &= m_\star + k_\star^\top (K + \sigma^2 \mathbf{J_w})^{-1} (\mathbf{y} - \mathbf{m_w}), \\ \Sigma_\star^{\text{RCGP}} &= k_{\star\star} - k_\star^\top (K + \sigma^2 \mathbf{J_w})^{-1} k_\star, \end{aligned} \tag{4}$$

where the matrix $\mathbf{J_w} = \text{diag}(\sigma^2 \mathbf{w}^{-2}/2) \in \mathbb{R}^{N \times N}$, the vector $\mathbf{m_w} = \mathbf{m} + \sigma^2 \nabla_y \log(\mathbf{w}^2) \in \mathbb{R}^N$, and $\mathbf{w} = [w(\mathbf{x}_1, y_1), \ldots, w(\mathbf{x}_N, y_N)]^\top \in \mathbb{R}^N$. While this closely resembles the results of standard GPs in (1), there are two key differences highlighted in **blue**. Note further that the above is a strict generalisation of GPs: the specific choice $w(\mathbf{x}, y) = \sigma/\sqrt{2}$ recovers standard GPs.

Multi-Output GPs In many applications, we are not interested in modelling a scalar-valued function, but a vector-valued latent function $\mathbf{f} : \mathcal{X} \to \mathbb{R}^T$, where each component $f_t : \mathcal{X} \to \mathbb{R}$ corresponds to one of T outputs such that $\mathbf{f} = [f_1, \ldots, f_T]^\top$. MOGPs extend standard GPs to this setting by also capturing the correlations across output dimensions, also known as co-kriging (Álvarez et al., 2012). From here on out, we focus exclusively on MOGPs, and all notation and definitions should be interpreted in this context. Let $Y = [\mathbf{y}_1, \ldots, \mathbf{y}_N]^\top$ and $\mathcal{E} = [\boldsymbol{\varepsilon}_1, \ldots, \boldsymbol{\varepsilon}_N]^\top$ denote $N \times T$ matrices where $\mathbf{y}_i = [y_{i,1}, \ldots, y_{i,T}]^\top$ and $\boldsymbol{\varepsilon}_i = [\varepsilon_{i,1}, \ldots, \varepsilon_{i,T}]^\top$.

We place a GP prior on the latent function $\mathbf{f} \sim \mathcal{GP}(\mathbf{m}, \mathcal{K})$, where, in contrast to scalar GPs, the mean $\mathbf{m} : \mathcal{X} \to \mathbb{R}^T$ is now *vector-valued*, and the kernel function $\mathcal{K} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{T \times T}$ is *matrix-valued* (Micchelli and Pontil, 2004). In line with this, the T elements in $\mathbf{m} = [m_1, \ldots, m_T]^\top$ denote the prior means for the T outputs. Additionally, the kernel $\mathcal{K}(\mathbf{x}, \mathbf{x}')$ is a positive semidefinite matrix-valued function, whose entries $[\mathcal{K}(\mathbf{x}, \mathbf{x}')]_{t,t'}$ encode the covariance between the outputs $f_t(\mathbf{x})$ and $f_{t'}(\mathbf{x}')$ and capture both the smoothness within the same output with respect to inputs $\mathbf{x}$ and $\mathbf{x}'$ and the correlations across outputs t and t'. Common approaches include the intrinsic coregionalisation model (ICM; Goovaerts, 1997), the linear model of coregionalisation (Alvarez et al., 2013), process convolutions (Journel and Huijbregts, 1978) and spectral mixtures (Parra and Tobar, 2017). For example, in the ICM, the kernel is assumed to be separable, such that it can be written as the product $[\mathcal{K}(\mathbf{x}, \mathbf{x}')]_{t,t'} = B_{t,t'} \kappa(\mathbf{x}, \mathbf{x}')$ with a coregionalisation matrix $B \in \mathbb{R}^{T \times T}$, which represents the covariance matrix of the latent functions across outputs, and a scalar-valued kernel $\kappa(\mathbf{x}, \mathbf{x}')$ representing properties of individual outputs.

When assuming that observations follow a Gaussian distribution and are independent across outputs such that $\text{vec}(\mathcal{E}) \sim \mathcal{N}(0, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma} = \Sigma \otimes I_N \in \mathbb{R}^{NT \times NT}$ with $\Sigma = \text{diag}(\sigma_1^2, \ldots, \sigma_T^2) \in \mathbb{R}^{T \times T}$ denoting the variances in each output, the MOGP posterior and predictive are available in closed form. Here, $\text{vec}(\cdot)$ denotes vectorisation of a matrix by stacking its columns, and $\otimes$ denotes the Kronecker product. We define $\mathbf{y}_{\text{vec}} = \text{vec}(Y) \in \mathbb{R}^{NT}$ and $\mathbf{m}_{\text{vec}} = \text{vec}(M) \in \mathbb{R}^{NT}$, where $M = [\mathbf{m}_1, \ldots, \mathbf{m}_N]^\top$ with $\mathbf{m}_i = \mathbf{m}(\mathbf{x}_i)$. In addition, the $NT \times NT$ block-structured kernel matrix

is $\mathbf{K} = [\mathcal{K}_{tt'}(X,X)]_{t,t'=1}^{T}$, where each $\mathcal{K}_{tt'}(X,X)$ is the Gram matrix between inputs for outputs t and t'. Finally, let $F = [\mathbf{f}_1, \ldots, \mathbf{f}_N]^\top$ be the $N \times T$ matrix of latent function evaluations, with $\mathbf{f}_i = \mathbf{f}(\mathbf{x}_i) = [f_1(\mathbf{x}_i), \ldots, f_T(\mathbf{x}_i)]^\top$ and $\mathbf{f}_{\text{vec}} = \text{vec}(F)$. Then, the closed-form MOGP predictive density for $f_\star = [f_1(\mathbf{x}_\star), \ldots, f_T(\mathbf{x}_\star)]^\top$ is given by $p(f_\star|\mathbf{x}_\star, Y, X) = \mathcal{N}(f_\star; \boldsymbol{\mu}_\star^{\text{MOGP}}, \boldsymbol{\Sigma}_\star^{\text{MOGP}})$, where

$$\begin{aligned} \boldsymbol{\mu}_\star^{\text{MOGP}} &= \mathbf{m}_\star + \mathbf{k}_\star^\top (\mathbf{K} + \boldsymbol{\Sigma}\, I_{NT})^{-1} (\mathbf{y}_{\text{vec}} - \mathbf{m}_{\text{vec}}), \\ \boldsymbol{\Sigma}_\star^{\text{MOGP}} &= \mathbf{k}_{\star\star} - \mathbf{k}_\star^\top (\mathbf{K} + \boldsymbol{\Sigma}\, I_{NT})^{-1} \mathbf{k}_\star, \end{aligned} \tag{5}$$

where $\mathbf{k}_\star = [\mathcal{K}(\mathbf{x}_\star, \mathbf{x}_1), \ldots, \mathcal{K}(\mathbf{x}_\star, \mathbf{x}_N)]^\top \in \mathbb{R}^{NT \times T}$, $\mathbf{k}_{\star\star} = \mathcal{K}(\mathbf{x}_\star, \mathbf{x}_\star) \in \mathbb{R}^{T \times T}$ and $\mathbf{m}_\star = \mathbf{m}(\mathbf{x}_\star) \in \mathbb{R}^T$. We note that it is straightforward to apply the expressions above in the case of heterotopic data—where only $T_i < T$ outputs are observed for input $\mathbf{x}_i$, by simply restricting $\mathbf{y}_{\text{vec}}, \mathbf{m}_{\text{vec}}$ and the corresponding rows and columns of $\mathbf{K}$ and $\boldsymbol{\Sigma}$ to the observed entries. In general, computing the MOGP predictive in (5) requires inverting an $NT \times NT$ matrix, leading to a worst-case computational complexity of $\mathcal{O}(N^3T^3)$, which quickly becomes expensive as either the number of samples N or number of outputs T grows.

Despite their large computational cost, MOGPs are very useful, particularly for predicting multiple correlated response variables. In this setting, they share information across outputs—which independent scalar GPs cannot do. However, they also assume Gaussian errors. While this assumption enables conjugacy and closed-form posteriors, it also constitutes an even more pronounced problem than in the scalar case: outliers can distort not only the predictions corresponding to their own output, but also those of *other* correlated outputs. While several robust multi-output GP algorithms have been proposed (Jylänki et al., 2011; Yu et al., 2007), they do not result in exact conjugate updates. As a result, they further inflate the already substantive computational budget of MOGPs.

3 Methodology

We again consider the weighted Fisher divergence, but instead of the scalar observations in (3), we now focus on the case of vector-valued observations, as also studied in Barp et al. (2019); Altamirano et al. (2023). This divergence measures the discrepancy between vector-valued scores using a weight function $W : \mathcal{X} \times \mathbb{R}^T \to \mathbb{R}^{T \times T}$ returning sufficiently regular positive semi-definite matrices, and is given by

$$\mathcal{L}_{\mathbf{W}}(F, Y, X) = \mathbb{E}_{\tilde{\mathbf{x}} \sim p_{0,x}} \left[\mathbb{E}_{\tilde{\mathbf{y}} \sim p_0(\cdot|\tilde{\mathbf{x}})} [\| W(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})^\top \left(s_{\text{model}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - s_{\text{data}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \right) \|_2^2] \right], \tag{6}$$

where $\tilde{\mathbf{x}} \in \mathbb{R}^d$ and $\tilde{\mathbf{y}} \in \mathbb{R}^T$ are random variables and $p_{0,x}$ and p_0 now represent corresponding densities. Additionally, s_{model} and s_{data} denote T-sized score vectors of the model and the true data, where $s_{\text{data}}(\mathbf{x}, \mathbf{y}) = \nabla_{\mathbf{y}} \log p_0(\mathbf{y}|\mathbf{x})$ and $s_{\text{model}}(\mathbf{x}, \mathbf{y}) = \nabla_{\mathbf{y}} \log p_\theta(\mathbf{y}|\mathbf{x})$ for true and posited model p_0 and p_θ, respectively. Importantly, $W(\mathbf{x}, \mathbf{y})$ is now a $T \times T$ matrix that depends on observations across outputs. Mild assumptions on W, such as pointwise invertibility and continuous differentiability, are sufficient for conjugacy (Liu et al., 2022), but for practical purposes we choose a diagonal form such that $W_i := W(\mathbf{x}_i, \mathbf{y}_i) = \text{diag}(w_1(\mathbf{x}_i, \mathbf{y}_i), \ldots, w_T(\mathbf{x}_i, \mathbf{y}_i))$, where $w_t : \mathcal{X} \times \mathbb{R}^T \to \mathbb{R} \backslash \{0\}$ is a scalar-valued continuously-differentiable weight function as used in RCGPs. This structure makes our weights interpretable—each observation $y_{i,t}$ is weighted by $w_{i,t} := w_t(\mathbf{x}_i, \mathbf{y}_i)$—and facilitates robust hyperparameter optimisation. Using the above to construct a generalised posterior as in (2) then results in the R-MOGP predictive.

Proposition 3.1. *Let* $\text{vec}(\mathcal{E}) \sim \mathcal{N}(0, \boldsymbol{\Sigma})$ *where* $\boldsymbol{\Sigma} = \Sigma \otimes I_N$ *for a* T*-dimensional diagonal* Σ*, and* $W_i = \text{diag}(w_1(\mathbf{x}_i, \mathbf{y}_i), \ldots, w_T(\mathbf{x}_i, \mathbf{y}_i))$*, where* $y_{i,t} \mapsto w_t(\mathbf{x}_i, \mathbf{y}_i)$ *is continuously differentiable. Then, the R-MOGP predictive density is given by* $p_W(f_\star|\mathbf{x}_\star, Y, X) = \mathcal{N}(f_\star; \boldsymbol{\mu}_\star^{\text{RMOGP}}, \boldsymbol{\Sigma}_\star^{\text{RMOGP}})$*, where*

$$\begin{aligned} \boldsymbol{\mu}_\star^{\text{RMOGP}} &= \mathbf{m}_\star + \mathbf{k}_\star^\top (\mathbf{K} + \boldsymbol{\Sigma}\, \mathbf{J}_{\mathbf{W}})^{-1} (\mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{W},\text{vec}}), \\ \boldsymbol{\Sigma}_\star^{\text{RMOGP}} &= \mathbf{k}_{\star\star} - \mathbf{k}_\star^\top (\mathbf{K} + \boldsymbol{\Sigma}\, \mathbf{J}_{\mathbf{W}})^{-1} \mathbf{k}_\star, \end{aligned}$$

where $\mathbf{J}_{\mathbf{W}} = \frac{1}{2} \boldsymbol{\Sigma} \mathbf{W}^{-2} \in \mathbb{R}^{NT \times NT}$, $\mathbf{m}_{\mathbf{W},\text{vec}} = \mathbf{m}_{\text{vec}} + \boldsymbol{\Sigma}(\nabla_{\mathbf{y}} \cdot \log(\mathbf{W}^2)) \in \mathbb{R}^{NT}$, $\mathbf{W} = \text{diag}(\mathbf{W}_1, \ldots, \mathbf{W}_T) \in \mathbb{R}^{NT \times NT}$, $\mathbf{W}_t = \text{diag}(w_t(\mathbf{x}_1, \mathbf{y}_1), \ldots, w_t(\mathbf{x}_N, \mathbf{y}_N)) \in \mathbb{R}^{N \times N}$*, and* $\nabla_{\mathbf{y}} \cdot \log(\mathbf{W}^2) = [\nabla_{\mathbf{y}} \cdot \log(\mathbf{W}_1^2)^\top, \ldots, \nabla_{\mathbf{y}} \cdot \log(\mathbf{W}_T^2)^\top]^\top \in \mathbb{R}^{NT}$.

The proof is in Appendix B.1. In the above result, we have abused notation, and written $\log(\mathbf{W})$ to denote a diagonal matrix obtained from $\mathbf{W}$ by taking entry-wise logarithms along its diagonal. Additionally, $\nabla_{\mathbf{y}} \cdot M$ denotes the divergence of a matrix M (see Appendix A). The resulting expressions are similar to the standard MOGP case presented in (5), with the key terms of difference highlighted in **blue** and the dimensionality of each quantity has been specified for clarity. Importantly, these terms introduce weights that allow information to be shared to borrow strength across channels when detecting outliers and reducing their influence on posterior inferences. As in the scalar case, which can be recovered by taking $T = 1$, $w_t(\mathbf{x}_i, \mathbf{y}_i) = \sigma_t/\sqrt{2}$ recovers the standard MOGP predictive, and using $w_t(\mathbf{x}_i, \mathbf{y}_i) = w_t(\mathbf{x}_i)$ corresponds to an MOGP with heteroscedastic noise (Gammelli et al., 2022; Lee and Liu, 2023). Evaluating the closed-form posterior of Proposition 3.1 imposes a computational burden of order $\mathcal{O}(N^3T^3)$—the same cost imposed by standard MOGP. Unlike standard MOGP, the R-MOGP can be made provably robust through the introduction of the weight term.

3.1 Weight Function Choice

In this paper, we focus on robustness to sparse outliers: more precisely, we consider the setting where only some entries in the observation vector at a particular input point are outliers. We restrict attention to this case since it is the type of outlier most commonly observed in real-world data (e.g. Štefelová et al., 2021; Su et al., 2024). Nevertheless our framework is flexible: through a simple change of weighting function, it can be geared towards other types of data pollution—such as outliers that occur jointly across all channels, or outliers in a subset of channels. While many functional forms of $w_t(\mathbf{x}, \mathbf{y})$ are sufficient to achieve the type of robustness we desire, we adopt the inverse multi-quadratic kernel. We do so as prior work has shown that it provides a good balance for performance in both misspecified and well-specified settings (Matsubara et al., 2022; Altamirano et al., 2024; Laplante et al., 2025). This kernel incorporates a centering function $\gamma_t : \mathcal{X} \times \mathbb{R}^T \to \mathbb{R}$, a decay rate function $c_t : \mathcal{X} \times \mathbb{R}^T \to \mathbb{R}$, and a constant $\beta_t > 0$, and is given by

$$w_t(\mathbf{x}, \mathbf{y}) = \beta_t \left(1 + \left(\frac{y_t - \gamma_t(\mathbf{x}, \mathbf{y})}{c_t(\mathbf{x}, \mathbf{y})}\right)^2\right)^{-\frac{1}{2}}. \tag{7}$$

This weight is widely used in robust methods (Barp et al., 2019; Matsubara et al., 2024; Liu and Briol, 2024) and downweights outlying observations. Specifically, the mapping $y_t \mapsto w_t(\mathbf{x}, \mathbf{y})$ is heavy-tailed, symmetric around $\gamma_t(\mathbf{x}, \mathbf{y})$, and decreasing as $|y_t - \gamma_t(\mathbf{x}, \mathbf{y})|$ increases, where $c_t(\mathbf{x}, \mathbf{y})$ determines this rate of decrease. Finally, the constant β_t sets the maximum of the weight and corresponds to the GB learning rate (Wu and Martin, 2023). Although these hyperparameters can in principle be estimated from data, Altamirano et al. (2024) argue that this is suboptimal and proposed several heuristic choices that performed reasonably well in practice. For example, they recommend $\gamma_t(\mathbf{x}, \mathbf{y}) = m_t(\mathbf{x})$. While often effective, this makes the method sensitive to prior mean specification—a well-known limitation of RCGPs (Ament et al., 2024; Laplante et al., 2025).

Unlike in the scalar setting discussed in this prior work, the multi-output framework provides an elegant way for us to reduce this sensitivity by borrowing statistical strength across outputs. Intuitively speaking, it allows us to use outputs from other channels to make an educated guess about what an outlier looks like. In technical terms, this means that we define the centering function for output channel t as the conditional expectation given the observations in the other output channels. For $\mathbf{y}_{i,-t} = [y_{i,1}, \ldots, y_{i,t-1}, y_{i,t+1}, \ldots, y_{i,T}]^\top \in \mathbb{R}^{T-1}$, this results in $\gamma_t(\mathbf{x}_i, \mathbf{y}_i) = \mu_{t|-t}(\mathbf{x}_i, \mathbf{y}_i) = \mathbb{E}_{\tilde{y}_t \sim p(\cdot|\mathbf{y}_{i,-t})}[\tilde{y}_t]$. Conveniently, the conditional expectation has a closed form which is given by

$$\mu_{t|-t}(\mathbf{x}_i, \mathbf{y}_i) = m_t(\mathbf{x}_i) + (C^{(i)}_{-t,t})^\top (C^{(i)}_{-t,-t})^{-1}(\mathbf{y}_{i,-t} - \mathbf{m}_{i,-t}).$$

Here $C^{(i)}_{-t,-t} = [\mathcal{K}(\mathbf{x}_i, \mathbf{x}_i) + \Sigma]_{-t,-t}$ is the $T-1$ dimensional covariance matrix of $\mathbf{y}_{i,-t}$, $C^{(i)}_{-t,t} = [\mathcal{K}(\mathbf{x}_i, \mathbf{x}_i)]_{-t,t}$ is the $T-1$ dimensional covariance vector between $y_{i,t}$ and $\mathbf{y}_{i,-t}$, and $\mathbf{m}_{i,-t}$ collects the mean functions corresponding to $\mathbf{y}_{i,-t}$. Under the ICM, these expressions simplify since the covariance matrix of $\mathbf{y}_i$ additively decomposes as $C^{(i)} = B + \Sigma$. The closed-form conditional expectation, however, also holds for more general multi-output kernels; full details are provided in Appendix C.3.

Having fixed γ_t, we use it to adaptively scale $c_t(\mathbf{x}, \mathbf{y})$ by choosing it as the $(1 - \epsilon_t)$-th quantile of $\{|y_{i,t} - \gamma_t(\mathbf{x}_i, \mathbf{y}_i)|\}_{i=1}^N$, where $\epsilon_t \in (0, 1)$ denotes the expected proportion of outliers in output

channel t. Finally, β_t allows us to incorporate application-specific information about channel t. For example, one might increase β_t when it is a more reliable channel, as is often the case in multi-fidelity modelling and causal inference (Dimitriou et al., 2024). As a default, we set $\beta_t = \sigma_t/\sqrt{2}$ to recover the standard MOGP in the absence of outliers.

3.2 Robustness to Outliers

With weights as chosen in the previous section, we analyse the outlier robustness of R-MOGPs (Huber, 1981). To this end, for some observation with index $m \in \{1, \ldots, N\}$ and output $s \in \{1, \ldots, T\}$ in $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$, we replace $y_{m,s}$ with an arbitrarily large contaminated value $y^c_{m,s}$. This results in the contaminated dataset $\mathcal{D}^c_{m,s} = (\mathcal{D} \setminus \{(\mathbf{x}_m, \mathbf{y}_m)\}) \cup \{(\mathbf{x}_m, \mathbf{y}^c_{m,s})\}$, where $\mathbf{y}^c_{m,s} = [y_{m,1}, \ldots, y^c_{m,s}, \ldots, y_{m,T}]^\top$. We quantify the effect of this contamination on the marginal posterior over output $t \in \{1, \ldots, T\}$ by measuring the divergence between the contaminated and uncontaminated posteriors. As a function of $|y^c_{m,s} - y_{m,s}|$, this divergence is referred to as the *posterior influence function* (PIF; Ghosh and Basu, 2016; Matsubara et al., 2022; Altamirano et al., 2023). For the special case of Gaussian posteriors, several works advocate using the Kullback–Leibler (KL) divergence (Altamirano et al., 2024; Duran-Martin et al., 2024; Laplante et al., 2025). With this in mind, we assess robustness to outliers by studying

$$\text{PIF}(y^c_{m,s}, f_t, \mathcal{D}) := \text{KL}\big(p_w(f_t \mid \mathcal{D}) \,\big\|\, p_w(f_t \mid \mathcal{D}^c_{m,s})\big).$$

As we show in our next result, the R-MOGP posterior is robust insofar as the effect of the contamination in output channel s on the marginal posterior over channel t is bounded, even for channels $t \neq s$, and even as the size of the contamination diverges so that $|y^c_{m,s} - y_{m,s}| \to \infty$. For standard MOGPs, as long as the covariance structure between output channels s and t is not diagonal, this behaviour is *not* replicable: instead, a single contaminated observation can now increase the PIF arbitrarily in any output t in the MOGP (see Appendix B.3). More generally, the MOGP can only provide the desired type of robustness against contaminations in output channel s in exchange for completely cutting the feedback between outputs s and t by imposing a block-diagonal matrix structure in which s and t fall into different blocks.

Proposition 3.2. *Suppose* $\mathbf{f} \sim \mathcal{GP}(\mathbf{m}, \mathcal{K})$, $\text{vec}(\mathcal{E}) \sim \mathcal{N}(0, \boldsymbol{\Sigma})$, *and let* w_t *be defined as in* (7), $W_i = \text{diag}(w_1(\mathbf{x}_i, \mathbf{y}_i), \ldots, w_T(\mathbf{x}_i, \mathbf{y}_i))$. *Then, there exists a constant* $C > 0$ *independent of* $y^c_{m,s}$ *for which*

$$\max_{1 \leq t \leq T} \sup_{y^c_{m,s} \in \mathbb{R}} \text{PIF}(y^c_{m,s}, f_t, \mathcal{D}) \leq C.$$

The proof is in Appendix B.3. In words, Proposition 3.2 says that the R-MOGP posterior is robust in *all* channels t against outliers of possibly infinite magnitude in any given channel s.

3.3 Parameter Optimisation

For R-MOGP to be practically useful, we need a way to optimise the noise variances Σ for each output and the kernel hyperparameters $\boldsymbol{\theta}$. In standard MOGPs, these are typically optimised through the marginal likelihood (Álvarez et al., 2012), but this is ill-posed for generalised Bayes posteriors since they do not correspond to conditional probability updates (Jewson and Rossell, 2022). In the context of RCGPs, Altamirano et al. (2024) therefore derived a closed form objective based on leave-one-out cross-validation (LOO-CV). Computing this objective scales as $\mathcal{O}(N^3T^3)$ (see Appendix B.2), which is the same computational cost as optimisation of the marginal likelihood in standard MOGPs (Sundararajan and Keerthi, 1999; Altamirano et al., 2024). Unfortunately, optimising this objective in the presence of outliers is problematic: it treats all observations equally—whether they are outliers or not—and thus tends to overfit to outliers. Indeed, RCGPs (Altamirano et al., 2024) exhibit poor predictive uncertainty calibration after hyperparameter optimisation. To this end, we propose to re-use the robustness-inducing weights within the hyperparameter optimisation routine via the weighted LOO-CV (w-LOO-CV) objective

$$\varphi_w(\Sigma, \boldsymbol{\theta}) = \sum_{i=1}^{N} \sum_{t=1}^{T} \left(\frac{w_{i,t}}{\beta_t}\right)^2 \log p_w(y_{i,t} | Y_{-(i,t)}, \Sigma, \boldsymbol{\theta}),$$

where $Y_{-(i,t)}$ denotes all elements in the output matrix Y except for the (i,t)-th element. We rescale the weights $w_{i,t}$ by $\beta_t = \sigma_t/\sqrt{2}$ to confine them to the unit interval, and to prevent weights with high variance from dominating the objective. In addition, we square the weights as we found it to lead to better numerical stability and improved performance. Implementation details are provided in Appendix C.1, and a further discussion on the proposed objective can be found in Appendix C.7.

3.4 Sparse Variational R-MOGP

A standard strategy to improve scalability of MOGPs with large N or T is to adopt a sparse variational approximation with Q inducing points, following the inducing point framework of Titsias (2009); Álvarez et al. (2010). We now show that the R-MOGP naturally admits such an approximation. To do so, we exploit the interpretation of MOGPs as linear combinations of R latent GPs (Álvarez et al., 2012) (see Appendix B.4). For each latent process we introduce Q inducing variables, yielding the *sparse variational R-MOGP* (R-SVMOGP).

Proposition 3.3. *Consider an ICM with R latent functions and coregionalisation matrix $B = AA^\top$, $A \in \mathbb{R}^{T\times R}$. Let $Z_r \in \mathbb{R}^{Q\times d}$ be inducing locations for latent function r and define inducing variables $U \in \mathbb{R}^{Q\times R}$. The R-SVMOGP predictive distribution is $p_W(f_\star|\mathbf{x}_\star, Y, X) = \mathcal{N}(f_\star; \boldsymbol{\mu}_\star^S, \boldsymbol{\Sigma}_\star^S)$, where*

$$\boldsymbol{\mu}_\star^S = \mathbf{m}_\star + \mathbf{k}_{U\star}^\top \mathbf{P}_U^{-1}\mathbf{K}_{FU}^\top \boldsymbol{\nu}$$
$$\boldsymbol{\Sigma}_\star^S = \mathbf{k}_{\star\star} - \mathbf{k}_{U\star}^\top \mathbf{K}_{UU}^{-1}\mathbf{k}_{U\star} + \mathbf{k}_{U\star}^\top \mathbf{P}_U^{-1}\mathbf{k}_{U\star},$$

where $\boldsymbol{\nu} = \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}(\mathbf{y}_{vec} - \mathbf{m}_{\mathbf{w},vec})$, $\mathbf{P}_U = \mathbf{K}_{UU} + \mathbf{K}_{FU}^\top \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}\mathbf{K}_{FU}$, $\mathbf{K}_{UU} = \operatorname{diag}(\kappa(Z_1,Z_1),\dots,\kappa(Z_R,Z_R)) \in \mathbb{R}^{QR\times QR}$ is a diagonal block matrix, and $\mathbf{K}_{FU} \in \mathbb{R}^{NT\times QR}$ is a block-matrix such that the t,r-th block of $\mathbf{K}_{FU}$ equals $a_{t,r}\kappa(X,Z_r)$ for $t \in \{1,\dots,T\}$ and $r \in \{1,\dots,R\}$. In addition, $\mathbf{k}_{U\star} \in \mathbb{R}^{QR\times T}$ is the block-matrix such that $[\mathbf{k}_{U\star}]_{r,t} = a_{t,r}\kappa(Z_r,\mathbf{x}_\star)$. The corresponding ELBO is given as:

$$\text{ELBO}(Z,\Sigma,\boldsymbol{\theta}) = -\frac{1}{2}\left(\operatorname{Tr}\left(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{w}}^{-1/2}(\mathbf{K} - \mathbf{K}_{FU}\mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^\top)\mathbf{J}_{\mathbf{w}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\right) + C(\Sigma,\boldsymbol{\theta})\right)$$
$$+ \frac{1}{2}\boldsymbol{\nu}^\top \mathbf{K}_{FU}\mathbf{P}_U^{-1}\mathbf{K}_{FU}^\top\boldsymbol{\nu} + \frac{1}{2}\log\left(\frac{\det(\mathbf{K}_{UU})}{\det(\mathbf{P}_U)}\right).$$

where $C(\Sigma,\boldsymbol{\theta})$ is a function that depends only on Σ and $\boldsymbol{\theta}$.

The proof is in Appendix B.4 and extends naturally to other multi-output kernels (Moreno-Muñoz et al., 2018; Nguyen and Bonilla, 2014). The resulting computational complexity reduces to $\mathcal{O}(NTRQ^2)$ (as compared to $\mathcal{O}(N^3T^3)$ for R-MOGPs), improving scalability when the number of inducing points Q is small enough.

4 Experiments

The code is available at `https://github.com/joshuarooijakkers/rmogp`. We focus on the ICM with a squared exponential kernel $\kappa(\mathbf{x},\mathbf{x}') = \exp(-\|\mathbf{x}-\mathbf{x}'\|_2^2/\ell^2)$ for some $\ell > 0$. We evaluate our experiments based on the root mean squared error (RMSE), as well as the negative log predictive density (NLPD). We start with a few synthetic examples illustrating the robustness of our method under different forms of misspecification, then move to applications to real-world data.

Synthetic Multi-task Imputation MOGPs are often used when some outputs are only partially observed across the domain, such as in causal inference (Dimitriou et al., 2024) and multi-fidelity modelling (Xi et al., 2018). For instance, missing signals from failing sensors can be reconstructed by exploiting correlations with signals from other sensors (Osborne et al., 2008). However, outliers can not only distort the MOGP's predictions within their own output, but also propagate these errors to correlated outputs where imputation is performed. Figure 2 illustrates this with a simulated ICM for $T = 2$ and a large correlation (≈ 0.88) among outputs. The MOGP's predictions are strongly affected by outliers and yield corrupted imputations, whilst the R-MOGP provides robustness and

more reliable imputation. In Appendix C.3, we further show that this behaviour also extends to non-ICM multi-output kernels.

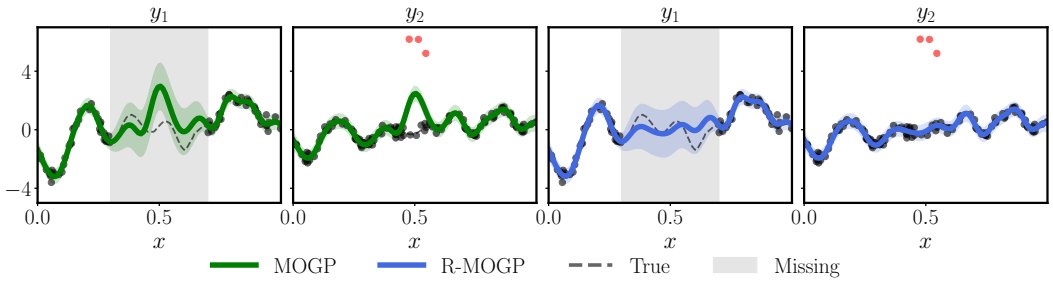


Figure 2: Synthetic imputation experiment with focused outliers. We consider a regression problem ($T = 2$) with structured missingness in y_1 and outliers in y_2. The MOGP predictives are sensitive to outliers, with errors propagating across outputs, whereas R-MOGP is robust in both outputs.

Synthetic Comparisons with Outliers We perform a similar experiment for $T = 3$ and 10% outliers of uniform distribution in the first output $t = 1$ (see Appendix C.5), with results in Table 1. As part of the comparison, we also consider an R-MOGP with the naive default RCGP weight w_{RCGP} of Altamirano et al. (2024) to show why careful choice of the weight function is crucial for the R-MOGP. Additionally, we compare against the performance on a dataset without outliers. The results show that in the well-specified regime, RMSE and NLPD are comparable across all methods, but that MOGPs experience a clear performance drop in the presence of outliers. While R-MOGP with the naive weight w_{RCGP} already improves robustness to outliers, an R-MOGP with the more appropriate weight w_{RMOGP} further outperforms it (see also Appendix C.8). In summary, R-MOGPs perform strongly under model misspecification, but are also competitive in the well-specified setting.

Table 1: Average RMSE and NLPD and standard deviation over 20 seeds under well-specified and outlier-contaminated settings.

Outliers	Method	RMSE	NLPD
✓	MOGP	0.20 ± 0.03	-0.37 ± 0.04
	R-MOGP (w_{RCGP})	0.18 ± 0.04	-0.44 ± 0.07
	R-MOGP (w_{RMOGP})	$\mathbf{0.11 \pm 0.01}$	$\mathbf{-0.67 \pm 0.08}$
✗	MOGP	0.093 ± 0.004	-0.94 ± 0.04
	R-MOGP (w_{RCGP})	0.093 ± 0.004	-0.94 ± 0.04
	R-MOGP (w_{RMOGP})	0.093 ± 0.004	-0.95 ± 0.04

Energy Efficiency Prediction Next, we compare our approach to existing state-of-the-art robust MOGPs; an MOGP with a Student-t (t-MOGP; Jylänki et al., 2011; Huang et al., 2024) and Laplace likelihood (L-MOGP; Kuss, 2006). As it addresses a different challenge, we do not compare against Yu et al. (2007): this method changed the *prior* over the latent function to be a Student-t process—rather than addressing misspecification in the *noise process* or *likelihood*. For our comparisons, we rely on the ENERGY EFFICIENCY dataset from the UCI repository, which relates energy efficiency of buildings to architectural characteristics. The goal is to predict $T = 2$ target variables—heating and cooling loads—using $d = 8$ covariates (e.g., surface area, overall height, relative compactness) and $N = 576$ training points. Performances across four different types of outlier scenarios are reported in Table 2 and runtimes per

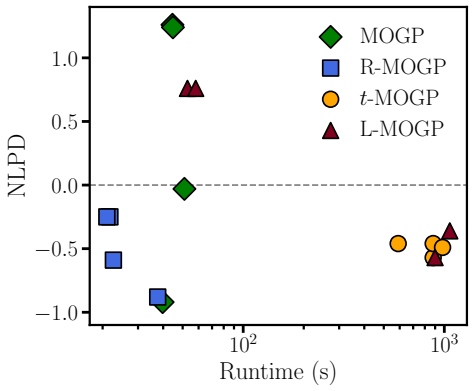


Figure 3: Trade-off between average total runtime and NLPD across several outlier scenarios and methods.

function evaluation are shown in Figure 4. Figure 3 additionally shows the trade-off between total runtime and NLPD for each method and scenario. Details of our computing setup are provided in Appendix C. In inferential terms, R-MOGP outperforms the MOGP across four different types of outlier scenarios, and remains competitive in the absence of outliers. Overall, R-MOGPs and t-MOGPs fare comparably well in the presence of outliers, substantially outperforming L-MOGP. In the absence of outliers, R-MOGP clearly outperforms both methods. Further, and as Figure 4 highlights, R-MOGP's closed form posteriors require a fraction of the computational cost of t-MOGPs and L-MOGPs—which require computationally intensive variational approximations. In Figure 3, we further find that R-MOGP achieves total runtimes that are orders of magnitude lower than those of the variational methods, as it also requires fewer iterations to converge. We found that L-MOGP converges early under both uniform and asymmetric outlier settings, at the expense of poorer predictive performance, which explains its lower total runtime. Lastly, as expected from the $\mathcal{O}(N^3T^3)$ complexity of both methods, computation times between R-MOGPs and MOGPs are of the same order of magnitude.

Table 2: Average and standard deviation of RMSE and NLPD on the ENERGY EFFICIENCY dataset under different outlier scenarios (10 seeds per scenario). Best results are highlighted in bold.

Outliers	Method	RMSE	NLPD
None	MOGP	$\mathbf{0.12 \pm 0.01}$	$\mathbf{-0.92 \pm 0.08}$
	R-MOGP (ours)	$\mathbf{0.12 \pm 0.01}$	$\mathbf{-0.88 \pm 0.07}$
	t-MOGP	0.14 ± 0.01	-0.57 ± 0.05
	L-MOGP	0.14 ± 0.01	-0.57 ± 0.05
Uniform	MOGP	2.27 ± 0.29	1.26 ± 0.06
	R-MOGP (ours)	0.17 ± 0.02	-0.25 ± 0.05
	t-MOGP	$\mathbf{0.14 \pm 0.01}$	$\mathbf{-0.46 \pm 0.03}$
	L-MOGP	0.31 ± 0.02	0.76 ± 0.02
Asymmetric	MOGP	2.18 ± 0.18	1.24 ± 0.06
	R-MOGP (ours)	0.16 ± 0.02	-0.31 ± 0.07
	t-MOGP	$\mathbf{0.14 \pm 0.01}$	$\mathbf{-0.46 \pm 0.02}$
	L-MOGP	0.30 ± 0.02	0.76 ± 0.01
Student-t	MOGP	0.19 ± 0.02	-0.03 ± 0.10
	R-MOGP (ours)	$\mathbf{0.14 \pm 0.01}$	$\mathbf{-0.59 \pm 0.04}$
	t-MOGP	$\mathbf{0.14 \pm 0.01}$	-0.49 ± 0.03
	L-MOGP	0.18 ± 0.06	-0.26 ± 0.24

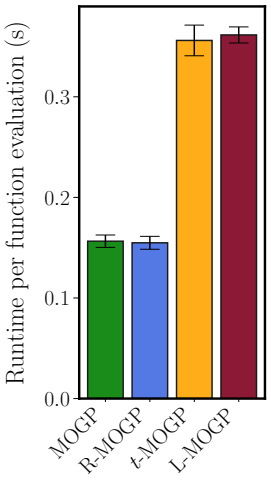


Figure 4: Average runtime per function evaluation on the ENERGY EFFICIENCY dataset across 40 runs.

Navitoclax Dose-Response Modelling Next, we apply our method to a setting where outliers arise naturally rather than synthetically: dose-response analysis in cancer research. In this context, MOGPs are commonly used to model the response of different types of cancer to various drugs (Wheeler, 2019; Moran et al., 2021; Rønneberg et al., 2021, 2023; Rønneberg and Lalchand, 2024; Giraldo Gutierrez et al., 2024). Each output function in an MOGP corresponds to cancer cells isolated from different patients, and the latent functions describe how cell *viability*—the proportion of cells that remain alive since the start of treatment—changes as drug dose increases (Giraldo Gutierrez et al., 2024; Rønneberg et al., 2023). Modelling this latent function helps pharmaceutical and medical practitioners to identify effective treatments: a steep decline in viability with increasing dose suggests that the drug has a strong effect, whereas a flat response suggests little to no effect. Importantly, summary statistics derived from these curves (most notably area under the curve, or half maximal inhibitory concentration IC_{50}) are explicitly linked to chemical properties of the drugs and genetic markers that enable patient-specific precision medicine. Yet, outliers and noisy measurements frequently arise in this setting, and stem from a mixture of biological and technical sources such as heterogeneity in cell populations, batch effects or pipetting errors (Malo et al., 2006). The resulting contaminations can severely distort MOGPs and their summary statistics (Wang et al., 2020), which can lead to worse decisions in clinical and drug discovery applications.

In Figure 1, the data point marked in red indicates such a naturally occurring outlier: for unknown reasons, the viability is significantly inflated, even though similar doses yield viability values close to zero. The figure is the result of a drug response analysis of the experimental drug *Navitoclax* based on data from the Genomics of Drug Sensitivity in Cancer Database (Yang et al., 2013). We compare the fit produced by a standard MOGP with our proposed approach on a subset of $T = 2$ cell lines with

$N = 21$ dose measurements given in $\log_{10}(\mu\text{M})$. For the prior mean, we use the constant function $m_t(\mathbf{x}) = 0.5$ to neutrally balance our prior beliefs halfway between cells being alive (1.0) or dead (0.0). We observe that the MOGP's fit in the first output is visibly distorted by the outlier, with both a biased predictive mean and a large uncertainty. In contrast, the R-MOGP reduces the influence of the outlier: the produced curve fits the observed points better. Additionally, its smaller degree of uncertainty aligns with that of the second cell line which is unburdened by outliers.

We quantitatively assess these results by examining the downstream implications of the different predictive distributions. In dose-response modelling, the IC_{50}—defined as the $\log_{10}(\text{dosage})$ at which 50% of cells remain viable—is a key quantity for precision medicine applications (Giraldo Gutierrez et al., 2024). For y_1, the estimated IC_{50} values for the MOGP, R-MOGP, t-MOGP, and L-MOGP are -0.65, -0.82, -0.86, and -0.86, respectively, with corresponding 95% credible intervals of $[-0.75, -0.50]$, $[-0.88, -0.71]$, $[-0.91, -0.82]$, and $[-0.91, -0.80]$. The standard MOGP clearly yields both a shifted IC_{50} estimate and substantially wider uncertainty, whereas the robust models produce more concentrated and consistent estimates. As a result, the presence of a single outlier can propagate to downstream clinical quantities when not explicitly accounted for. We provide further quantitative analysis in Appendix C.11.

Electricity Load Forecasting Having established the performance of R-MOGP on relatively small-scale datasets, we now consider a larger-scale task to evaluate R-SVMOGP. Specifically, we study daily electricity load forecasting, a setting where MOGPs are widely used in practice (Ghasempour and Martínez-Ramón, 2023a,b). We analyse one-dimensional time series ($d = 1$) of country-level aggregated electricity loads for $T = 5$ Northern European countries from January 2010 to September 2020. The ELECTRICITY LOAD dataset is obtained from Open Power System Data (Wiese et al., 2019). The training set contains $N = 3{,}415$ observations per country, resulting in a total of $NT = 17{,}075$ observations. We compare R-SVMOGP against standard SVMOGP and the sparse variational approaches of t-MOGP (t-SVMOGP) and L-MOGP (L-SVMOGP). To assess robustness, we contaminate one country with $\epsilon_1 = 0.05$ focused outliers that mimic the effect of a rare national political event (see Appendix C.12). Figure 5 reports runtime and test performance as the training set size increases, using $Q = \sqrt{N}$ inducing points per output. Although R-SVMOGP, t-SVMOGP, and L-SVMOGP all achieve significant improvements in accuracy, only R-SVMOGP retains the same computational scaling as standard SVMOGP. In contrast, t-SVMOGP and L-SVMOGP scale less favourably because their non-conjugate formulations require numerical integration to evaluate the ELBO.

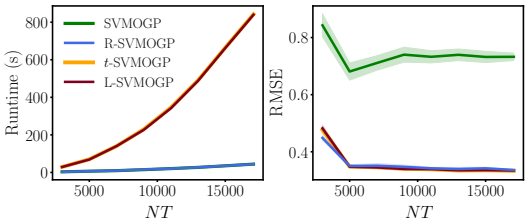


Figure 5: Average runtime in seconds (left) and RMSE (right) across 10 runs on ELECTRICITY LOAD forecasting.

5 Conclusion

MOGPs provide a probabilistic framework for modelling multiple correlated response variables by sharing information across outputs, but are sensitive to outliers and model misspecification if modelled with Gaussian errors. While several strategies exist in the literature to address this lack of robustness, they do so at the expense of exact conjugate updates. To overcome this limitation, we propose R-MOGP: an MOGP that jointly models correlated response variables, retains analytical tractability, and is provably robust to outliers of arbitrarily large magnitude in any individual output. We further resolve the challenges that arise from a naive extension of RCGPs to the multi-output setting by borrowing statistical strength across outputs and introducing a novel hyperparameter optimisation procedure. Empirically, we show that R-MOGPs provide performances comparable to existing robust MOGPs at a fraction of the latter's computational cost. While we explored sparse outliers, R-MOGPs can be tuned for various other types of outliers by adjusting their weight function. For example, they could be adapted to the case where entire output channels are treated as outliers, or where outliers occur across all channels at the same point. Lastly, R-MOGPs could be used in various applications of MOGPs, including Bayesian optimisation (Swersky et al., 2013), multi-fidelity modelling (Kennedy and O'Hagan, 2000), and active learning frameworks (Li et al., 2022).

Acknowledgments and Disclosure of Funding

We thank Dick van Dijk and William Laplante for valuable discussions on early drafts of this paper. JR was supported by the Dutch Cultural Fund Grant (Cultuurfondsbeurs). LR was supported by the European Union's Horizon Europe research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 101126636, and Integreat — Norwegian Centre for knowledge-driven machine learning, Research Council of Norway project number 332645. FXB and JK were supported by the EPSRC grant [EP/Y011805/1]. FXB was further supported by the EPSRC grant [EP/Y022300/1]. MA was supported by the Bloomberg Data Science PhD fellowship.

References

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. TensorFlow: A system for large-scale machine learning. In *USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2016.

Ahmed M Alaa and Mihaela Van Der Schaar. Bayesian inference of individualized treatment effects using multi-task Gaussian processes. *Advances in Neural Information Processing Systems*, 2017.

Matias Altamirano, François-Xavier Briol, and Jeremias Knoblauch. Robust and scalable Bayesian online changepoint detection. In *International Conference on Machine Learning*, 2023.

Matias Altamirano, François-Xavier Briol, and Jeremias Knoblauch. Robust and conjugate Gaussian process regression. In *International Conference on Machine Learning*, 2024.

Mauricio Álvarez, David Luengo, Michalis Titsias, and Neil D Lawrence. Efficient multioutput Gaussian processes through variational inducing kernels. In *International Conference on Artificial Intelligence and Statistics*, 2010.

Mauricio A. Álvarez, Lorenzo Rosasco, and Neil D. Lawrence. Kernels for vector-valued functions: A review. *Foundations and Trends in Machine Learning*, 4(3):195–266, 2012.

Mauricio A Alvarez, David Luengo, and Neil D Lawrence. Linear latent force models using Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2693–2705, 2013.

Sebastian Ament, Elizabeth Santorella, David Eriksson, Ben Letham, Maximilian Balandat, and Eytan Bakshy. Robust Gaussian processes via relevance pursuit. *Advances in Neural Information Processing Systems*, 2024.

Alessandro Barp, Francois-Xavier Briol, Andrew Duncan, Mark Girolami, and Lester Mackey. Minimum Stein discrepancy estimators. *Advances in Neural Information Processing Systems*, 2019.

Pier Giovanni Bissiri, Chris C Holmes, and Stephen G Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78 (5):1103–1130, 2016.

Edwin V Bonilla, Kian Ming A. Chai, and Christopher K. I. Williams. Multi-task Gaussian process prediction. *Advances in Neural Information Processing Systems*, 2007.

Ayman Boustati, Omer Deniz Akyildiz, Theodoros Damoulas, and Adam Johansen. Generalised Bayesian filtering via sequential Monte Carlo. *Advances in Neural Information Processing Systems*, 2020.

Nina Boyarchenko, Andreas Fuster, and David O Lucca. Understanding mortgage spreads. *The Review of Financial Studies*, 32(10):3799–3850, 2019.

Richard H Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, 1991.

Taco de Wolff, Alejandro Cuevas, and Felipe Tobar. Gaussian process imputation of multiple financial series. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

Charita Dellaporta, Jeremias Knoblauch, Theodoros Damoulas, and François-Xavier Briol. Robust Bayesian inference for simulator-based models via the MMD posterior bootstrap. In *International Conference on Artificial Intelligence and Statistics*, 2022.

Evangelos Dimitriou, Edwin Fong, Karla Diaz-Ordaz, and Brieuc Lehmann. Data fusion for heterogeneous treatment effect estimation with multi-task Gaussian processes. *arXiv preprint arXiv:2405.20957*, 2024.

Gerardo Duran-Martin, Matias Altamirano, Alexander Y Shestopaloff, Leandro Sánchez-Betancourt, Jeremias Knoblauch, Matt Jones, François-Xavier Briol, and Kevin Murphy. Outlier-robust Kalman filtering through generalised Bayes. In *International Conference on Machine Learning*, 2024.

Abdelhamid Ezzerg, Ilija Bogunovic, and Jeremias Knoblauch. Robust Bayesian optimisation with unbounded corruptions. In *International Conference on Machine Learning*, 2026.

Edwin Fong, Chris Holmes, and Stephen G Walker. Martingale posterior distributions. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 85(5):1357–1391, 2023.

Futoshi Futami, Issei Sato, and Masashi Sugiyama. Variational inference based on robust divergences. In *International Conference on Artificial Intelligence and Statistics*, 2018.

Daniele Gammelli, Kasper Pryds Rolsted, Dario Pacino, and Filipe Rodrigues. Generalized multi-output Gaussian process censored regression. *Pattern Recognition*, 129:108751, 2022.

Roman Garnett. *Bayesian Optimization*. Cambridge University Press, 2023.

Marc G. Genton and William Kleiber. Cross-covariance functions for multivariate geostatistics. *Statistical Science*, 30(2):147–163, 2015.

Alireza Ghasempour and Manel Martínez-Ramón. Electric load forecasting using multiple output Gaussian processes and multiple kernel learning. In *IEEE Symposium on Industrial Electronics & Applications (ISIEA)*, 2023a.

Alireza Ghasempour and Manel Martínez-Ramón. Short-term electric load prediction in smart grid using multi-output Gaussian processes regression. In *IEEE Kansas Power and Energy Conference (KPEC)*, 2023b.

Abhik Ghosh and Ayanendranath Basu. Robust Bayes estimation using the density power divergence. *Annals of the Institute of Statistical Mathematics*, 68:413–437, 2016.

Juan-José Giraldo Gutierrez, Evelyn Lau, Subhashini Dharmapalan, Melody Parker, Yurui Chen, Mauricio A Álvarez, and Dennis Wang. Multi-output prediction of dose–response curves enables drug repositioning and biomarker discovery. *npj Precision Oncology*, 8(1):209, 2024.

Gene H Golub and Charles F Van Loan. *Matrix Computations*. Johns Hopkins University Press, 2013.

Joan Gonzalvez, Edmond Lezmi, Thierry Roncalli, and Jiali Xu. Financial applications of Gaussian processes and Bayesian optimization. *arXiv preprint arXiv:1903.04841*, 2019.

Pierre Goovaerts. *Geostatistics for Natural Resources Evaluation*. Oxford University Press, 1997.

Peter Grünwald. The safe Bayesian: learning the learning rate via the mixability gap. In *International Conference on Algorithmic Learning Theory*, 2012.

Frank R. Hampel, Elvezio M. Ronchetti, Peter J. Rousseeuw, and Werner A. Stahel. *Robust Statistics: The Approach Based on Influence Functions*. John Wiley & Sons, 1986.

Haojie Huang, Xin Peng, Wei Du, and Weimin Zhong. Robust sparse Gaussian process regression for soft sensing in industrial big data under the outlier condition. *IEEE Transactions on Instrumentation and Measurement*, 73:1–11, 2024.

Peter J Huber. *Robust Statistics*. John Wiley & Sons, 1981.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005.

Jack Jewson and David Rossell. General Bayesian loss function selection and the use of improper models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 84(5): 1640–1665, 2022.

A. G. Journel and C. J. Huijbregts. *Mining Geostatistics*. Academic Press, 1978.

Pasi Jylänki, Jarno Vanhatalo, and Aki Vehtari. Robust Gaussian process regression with a Student-t likelihood. *Journal of Machine Learning Research*, 12(99):3227–3257, 2011.

Marc C Kennedy and Anthony O'Hagan. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87(1):1–13, 2000.

Jeremias Knoblauch, Jack Jewson, and Theodoros Damoulas. Doubly robust Bayesian inference for non-stationary streaming data with β-divergences. *Advances in Neural Information Processing Systems*, 2018.

Jeremias Knoblauch, Jack Jewson, and Theodoros Damoulas. An optimization-centric view on Bayes' rule: Reviewing and generalizing variational inference. *Journal of Machine Learning Research*, 23 (132):1–109, 2022.

Malte Kuss. *Gaussian process models for robust regression, classification, and reinforcement learning*. PhD thesis, Technische Universität Darmstadt, 2006.

William Laplante, Matias Altamirano, Andrew Duncan, Jeremias Knoblauch, and François-Xavier Briol. Robust and conjugate spatio-temporal Gaussian processes. In *International Conference on Machine Learning*, 2025.

Taehee Lee and Jun S Liu. Multi-response heteroscedastic Gaussian process models and their inference. *arXiv preprint arXiv:2308.15370*, 2023.

Christophe Leys, Olivier Klein, Yves Dominicy, and Christophe Ley. Detecting multivariate outliers: Use a robust variant of the Mahalanobis distance. *Journal of Experimental Social Psychology*, 74: 150–156, 2018.

Cen-You Li, Barbara Rakitsch, and Christoph Zimmer. Safe active learning for multi-output Gaussian processes. In *International Conference on Artificial Intelligence and Statistics*, 2022.

Howard Lightfoot, Dieudonne van der Meer, and Daniel J. Vis. *gdscIC50: Pipeline for GDSC Curve Fitting*. Genome Research Limited and The Netherlands Cancer Institute (NKI), 2023. R package version 0.99.4.

Song Liu, Takafumi Kanamori, and Daniel J Williams. Estimating density models with truncation boundaries using score matching. *Journal of Machine Learning Research*, 23(186):1–38, 2022.

Xing Liu and François-Xavier Briol. On the robustness of kernel goodness-of-fit tests. *arXiv preprint arXiv:2408.05854*, 2024.

Yifan Lu, Jiayi Ma, Leyuan Fang, Xin Tian, and Junjun Jiang. Robust and scalable Gaussian process regression and its applications. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Wesley J Maddox, Maximilian Balandat, Andrew G Wilson, and Eytan Bakshy. Bayesian optimization with high-dimensional outputs. *Advances in Neural Information Processing Systems*, 2021.

Nathalie Malo, James A Hanley, Sonia Cerquozzi, Jerry Pelletier, and Robert Nadon. Statistical practice in high-throughput screening data analysis. *Nature Biotechnology*, 24(2):167–175, 2006.

Kanti V Mardia, John T Kent, and Charles C Taylor. *Multivariate Analysis*. John Wiley & Sons, 2024.

Takuo Matsubara, Jeremias Knoblauch, François-Xavier Briol, and Chris J Oates. Robust generalised Bayesian inference for intractable likelihoods. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 84(3):997–1022, 2022.

Takuo Matsubara, Jeremias Knoblauch, François-Xavier Briol, and Chris J Oates. Generalized Bayesian inference for discrete intractable likelihood. *Journal of the American Statistical Association*, 119(547):2345–2355, 2024.

Alexander G de G Matthews, Mark Van Der Wilk, Tom Nickson, Keisuke Fujii, Alexis Boukouvalas, Pablo León-Villagrá, Zoubin Ghahramani, and James Hensman. GPflow: A Gaussian process library using TensorFlow. *Journal of Machine Learning Research*, 18(40):1–6, 2017.

Yann McLatchie, Badr-Eddine Cherief-Abdellatif, David T Frazier, and Jeremias Knoblauch. Predictively oriented posteriors. *arXiv preprint arXiv:2510.01915*, 2025a.

Yann McLatchie, Edwin Fong, David T. Frazier, and Jeremias Knoblauch. Predictive performance of power posteriors. *Biometrika*, 112(3):asaf034, 2025b.

Charles A. Micchelli and Massimiliano Pontil. Kernels for multi-task learning. *Advances in Neural Information Processing Systems*, 2004.

Jeffrey W Miller and David B Dunson. Robust Bayesian inference via coarsening. *Journal of the American Statistical Association*, 114(527):1113–1125, 2019.

Kelly R. Moran, David Dunson, Matthew W. Wheeler, and Amy H. Herring. Bayesian joint modeling of chemical structure and dose response curves. *The Annals of Applied Statistics*, 15(3):1405–1430, 2021.

Pablo Moreno-Muñoz, Antonio Artés, and Mauricio Alvarez. Heterogeneous multi-output Gaussian process prediction. *Advances in Neural Information Processing Systems*, 2018.

Trung V. Nguyen and Edwin V. Bonilla. Collaborative multi-output Gaussian processes. In *Conference on Uncertainty in Artificial Intelligence*, 2014.

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

Michael A Osborne, Stephen J Roberts, Alex Rogers, Sarvapali D Ramchurn, and Nicholas R Jennings. Towards real-time information processing of sensor network data using computationally efficient multi-output Gaussian processes. In *International Conference on Information Processing in Sensor Networks*, 2008.

Gabriel Parra and Felipe Tobar. Spectral mixture kernels for multi-output Gaussian processes. *Advances in Neural Information Processing Systems*, 2017.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Leiv Rønneberg and Vidhi Lalchand. Scalable permutation invariant multi-output Gaussian processes for cancer drug response. In *NeurIPS 2024 Workshop on Bayesian Decision-making and Uncertainty*, 2024.

Leiv Rønneberg, Andrea Cremaschi, Robert Hanes, Jorrit M Enserink, and Manuela Zucknick. Bayesynergy: flexible Bayesian modelling of synergistic interaction effects in in vitro drug combination experiments. *Briefings in Bioinformatics*, 22(6), 2021.

Leiv Rønneberg, Paul DW Kirk, and Manuela Zucknick. Dose–response prediction for in-vitro drug combination datasets: a probabilistic approach. *BMC Bioinformatics*, 24(1):161, 2023.

Peter J Rousseeuw. Least median of squares regression. *Journal of the American Statistical Association*, 79(388):871–880, 1984.

Peter J Rousseeuw and Mia Hubert. Robust statistics for outlier detection. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(1):73–79, 2011.

Peter J Rousseeuw and Katrien Van Driessen. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41(3):212–223, 1999.

Zheyang Shen, Jeremias Knoblauch, Samuel Power, and Chris J Oates. Prediction-centric uncertainty quantification via MMD. In *International Conference on Artificial Intelligence and Statistics*, 2025.

Marshal Sinaga, Julien Martinelli, and Samuel Kaski. Robust and computation-aware Gaussian processes. *Advances in Neural Information Processing Systems*, 2025.

Nikola Štefelová, Andreas Alfons, Javier Palarea-Albaladejo, Peter Filzmoser, and Karel Hron. Robust regression with compositional covariates including cellwise outliers. *Advances in Data Analysis and Classification*, 15(4):869–909, 2021.

Michael L. Stein. *Interpolation of Spatial Data: Some Theory for Kriging*. Springer, 1999.

Peng Su, Garth Tarr, and Samuel Muller. Robust variable selection under cellwise contamination. *Journal of Statistical Computation and Simulation*, 94(6):1371–1387, 2024.

Zhuo Sun, Alessandro Barp, and François-Xavier Briol. Vector-valued control variates. In *International Conference on Machine Learning*, 2023.

Sellamanickam Sundararajan and Sathiya Keerthi. Predictive approaches for choosing hyperparameters in Gaussian processes. *Advances in Neural Information Processing Systems*, 1999.

Kevin Swersky, Jasper Snoek, and Ryan P Adams. Multi-task Bayesian optimization. *Advances in Neural Information Processing Systems*, 2013.

Michalis Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *International Conference on Artificial Intelligence and Statistics*, 2009.

James I Vickery and Joshua Wright. TBA trading and liquidity in the agency MBS market. *Economic Policy Review*, 19(1), 2013.

Daniel J Vis, Lorenzo Bombardelli, Howard Lightfoot, Francesco Iorio, Mathew J Garnett, and Lodewyk FA Wessels. Multilevel models improve precision and speed of IC50 estimates. *Pharmacogenomics*, 17(7):691–700, 2016.

Dennis Wang, James Hensman, Ginte Kutkaite, Tzen S Toh, Ana Galhoz, GDSC Screening Team, Jonathan R Dry, Julio Saez-Rodriguez, Mathew J Garnett, Michael P Menden, and Frank Dondelinger. A statistical framework for assessing pharmacological responses and biomarkers using uncertainty estimates. *eLife*, 9:e60352, 2020.

Matthew W. Wheeler. Bayesian additive adaptive basis tensor product models for modeling high dimensional surfaces: An application to high-throughput toxicity testing. *Biometrics*, 75(1): 193–201, 2019.

Frauke Wiese, Ingmar Schlecht, Wolf-Dieter Bunke, Clemens Gerbaulet, Lion Hirth, Martin Jahn, Friedrich Kunz, Casimir Lorenz, Jonathan Mühlenpfordt, Juliane Reimann, et al. Open Power System Data–frictionless data for electricity system modelling. *Applied Energy*, 236:401–409, 2019.

Veit David Wild, Sahra Ghalebikesabi, Dino Sejdinovic, and Jeremias Knoblauch. A rigorous link between deep ensembles and (variational) Bayesian methods. *Advances in Neural Information Processing Systems*, 2023.

Pei-Shien Wu and Ryan Martin. A comparison of learning rate selection methods in generalized Bayesian inference. *Bayesian Analysis*, 18(1):105–132, 2023.

George Wynne, François-Xavier Briol, and Mark Girolami. Convergence guarantees for Gaussian process means with misspecified likelihoods and smoothness. *Journal of Machine Learning Research*, 22(123):1–40, 2021.

Xiaoyue Xi, François-Xavier Briol, and Mark Girolami. Bayesian quadrature for multiple related integrals. In *International Conference on Machine Learning*, 2018.

Jiazhen Xu, Janice L Scealy, Andrew TA Wood, and Tao Zou. Generalized score matching for regression. *arXiv preprint arXiv:2203.09864*, 2022.

Wanjuan Yang, Jorge Soares, Patricia Greninger, Elena J Edelman, Howard Lightfoot, Simon Forbes, Nidhi Bindal, Dave Beare, James A Smith, I Richard Thompson, et al. Genomics of Drug Sensitivity in Cancer (GDSC): a resource for therapeutic biomarker discovery in cancer cells. *Nucleic Acids Research*, 41(D1):D955–D961, 2013.

Shipeng Yu, Volker Tresp, and Kai Yu. Robust multi-task learning with t-processes. In *International Conference on Machine Learning*, 2007.

Arnold Zellner. Optimal information processing and Bayes's theorem. *The American Statistician*, 42 (4):278–280, 1988.

Supplementary Materials

We introduce notation in Appendix A. Appendix B presents the proofs of our theoretical results. Finally, Appendix C contains details of our numerical experiments and additional results that complement the main text.

A Notation

In this section, we recap the notation used in this paper:

- Let $\nabla_y = [\partial/\partial y_1, \dots, \partial/\partial y_m]^\top$, and suppose $f : \mathbb{R}^m \to \mathbb{R}^m$ and $g : \mathbb{R}^m \to \mathbb{R}^{m\times m}$ are two sufficiently regular functions. The divergence operator is defined as:

$$(\nabla_y \cdot f)(\mathbf{y}) = \sum_{i=1}^{m} \frac{\partial f_i(\mathbf{y})}{\partial y_i}, \qquad (\nabla_y \cdot g)_j(\mathbf{y}) = \sum_{i=1}^{m} \frac{\partial g_{ij}(\mathbf{y})}{\partial y_i}, \quad \forall j \in \{1, \dots, m\},$$

  where f_i and g_{ij} denote the i-th component of f and the (i, j)-th component of g, respectively.
- We denote by $\tilde{\mathbf{y}} \sim \mathcal{N}(\mu, \Sigma)$ that $\tilde{\mathbf{y}} \in \mathbb{R}^m$ follows an m-variate Gaussian distribution with mean $\mu \in \mathbb{R}^m$ and covariance matrix $\Sigma \in \mathbb{R}^{m\times m}$. Furthermore, $p(\tilde{\mathbf{y}}) = \mathcal{N}(\tilde{\mathbf{y}}; \mu, \Sigma)$ is the corresponding probability density function.
- Let $\mathbf{v} = [v_1, \dots, v_n]^\top \in \mathbb{R}^n$, and $\mathbf{M}_1, \dots, \mathbf{M}_m \in \mathbb{R}^{n\times n}$. The diagonal operator is defined as:

$$\operatorname{diag}(\mathbf{v}) = \begin{bmatrix} v_1 & 0 & \cdots & 0 \\ 0 & v_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & v_n \end{bmatrix} \in \mathbb{R}^{n\times n},$$

$$\operatorname{diag}(\mathbf{M}_1, \dots, \mathbf{M}_m) = \begin{bmatrix} \mathbf{M}_1 & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{M}_m \end{bmatrix} \in \mathbb{R}^{mn\times mn}.$$

- Let $\mathbf{v} = [v_1, \dots, v_n]^\top \in \mathbb{R}^n$. Then, $\mathbf{v}^2 = [v_1^2, \dots, v_n^2]^\top$, and $\log(\mathbf{v}) = [\log(v_1), \dots, \log(v_n)]^\top$. Additionally, we abuse notation so that $\log(\operatorname{diag}(\mathbf{v})) = \operatorname{diag}(\log(\mathbf{v}))$.
- Let $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$, where a_{ij} is the (i, j)-th entry of A. The Kronecker product $A \otimes B$ is the block matrix defined by:

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix} \in \mathbb{R}^{mp\times nq}.$$

- Let $A \in \mathbb{R}^{m\times n}$, where a_{ij} is the (i, j)-th entry of A. We define the vectorisation of A column-wise as

$$\operatorname{vec}(A) = [a_{11}, \dots, a_{m1}, \cdots, a_{1n}, \dots, a_{mn}]^\top \in \mathbb{R}^{mn}$$

- Let $A \in \mathbb{R}^{m\times n}$ and $\mathbf{v} \in \mathbb{R}^m$. We define the indexing operator $[A]_{i,j} = a_{ij}$ and $[\mathbf{v}]_i = v_i$. Furthermore, for integer indices $i_1 \le i_2$ and $j_1 \le j_2$, we define the slicing operators as:

$$[A]_{i_1:i_2,\, j_1:j_2} = \begin{bmatrix} a_{i_1 j_1} & \cdots & a_{i_1 j_2} \\ \vdots & \ddots & \vdots \\ a_{i_2 j_1} & \cdots & a_{i_2 j_2} \end{bmatrix} \in \mathbb{R}^{(i_2-i_1+1)\times(j_2-j_1+1)},$$

$$[\mathbf{v}]_{i_1:i_2} = [v_{i_1}, \dots, v_{i_2}]^\top \in \mathbb{R}^{(i_2-i_1+1)}.$$

  That is, $[\cdot]_{i_1:i_2,\, j_1:j_2}$ extracts the submatrix of rows i_1 through i_2 and columns j_1 through j_2, while $[\cdot]_{i_1:i_2}$ extracts the corresponding subvector.

Finally, we highlight that by expanding the term $\nabla_y \cdot \log(\mathbf{W}^2)$ as part of $\mathbf{m}_{\mathbf{W},\text{vec}}$ in Proposition 3.1, we get

$$\begin{aligned}\nabla_y \cdot \log(\mathbf{W}^2) &= [\nabla_y \cdot \log(\mathbf{W}_1^2)^\top, \ldots, \nabla_y \cdot \log(\mathbf{W}_T^2)^\top]^\top \\ &= 2\Big[\frac{\partial \log(w_1(\mathbf{x}_1, \mathbf{y}_1))}{\partial y_{1,1}}, \ldots, \frac{\partial \log(w_1(\mathbf{x}_N, \mathbf{y}_N))}{\partial y_{N,1}}, \cdots, \\ &\qquad \frac{\partial \log(w_T(\mathbf{x}_1, \mathbf{y}_1))}{\partial y_{1,T}}, \ldots, \frac{\partial \log(w_T(\mathbf{x}_N, \mathbf{y}_N))}{\partial y_{N,T}}\Big]^\top \in \mathbb{R}^{NT}\end{aligned}$$

B Proofs of Theoretical Results

B.1 Proof of Proposition 3.1

Recall the weighted Fisher divergence introduced in Section 3:

$$D_W(\mathbf{f}) = \mathbb{E}_{\tilde{\mathbf{x}} \sim p_{0,x}}\Big[\mathbb{E}_{\tilde{\mathbf{y}} \sim p_0(\cdot|\tilde{\mathbf{x}})}\Big[\|W(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})^\top \big(s_{\text{model}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - s_{\text{data}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})\big)\|_2^2\Big]\Big], \tag{8}$$

where $s_{\text{model}} = \nabla_{\mathbf{y}} \log p_\theta(\mathbf{y}|\mathbf{x})$ and $s_{\text{data}} = \nabla_{\mathbf{y}} \log p_0(\mathbf{y}|\mathbf{x})$ are score functions of the posited model and the true data, respectively, and W is a matrix-valued weight function that is point-wise invertible and continuously differentiable. As we show in the following, this divergence induces conjugacy through a quadratic loss function and naturally leads to the derivation of our proposed R-MOGP posterior and predictive distributions.

Quadratic Loss Function To begin, we have the following quadratic loss function below.

Lemma B.1. *Under the weighted score matching divergence defined in Equation (8), and a Gaussian model $p_\theta(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}; \mathbf{f}(\mathbf{x}), \Sigma)$, the generalised Bayes loss function in Equation (2) takes a quadratic form in $\mathbf{f}$:*

$$\mathcal{L}_W(F, Y, X) = \frac{1}{N}\sum_{i=1}^{N}\Big[\mathbf{f}_i^\top A_i \mathbf{f}_i + \mathbf{f}_i^\top \mathbf{b}_i + c_i\Big], \tag{9}$$

where

$$A_i = \Sigma^{-1} W_i W_i^\top \Sigma^{-1}, \quad \mathbf{b}_i = -2(\Sigma^{-1} W_i W_i^\top \Sigma^{-1} \mathbf{y}_i - \nabla_{\mathbf{y}} \cdot W_i W_i^\top \Sigma^{-1}),$$

$$c_i = \mathbf{y}_i^\top \Sigma^{-1} W_i W_i^\top \Sigma^{-1} \mathbf{y}_i - \nabla_{\mathbf{y}} \cdot (W_i W_i^\top \Sigma^{-1} \mathbf{y}_i),$$

for $W_i := W(\mathbf{x}_i, \mathbf{y}_i)$ are terms that do not depend on $\mathbf{f}_i$. They do, however, depend on $\mathbf{x}_i$ and $\mathbf{y}_i$, either directly or via W_i.

Proof. Direct evaluation of (8) requires access to the true data score s_{data}, which is generally unknown. Fortunately, under mild smoothness and boundary conditions, integration by parts allows us to reformulate the expression up to an additive constant independent of $\mathbf{f}$, denoted by $\overset{+C}{=}$. This weighted version of score matching was first derived in Barp et al. (2019, Theorem 2), who refer to it as "diffusion score matching." Using this result, we obtain

$$\begin{aligned}D_W(\mathbf{f}) \overset{+C}{=} \mathbb{E}_{\tilde{\mathbf{x}} \sim p_{0,x}}\Big[\mathbb{E}_{\tilde{\mathbf{y}} \sim p_0(\cdot|\tilde{\mathbf{x}})}\Big[&\big\|W(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})^\top s_{\text{model}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})\big\|_2^2 \\ &+ 2\nabla_{\mathbf{y}} \cdot \{W(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) W(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})^\top s_{\text{model}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})\}\Big]\Big].\end{aligned}$$

Crucially, this term does not depend on s_{data} any more, and only features $p_{0,x}$ through the expectation. This leads to the natural estimator

$$\begin{aligned}\mathcal{L}_W(F, Y, X) = \frac{1}{N}\sum_{i=1}^{N}\Big[&\big\|W(\mathbf{x}_i, \mathbf{y}_i)^\top s_{\text{model}}(\mathbf{x}_i, \mathbf{y}_i)\big\|_2^2 \\ &+ 2\nabla_{\mathbf{y}} \cdot \Big(W(\mathbf{x}_i, \mathbf{y}_i) W(\mathbf{x}_i, \mathbf{y}_i)^\top s_{\text{model}}(\mathbf{x}_i, \mathbf{y}_i)\Big)\Big].\end{aligned}$$

For a Gaussian model $\varepsilon_i \sim \mathcal{N}(0, \Sigma)$, the score vector is given by

$$\begin{aligned} s_{\text{model}}(\mathbf{x}_i, \mathbf{y}_i) &= \nabla_{\mathbf{y}} \log \left(\frac{1}{(2\pi)^{T/2}|\Sigma|^{1/2}} \exp\Big\{ -\frac{1}{2}(\mathbf{y}_i - \mathbf{f}(\mathbf{x}_i))^\top \Sigma^{-1}(\mathbf{y}_i - \mathbf{f}(\mathbf{x}_i)\Big\}\right) \\ &= \nabla_{\mathbf{y}} \big(-\frac{1}{2}(\mathbf{y}_i - \mathbf{f}(\mathbf{x}_i))^\top \Sigma^{-1}(\mathbf{y}_i - \mathbf{f}(\mathbf{x}_i)\big) \\ &= -\frac{1}{2} \cdot 2\Sigma^{-1}(\mathbf{y}_i - \mathbf{f}(\mathbf{x}_i)) = \Sigma^{-1}(\mathbf{f}(\mathbf{x}_i) - \mathbf{y}_i) = \Sigma^{-1}(\mathbf{f}_i - \mathbf{y}_i), \end{aligned}$$

where we used that the gradient of a constant vanishes and the factor of 2 comes from differentiating the quadratic form. The loss can then be rewritten as

$$\mathcal{L}_W(F, Y, X) = \frac{1}{N}\sum_{i=1}^{N} \Bigg[\underbrace{\| W_i^\top \Sigma^{-1} (\mathbf{f}_i - \mathbf{y}_i) \|_2^2}_{(\star)} + 2 \underbrace{\nabla \cdot \left(W_i W_i^\top \Sigma^{-1} (\mathbf{f}_i - \mathbf{y}_i) \right)}_{(\star\star)} \Bigg].$$

We define $Q_i := W_i W_i^\top \Sigma^{-1}$. We can express $(\star)$ as

$$\begin{aligned} (\star) &= (\mathbf{f}_i - \mathbf{y}_i)^\top \Sigma^{-1} W_i W_i^\top \Sigma^{-1} (\mathbf{f}_i - \mathbf{y}_i) \\ &= \mathbf{f}_i^\top \Sigma^{-1} W_i W_i^\top \Sigma^{-1} \mathbf{f}_i - 2\mathbf{f}_i^\top \Sigma^{-1} W_i W_i^\top \Sigma^{-1} \mathbf{y}_i + \mathbf{y}_i^\top \Sigma^{-1} W_i W_i^\top \Sigma^{-1} \mathbf{y}_i \\ &= \mathbf{f}_i^\top \Sigma^{-1} Q_i \mathbf{f}_i - 2\mathbf{f}_i^\top \Sigma^{-1} Q_i \mathbf{y}_i + \mathbf{y}_i^\top \Sigma^{-1} Q_i \mathbf{y}_i. \end{aligned}$$

by expanding the quadratic form. Similarly, for $(\star\star)$

$$\begin{aligned} (\star\star) &= \nabla_{\mathbf{y}} \cdot \left(W_i W_i^\top \Sigma^{-1} \mathbf{f}_i \right) - \nabla_{\mathbf{y}} \cdot \left(W_i W_i^\top \Sigma^{-1} \mathbf{y}_i \right) \\ &= \nabla_{\mathbf{y}} \cdot (Q_i \mathbf{f}_i) - \nabla_{\mathbf{y}} \cdot (Q_i \mathbf{y}_i) \\ &= \mathbf{f}_i^\top \left(\nabla_{\mathbf{y}} \cdot Q_i \right) - \nabla_{\mathbf{y}} \cdot (Q_i \mathbf{y}_i) \,. \end{aligned}$$

where we used the product rule for the divergence operator. Substituting these expression in the loss, we get

$$\begin{aligned} &\mathcal{L}_W(F, Y, X) \\ &= \frac{1}{N}\sum_{i=1}^{N} \left[\mathbf{f}_i^\top \Sigma^{-1} Q_i \mathbf{f}_i - 2\mathbf{f}_i^\top \Sigma^{-1} Q_i \mathbf{y}_i + 2\mathbf{f}_i^\top \left(\nabla_{\mathbf{y}} \cdot Q_i \right) + \mathbf{y}_i^\top \Sigma^{-1} Q_i \mathbf{y}_i - \nabla_{\mathbf{y}} \cdot (Q_i \mathbf{y}_i) \right] \\ &= \frac{1}{N}\sum_{i=1}^{N} \left[\mathbf{f}_i^\top \Sigma^{-1} Q_i \mathbf{f}_i - 2\mathbf{f}_i^\top \left(\Sigma^{-1} Q_i \mathbf{y}_i - \nabla_{\mathbf{y}} \cdot Q_i \right) + \mathbf{y}_i^\top \Sigma^{-1} Q_i \mathbf{y}_i - \nabla_{\mathbf{y}} \cdot (Q_i \mathbf{y}_i) \right]. \end{aligned}$$

This loss function is quadratic in $\mathbf{f}_i$ where $A_i = \Sigma^{-1} Q_i$ and $\mathbf{b}_i = -2(\Sigma^{-1} Q_i \mathbf{y}_i - \nabla_{\mathbf{y}} \cdot Q_i)$, and $c_i = \mathbf{y}_i^\top \Sigma^{-1} Q_i \mathbf{y}_i - \nabla_{\mathbf{y}} \cdot (Q_i \mathbf{y}_i)$ as in Equation (9). □

We note that this result generalises Altamirano et al. (2024, Appendix A.1), who show that the loss function for scalar RCGPs is quadratic in $\mathbf{f}$ using the univariate weighted score-matching divergence. The corresponding RCGP loss function can be recovered as a special case by setting $T = 1$, $\mathbf{\Sigma} = \sigma^2$ and $W_i = w(\mathbf{x}_i, y_i) \in \mathbb{R}$.

Posterior Distribution We have the following closed-form solution for the posterior distribution.

Lemma B.2. *Let* $\text{vec}(\mathcal{E}) \sim \mathcal{N}(0, \mathbf{\Sigma})$ *where* $\mathbf{\Sigma} = \Sigma \otimes I_N$ *for a* T*-dimensional diagonal* Σ*, and* $W_i = \text{diag}(w_1(\mathbf{x}_i, \mathbf{y}_i), \ldots, w_T(\mathbf{x}_i, \mathbf{y}_i))$*, where* $y_{i,t} \mapsto w_t(\mathbf{x}_i, \mathbf{y}_i)$ *is continuously differentiable. Then, the R-MOGP posterior distribution over* $\mathbf{f}_{\text{vec}}$ *is given by*

$$\begin{aligned} p_W(\mathbf{f}_{\text{vec}} | \mathbf{y}_{\text{vec}}, \mathbf{x}_{\text{vec}}) &= \mathcal{N}(\mathbf{f}_{\text{vec}}; \boldsymbol{\mu}_{\text{vec}}^{\text{RMOGP}}, \mathbf{\Sigma}^{\text{RMOGP}}), \\ \boldsymbol{\mu}_{\text{vec}}^{\text{RMOGP}} &= \mathbf{m}_{\text{vec}} + \mathbf{K}(\mathbf{K} + \mathbf{\Sigma}\,\mathbf{J}_{\mathbf{W}})^{-1}(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{W},\text{vec}}), \\ \mathbf{\Sigma}^{\text{RMOGP}} &= \mathbf{K}(\mathbf{K} + \mathbf{\Sigma}\,\mathbf{J}_{\mathbf{W}})^{-1}\mathbf{\Sigma}\,\mathbf{J}_{\mathbf{W}}, \end{aligned}$$

where $\mathbf{J}_{\mathbf{W}} = \frac{1}{2}\boldsymbol{\Sigma}\mathbf{W}^{-2}$ *and* $\mathbf{m}_{\mathbf{W},\text{vec}} = \mathbf{m}_{\text{vec}} + \boldsymbol{\Sigma}\left(\nabla_{\mathbf{y}} \cdot \log(\mathbf{W}^2)\right)$, *where* $\mathbf{W} = \text{diag}(\mathbf{W}_1, \ldots, \mathbf{W}_T)$ *and* $\nabla_{\mathbf{y}} \cdot \log(\mathbf{W}^2) = [\nabla_{\mathbf{y}} \cdot \log(\mathbf{W}_1^2)^\top, \ldots, \nabla_{\mathbf{y}} \cdot \log(\mathbf{W}_T^2)^\top]^\top$.

Proof. The diagonal structure $W_i = \text{diag}(w_1(\mathbf{x}_i, \mathbf{y}_i), \ldots, w_T(\mathbf{x}_i, \mathbf{y}_i))$ implies that $\Sigma^{-1}Q_i = \Sigma^{-2}W_i^2$, since Σ is also diagonal. Additionally, we can write $\nabla_{\mathbf{y}} \cdot Q_i = \nabla_{\mathbf{y}} \cdot (W_i W_i^\top \Sigma^{-1}) = \Sigma^{-1}(\nabla_{\mathbf{y}} \cdot W_i^2)$ since Σ^{-1} is independent of $\mathbf{y}$. In addition, from $\nabla_{\mathbf{y}} \cdot \log(W_i^2) = W_i^{-2}\nabla_{\mathbf{y}} \cdot W_i^2$ it follows that $\nabla_{\mathbf{y}} \cdot W_i^2 = \Sigma^{-1}W_i^2\Sigma(\nabla_{\mathbf{y}} \cdot \log(W_i^2))$. Combining these expressions, we obtain $\nabla_{\mathbf{y}} \cdot Q_i = \Sigma^{-2}W_i^2\Sigma\nabla_{\mathbf{y}} \log(W_i^2)$. We can then simplify the loss function as

$$\begin{aligned}
\mathcal{L}_W(F, Y, X) &\overset{+C}{=} \frac{1}{N}\sum_{i=1}^{N}\left[\mathbf{f}_i^\top \Sigma^{-2}W_i^2\mathbf{f}_i - 2\mathbf{f}_i^\top\Sigma^{-2}W_i^2\Big(\mathbf{y}_i - \Sigma\nabla_{\mathbf{y}} \cdot \log(W_i^2)\Big)\right] \\
&= \frac{1}{2N}\left[2\mathbf{f}_{\text{vec}}^\top\boldsymbol{\Sigma}^{-2}\mathbf{W}^2\mathbf{f}_{\text{vec}} - 4\mathbf{f}_{\text{vec}}^\top\boldsymbol{\Sigma}^{-2}\mathbf{W}^2\Big(\mathbf{y}_{\text{vec}} - \boldsymbol{\Sigma}\nabla_{\mathbf{y}} \cdot \log(\mathbf{W}^2)\Big)\right] \\
&= \frac{1}{2N}\left[\mathbf{f}_{\text{vec}}^\top\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\mathbf{f}_{\text{vec}} - 2\mathbf{f}_{\text{vec}}^\top\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{W},\text{vec}} + \mathbf{m}_{\text{vec}})\right],
\end{aligned}$$

where $\mathbf{W} = \text{diag}(\mathbf{W}_1, \ldots, \mathbf{W}_T)$, $\mathbf{m}_{\mathbf{W},\text{vec}} = \mathbf{m}_{\text{vec}} + \boldsymbol{\Sigma}\nabla_{\mathbf{y}} \cdot \log(\mathbf{W}^2)$, and $\mathbf{J}_{\mathbf{W}} = \frac{1}{2}\boldsymbol{\Sigma}\mathbf{W}^{-2}$. In the second equality, since Σ and W_i are diagonal, all matrix products are interpreted as element-wise (Hadamard) multiplications. We can then derive the R-MOGP posterior by substituting the above loss function and the MOGP prior into the GB update, given in Equation (2):

$$\begin{aligned}
p_W(\mathbf{f}_{\text{vec}}|Y, X) &\propto p(\mathbf{f}_{\text{vec}}|X)\exp(-N\mathcal{L}_W(F, Y, X)) \\
&\propto \exp\left\{-\frac{1}{2}\left(\mathbf{f}_{\text{vec}} - \mathbf{m}_{\text{vec}}\right)^\top\mathbf{K}^{-1}\left(\mathbf{f}_{\text{vec}} - \mathbf{m}_{\text{vec}}\right)\right\} \\
&\quad \times \exp\left\{-\frac{1}{2}\Big(\mathbf{f}_{\text{vec}}^\top\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\mathbf{f}_{\text{vec}} - 2\mathbf{f}_{\text{vec}}^\top\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\left(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{W},\text{vec}} + \mathbf{m}_{\text{vec}}\right)\Big)\right\} \\
&= \exp\Bigg\{-\frac{1}{2}\Bigg[\left(\mathbf{f}_{\text{vec}} - \mathbf{m}_{\text{vec}}\right)^\top\mathbf{K}^{-1}\left(\mathbf{f}_{\text{vec}} - \mathbf{m}_{\text{vec}}\right) + \mathbf{f}_{\text{vec}}^\top\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\mathbf{f}_{\text{vec}} \\
&\qquad - 2\mathbf{f}_{\text{vec}}^\top\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\left(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{W},\text{vec}} + \mathbf{m}_{\text{vec}}\right)\Bigg]\Bigg\},
\end{aligned}$$

where in the second equality, the terms are combined under a single exponential function. We simplify the function by only considering the terms that depend on $\mathbf{f}_{\text{vec}}$:

$$\begin{aligned}
p_W(\mathbf{f}_{\text{vec}}|Y, X) &\propto \exp\Bigg\{-\frac{1}{2}\Bigg[\mathbf{f}_{\text{vec}}^\top\mathbf{K}^{-1}\mathbf{f}_{\text{vec}} - 2\mathbf{f}_{\text{vec}}^\top\mathbf{K}^{-1}\mathbf{m}_{\text{vec}} + \mathbf{f}_{\text{vec}}^\top\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\mathbf{f}_{\text{vec}} \\
&\qquad - 2\mathbf{f}_{\text{vec}}^\top\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\big(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{W},\text{vec}} + \mathbf{m}_{\text{vec}}\big)\Bigg]\Bigg\} \\
&= \exp\Bigg\{-\frac{1}{2}\Bigg[\mathbf{f}_{\text{vec}}^\top\big(\mathbf{K}^{-1} + \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\big)\mathbf{f}_{\text{vec}} - 2\mathbf{f}_{\text{vec}}^\top\mathbf{K}^{-1}\mathbf{m}_{\text{vec}} \\
&\qquad - 2\mathbf{f}_{\text{vec}}^\top\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\big(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{W},\text{vec}} + \mathbf{m}_{\text{vec}}\big)\Bigg]\Bigg\}
\end{aligned}$$

We can then factor out $\mathbf{m}_{\text{vec}}$ in the second term:

$$\begin{aligned}
&= \exp\Bigg\{-\frac{1}{2}\Bigg[\mathbf{f}_{\text{vec}}^\top(\mathbf{K}^{-1} + \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1})\mathbf{f}_{\text{vec}} \\
&\qquad - 2\mathbf{f}_{\text{vec}}^\top\Big(\mathbf{K}^{-1}\mathbf{m}_{\text{vec}} + \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\left(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{W},\text{vec}} + \mathbf{m}_{\text{vec}}\right)\Big)\Bigg]\Bigg\} \\
&= \exp\Bigg\{-\frac{1}{2}\Bigg[\mathbf{f}_{\text{vec}}^\top(\mathbf{K}^{-1} + \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1})\mathbf{f}_{\text{vec}} \\
&\qquad - 2\mathbf{f}_{\text{vec}}^\top\Big((\mathbf{K}^{-1} + \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1})\mathbf{m}_{\text{vec}} + \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\left(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{W},\text{vec}}\right)\Big)\Bigg]\Bigg\} \qquad (10)
\end{aligned}$$

By completing squares, the posterior follows an NT-variate Gaussian distribution:

$$
\begin{aligned}
p_W(\mathbf{f}_{\text{vec}}|Y,X) &\propto \exp\left\{-\frac{1}{2}(\mathbf{f}_{\text{vec}}-\boldsymbol{\mu}_{\text{vec}}^{\text{RMOGP}})^\top(\boldsymbol{\Sigma}^{\text{RMOGP}})^{-1}(\mathbf{f}_{\text{vec}}-\boldsymbol{\mu}_{\text{vec}}^{\text{RMOGP}})\right\},\\
\boldsymbol{\Sigma}^{\text{RMOGP}} &= (\mathbf{K}^{-1}+\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1})^{-1} = \mathbf{K}(\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_{\mathbf{w}})^{-1}\boldsymbol{\Sigma}\mathbf{J}_{\mathbf{w}}\\
\boldsymbol{\mu}_{\text{vec}}^{\text{RMOGP}} &= \boldsymbol{\Sigma}^{\text{RMOGP}}((\mathbf{K}^{-1}+\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1})\mathbf{m}_{\text{vec}}+\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}(\mathbf{y}_{\text{vec}}-\mathbf{m}_{\mathbf{w},\text{vec}}))\\
&= \mathbf{m}_{\text{vec}}+\mathbf{K}(\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_{\mathbf{w}})^{-1}(\mathbf{y}_{\text{vec}}-\mathbf{m}_{\mathbf{w},\text{vec}}),
\end{aligned}
$$

where we obtain $\boldsymbol{\Sigma}^{\text{RMOGP}}$ by using that for two invertible matrices A and B, it holds that $(A^{-1}+B^{-1})^{-1}=A(A+B)^{-1}B$. Additionally, we obtain $\boldsymbol{\mu}^{\text{RMOGP}}$ since the second term in Equation (10) equals $-2\mathbf{f}_{\text{vec}}(\boldsymbol{\Sigma}^{\text{RMOGP}})^{-1}\boldsymbol{\mu}^{\text{RMOGP}}$, and $\boldsymbol{\Sigma}$ and $\mathbf{J}_{\mathbf{w}}$ are both diagonal. Note that $\boldsymbol{\Sigma}^{\text{RMOGP}}$ is positive semidefinite, since it is the inverse of a sum of two positive semidefinite matrices. □

We again note that this posterior is a generalisation of the posterior predictive in Altamirano et al. (2024, Appendix A.1). We recover this posterior by selecting $T=1$, $\boldsymbol{\Sigma}=\sigma^2$, and $W_i=w(\mathbf{x}_i,y_i)$.

Predictive Distribution We now prove Proposition 3.1 which states that, under the posterior defined in Lemma B.2, the posterior predictive distribution over $\mathbf{f}_\star=\mathbf{f}(\mathbf{x}_\star)$ at $\mathbf{x}_\star$ is Gaussian:

$$
\begin{aligned}
p_W(\mathbf{f}_\star|\mathbf{x}_\star,Y,X) &= \mathcal{N}(\mathbf{f}_\star;\boldsymbol{\mu}_\star^{\text{RMOGP}},\boldsymbol{\Sigma}_\star^{\text{RMOGP}}),\\
\boldsymbol{\mu}_\star^{\text{RMOGP}} &= \mathbf{m}_\star+\mathbf{k}_\star^\top(\mathbf{K}+\boldsymbol{\Sigma}\,\mathbf{J}_{\mathbf{w}})^{-1}(\mathbf{y}_{\text{vec}}-\mathbf{m}_{\mathbf{W},\text{vec}}),\\
\boldsymbol{\Sigma}_\star^{\text{RMOGP}} &= \mathbf{k}_{\star\star}-\mathbf{k}_\star^\top(\mathbf{K}+\boldsymbol{\Sigma}\,\mathbf{J}_{\mathbf{w}})^{-1}\mathbf{k}_\star.
\end{aligned}
$$

Proof. We first derive the predictive for $\mathbf{m}_{\text{vec}}=0$ and then extend it to an arbitrary prior mean. We note that conditioning on the matrices F and Y is equivalent to conditioning on their vectorised forms, $\mathbf{f}_{\text{vec}}=\text{vec}(F)$ and $\mathbf{y}_{\text{vec}}=\text{vec}(Y)$, since they contain the same information. Given the posterior distribution over $\mathbf{f}_{\text{vec}}$, the conditional distribution over $\mathbf{f}_\star$ is given by

$$
\begin{aligned}
p(\mathbf{f}_\star|\mathbf{x}_\star,\mathbf{f}_{\text{vec}},\mathbf{y}_{\text{vec}},X) &= \mathcal{N}(\mathbf{f}_\star;\breve{\mu},\breve{\Sigma})\\
\breve{\mu} &= \mathbf{k}_\star^\top\mathbf{K}^{-1}\mathbf{f}_{\text{vec}} = a^\top\mathbf{f}_{\text{vec}}\\
\breve{\Sigma} &= \mathbf{k}_{\star\star}-\mathbf{k}_\star^\top\mathbf{K}^{-1}\mathbf{k}_\star,
\end{aligned}
$$

where $a=\mathbf{K}^{-1}\mathbf{k}_\star$ (Álvarez et al., 2012). Following Appendix A.1 of Altamirano et al. (2024), it can be shown that the density of the predictive distribution is also Gaussian, since it is obtained by integrating the product of two Gaussian densities:

$$
\begin{aligned}
p_W(\mathbf{f}_\star|\mathbf{x}_\star,\mathbf{y}_{\text{vec}},X) &= \int_{\mathbb{R}^{NT}} p(\mathbf{f}_\star|\mathbf{x}_\star,\mathbf{f}_{\text{vec}},\mathbf{y}_{\text{vec}},X)p_W(\mathbf{f}_{\text{vec}}|\mathbf{y}_{\text{vec}},X)d\mathbf{f}_{\text{vec}}\\
&= \mathcal{N}(\mathbf{f}_\star;\boldsymbol{\mu}_\star^{\text{RMOGP}},\boldsymbol{\Sigma}_\star^{\text{RMOGP}}),\\
\boldsymbol{\mu}_\star^{\text{RMOGP}} &= a^\top\boldsymbol{\mu}_{\text{vec}}^{\text{RMOGP}},\\
\boldsymbol{\Sigma}_\star^{\text{RMOGP}} &= \breve{\Sigma}+a^\top\boldsymbol{\Sigma}^{\text{RMOGP}}a.
\end{aligned}
$$

For more details on the derivation, we refer the reader to Altamirano et al. (2024). By substituting for $a=\mathbf{K}^{-1}\mathbf{k}_\star$, and $\boldsymbol{\mu}_{\text{vec}}^{\text{RMOGP}}$ given in Lemma B.2 in $\boldsymbol{\mu}_{\text{vec}}^{\text{RMOGP}}$, we obtain the desired expression for the R-MOGP predictive mean:

$$
\begin{aligned}
\boldsymbol{\mu}_\star^{\text{RMOGP}} &= \mathbf{k}_\star^\top\mathbf{K}^{-1}\mathbf{K}(\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_{\mathbf{w}})^{-1}(\mathbf{y}_{\text{vec}}-\mathbf{m}_{\mathbf{w},\text{vec}})\\
&= \mathbf{k}_\star^\top(\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_{\mathbf{w}})^{-1}(\mathbf{y}_{\text{vec}}-\mathbf{m}_{\mathbf{w},\text{vec}}).
\end{aligned}
$$

Similarly, we obtain the R-MOGP predictive variance by substituting for $\breve{\Sigma}=\mathbf{k}_{\star\star}-\mathbf{k}_\star^\top\mathbf{K}^{-1}\mathbf{k}_\star$, $a=\mathbf{K}^{-1}\mathbf{k}_\star$, and $\boldsymbol{\Sigma}^{\text{RMOGP}}$ given in Lemma B.2 in $\boldsymbol{\Sigma}_\star^{\text{RMOGP}}$:

$$
\begin{aligned}
\boldsymbol{\Sigma}_\star^{\text{RMOGP}} &= \mathbf{k}_{\star\star}-\mathbf{k}_\star^\top\mathbf{K}^{-1}\mathbf{k}_\star+\mathbf{k}_\star^\top\mathbf{K}^{-1}(\mathbf{K}^{-1}+\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1})^{-1}\mathbf{K}^{-1}\mathbf{k}_\star\\
&= \mathbf{k}_{\star\star}-\mathbf{k}_\star^\top\mathbf{K}^{-1}\mathbf{k}_\star+\mathbf{k}_\star^\top\mathbf{K}^{-1}(\mathbf{K}-\mathbf{K}(\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_{\mathbf{w}})^{-1}\mathbf{K})\mathbf{K}^{-1}\mathbf{k}_\star
\end{aligned}
$$

where in the second equality we use the Woodbury matrix identity (Golub and Van Loan, 2013, Chapter 2.1.3) which states that for invertible matrices A and B, where B is diagonal, it holds that $(A^{-1}+B^{-1})^{-1} = A - A(A+B)^{-1}A$. We can then expand the bracketed expressions and simplify:

$$\begin{aligned}\boldsymbol{\Sigma}_\star^{\text{RMOGP}} &= \mathbf{k}_{\star\star} - \mathbf{k}_\star^\top \mathbf{K}^{-1}\mathbf{k}_\star + \mathbf{k}_\star^\top \mathbf{K}^{-1}\mathbf{k}_\star - \mathbf{k}_\star^\top (\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w})^{-1}\mathbf{k}_\star \\ &= \mathbf{k}_{\star\star} - \mathbf{k}_\star^\top (\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w})^{-1}\mathbf{k}_\star,\end{aligned}$$

Now, if the prior mean function $\mathbf{m}_{\text{vec}}$ is non-zero, we replace $\mathbf{f}_{\text{vec}}$ by $\mathbf{f}_{\text{vec}} - \mathbf{m}_{\text{vec}}$, since for a function $\mathbf{f} \sim \mathcal{GP}(\mathbf{m}, \mathcal{K})$ it holds that $\mathbf{f} - \mathbf{m} \sim \mathcal{GP}(\mathbf{0}, \mathcal{K})$. Thus, we add back $\mathbf{m}_{\text{vec}}$ in the posterior, which results in the desired distribution. □

B.2 LOO-CV Predictive Distribution

We optimise the noise variance in each channel Σ, as well as the kernel hyperparameters $\boldsymbol{\theta}$ through the w-LOO-CV approach, as discussed in Section 3.3. Specifically, we optimise

$$\varphi_w(\Sigma, \boldsymbol{\theta}) = \sum_{i=1}^{N}\sum_{t=1}^{T}\left(\frac{w_{i,t}}{\beta_t}\right)^2 \log p_w(y_{i,t}|Y_{-(i,t)}, \Sigma, \boldsymbol{\theta}).$$

A naive implementation would involve fitting the model NT times, resulting in a computational complexity of $\mathcal{O}(N^4T^4)$. However, the LOO objective allows us to derive an analytical formulation for $p_w(y_{i,t}|Y_{-(i,t)}, \Sigma, \boldsymbol{\theta})$ which reduces the computational cost to $\mathcal{O}(N^3T^3)$ by only computing $(\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w})^{-1}$ once (for each unique set of parameters). This matches the computational complexity of evaluating the marginal likelihood (Altamirano et al., 2024; Sundararajan and Keerthi, 1999).

Lemma B.3. *Let* $\text{vec}(\mathcal{E}) \sim \mathcal{N}(0, \boldsymbol{\Sigma})$ *where* $\boldsymbol{\Sigma} = \Sigma \otimes I_N$ *for a* T*-dimensional diagonal* Σ*, and* $W_i = \text{diag}(w_1(\mathbf{x}_i, \mathbf{y}_i), \ldots, w_T(\mathbf{x}_i, \mathbf{y}_i))$*, where* $y_{i,t} \mapsto w_t(\mathbf{x}_i, \mathbf{y}_i)$ *is continuously differentiable. The closed-form LOO predictive likelihood for observation* $y_{i,t}$ *is given as:*

$$\begin{aligned}p_w(y_{i,t}|Y_{-(i,t)}, \boldsymbol{\theta}, \Sigma) &= \mathcal{N}(\mu_{i,t}^{\text{RMOGP}}, (\sigma_{i,t}^{\text{RMOGP}})^2 + \sigma_t^2) \\ \mu_{i,t}^{\text{RMOGP}} &= z_{i,t} + m_{i,t} - \frac{[(\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w})^{-1}\mathbf{z}_{\text{vec}}]_{(t-1)N+i}}{[(\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w})^{-1}]_{(t-1)N+i,(t-1)N+i}} \\ (\sigma_{i,t}^{\text{RMOGP}})^2 &= \frac{1}{[(\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w})^{-1}]_{(t-1)N+i,(t-1)N+i}} - \frac{\sigma_t^4}{2} w_t(\mathbf{x}_i, \mathbf{y}_i)^{-2},\end{aligned}$$

where $\mathbf{z}_{\text{vec}} = \mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{W},\text{vec}}$.

Proof. Without loss of generality, we derive the predictive for $i = N$ and $t = T$, and this can be extended for arbitrary $i \in \{1, \ldots, N\}$ and $t \in \{1, \ldots, T\}$ using a permutation matrix. Let $p_w(y_{N,T}) = \mathcal{N}(\mu_{N,T}^{\text{RMOGP}}, (\sigma_{N,T}^{\text{RMOGP}})^2 + \sigma_T^2)$ be the predictive for observation $y_{N,T}$ given all other observations with

$$\begin{aligned}\mu_{N,T}^{\text{RMOGP}} &= m_T(\mathbf{x}_N) + \mathbf{K}_{1:NT-1,NT}^\top \big[\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w}\big]^{-1}_{1:NT-1,1:NT-1}\mathbf{z}_{1:NT-1} \\ (\sigma_{N,T}^{\text{RMOGP}})^2 &= \mathbf{K}_{(NT,NT)} - \mathbf{K}_{1:NT-1,NT}^\top \big[\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w}\big]^{-1}_{1:NT-1,1:NT-1}\mathbf{K}_{1:NT-1,NT},\end{aligned}$$

where $[\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w}]^{-1}_{1:NT-1,1:NT-1}$ represents the submatrix formed from the rows $\{1, \ldots, NT-1\}$ and columns $\{1, \ldots, NT-1\}$ of $\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w}$, and $\mathbf{K}_{1:NT-1,NT}$ denotes the submatrix formed from the NT-th column and the rows $\{1, \ldots, NT-1\}$ of $\mathbf{K}$. Finally, $\mathbf{K}_{(NT,NT)} = \mathcal{K}_{TT}(\mathbf{x}_N, \mathbf{x}_N)$ represents the element from the NT-th row and NT-th column of $\mathbf{K}$. Through this notation, we can write $\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w}$ as block matrix

$$\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w} = \begin{bmatrix}\big[\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w}\big]_{1:NT-1,1:NT-1} & \mathbf{K}_{1:NT-1,NT} \\ \mathbf{K}_{1:NT-1,NT}^\top & \mathbf{K}_{(NT,NT)} + \frac{1}{2}\sigma_T^4 w_T(\mathbf{x}_N, \mathbf{y}_N)^{-2}\end{bmatrix} = \begin{bmatrix}A & B \\ C & D\end{bmatrix},$$

where we use A, B, C and D for clarity in the derivation. Applying block matrix inversion leads to

$$(\mathbf{K}+\boldsymbol{\Sigma}\mathbf{J}_\mathbf{w})^{-1} = \begin{bmatrix}A^{-1} + A^{-1}B(D-CA^{-1}B)^{-1}CA^{-1} & -A^{-1}B(D-CA^{-1}B)^{-1} \\ -(D-CA^{-1}B)^{-1}CA^{-1} & (D-CA^{-1}B)^{-1}\end{bmatrix}$$

Therefore, the (NT, NT)-th element of this matrix is

$$\begin{aligned}\left[(\mathbf{K}+\mathbf{\Sigma J_w})^{-1}\right]_{(NT,NT)} &= (D-CA^{-1}B)^{-1}\\ &= \Big(\mathbf{K}_{(NT,NT)} + \frac{1}{2}\sigma_T^4 w_T(\mathbf{x}_N,\mathbf{y}_N)^{-2}\\ &\quad - \mathbf{K}_{1:NT-1,NT}^{\top}\left[\mathbf{K}+\mathbf{\Sigma J_w}\right]^{-1}_{1:NT-1,1:NT-1}\mathbf{K}_{1:NT-1,NT}\Big)^{-1}.\end{aligned}$$

We can then take the inverse on both sides and rearrange terms to get

$$\begin{aligned}\mathbf{K}_{(NT,NT)} - \mathbf{K}_{1:NT-1,NT}^{\top}\left[\mathbf{K}+\mathbf{\Sigma J_w}\right]^{-1}_{1:NT-1,1:NT-1}\mathbf{K}_{1:NT-1,NT}&\\ = \frac{1}{\left[(\mathbf{K}+\mathbf{\Sigma J_w})^{-1}\right]_{(NT,NT)}} - \frac{1}{2}\sigma_T^4 w_T(\mathbf{x}_N,\mathbf{y}_N)^{-2}.&\end{aligned}$$

This leads to the desired predictive variance

$$(\sigma^{\text{RMOGP}}_{N,T})^2 = \frac{1}{[(\mathbf{K}+\mathbf{\Sigma J_w})^{-1}]_{(NT,NT)}} - \frac{\sigma_T^4}{2} w_T(\mathbf{x}_N,\mathbf{y}_N)^{-2}.$$

Similarly, the predictive mean can be obtained by evaluating

$$\begin{aligned}&(\mathbf{K}+\mathbf{\Sigma J_w})^{-1}\mathbf{z}_{\text{vec}}\\ &\quad= \begin{bmatrix} A^{-1}+A^{-1}B(D-CA^{-1}B)^{-1}CA^{-1} & -A^{-1}B(D-CA^{-1}B)^{-1}\\ -(D-CA^{-1}B)^{-1}CA^{-1} & (D-CA^{-1}B)^{-1}\end{bmatrix}\begin{bmatrix}[\mathbf{z}_{\text{vec}}]_{1:NT-1}\\ z_{N,T}\end{bmatrix}.\end{aligned}$$

This leads to the expression

$$\begin{aligned}[(\mathbf{K}+\mathbf{\Sigma J_w})^{-1}\mathbf{z}_{\text{vec}}]_{NT} &= -(D-CA^{-1}B)^{-1}CA^{-1}[\mathbf{z}_{\text{vec}}]_{1:NT-1} + (D-CA^{-1}B)^{-1}z_{N,T}\\ &= (D-CA^{-1}B)^{-1}\big(z_{N,T} - CA^{-1}[\mathbf{z}_{\text{vec}}]_{1:NT-1}\big).\end{aligned}$$

We can then substitute $(D-CA^{-1}B)^{-1} = [(\mathbf{K}+\mathbf{\Sigma J_w})^{-1}]_{(NT,NT)}$, and replace A and C by their values to get

$$\begin{aligned}&[(\mathbf{K}+\mathbf{\Sigma J_w})^{-1}\mathbf{z}_{\text{vec}}]_{NT}\\ &= [(\mathbf{K}+\mathbf{\Sigma J_w})^{-1}]_{(NT,NT)}\big(z_{N,T} - \mathbf{K}_{1:NT-1,NT}^{\top}\left[\mathbf{K}+\mathbf{\Sigma J_w}\right]^{-1}_{1:NT-1,1:NT-1}[\mathbf{z}_{\text{vec}}]_{1:NT-1}\big).\end{aligned}$$

Finally, by rearranging terms, we get

$$\mathbf{K}_{1:NT-1,NT}^{\top}\left[\mathbf{K}+\mathbf{\Sigma J_w}\right]^{-1}_{1:NT-1,1:NT-1}[\mathbf{z}_{\text{vec}}]_{1:NT-1} = z_{N,T} - \frac{[(\mathbf{K}+\mathbf{\Sigma J_w})^{-1}\mathbf{z}_{\text{vec}}]_{NT}}{[(\mathbf{K}+\mathbf{\Sigma J_w})^{-1}]_{(NT,NT)}},$$

which leads to the desired predictive mean

$$\mu^{\text{RMOGP}}_{N,T} = z_{N,T} + m_T(\mathbf{x}_N) - \frac{[(\mathbf{K}+\mathbf{\Sigma J_w})^{-1}\mathbf{z}_{\text{vec}}]_{NT}}{[(\mathbf{K}+\mathbf{\Sigma J_w})^{-1}]_{(NT,NT)}}.$$

□

B.3 Proof of Proposition 3.2

We consider the contamination of dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$, where the observation $y_{m,s}$ with index $m \in \{1, \dots N\}$ in the channel $s \in \{1, \dots, T\}$ is replaced by an arbitrary large contaminated value $y^c_{m,s}$. This results in the contaminated dataset $\mathcal{D}^c_{m,s} = (\mathcal{D} \setminus \{(\mathbf{x}_m, \mathbf{y}_m)\}) \cup \{(\mathbf{x}_m, \mathbf{y}^c_{m,s})\}$, where $\mathbf{y}^c_{m,s} = [y_{m,1}, \dots, y^c_{m,s}, \dots, y_{m,T}]^{\top}$.

PIF for MOGP First, we show that a single contaminated observation can increase the PIF arbitrarily in any output t in the MOGP.

Lemma B.4. *Suppose* $\mathbf{f} \sim \mathcal{GP}(\mathbf{m}, \mathcal{K})$*, where* f_t *denotes the* t*-th channel of* $\mathbf{f}$*, and suppose* $\text{vec}(\mathcal{E}) \sim \mathcal{N}(0, \mathbf{\Sigma})$*. Then, for a standard MOGP, there exists a positive constant* C_1 *independent of* $y^c_{m,s}$ *such that*

$$\text{PIF}_{\text{MOGP}}(y^c_{m,s}, f_t, \mathcal{D}) = C_1(y^c_{m,s} - y_{m,s})^2,$$

Proof. Without loss of generality, we prove the bound for $t = 1$, and this can be extended for arbitrary $t \in \{1, \ldots, T\}$ using a permutation matrix. Let $p(f_1(X)|\mathcal{D})$ and $p(f_1(X)|\mathcal{D}^c_{m,s})$ be the uncontaminated and contaminated marginal MOGP posterior of the first output function, where $f_1(X) = [f_{1,1}, \ldots, f_{N,1}]^\top$. This marginal posterior given the uncontaminated data follows the distribution

$$\begin{aligned} p(f_1(X)|\mathcal{D}) &= \mathcal{N}(f_1(X); \mu_1, \Sigma_{11}), \\ \mu_1 &= [\mathbf{m}_{\text{vec}}]_{1:N} + [\mathbf{K}(\mathbf{K} + \mathbf{\Sigma})^{-1}(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\text{vec}})]_{1:N}, \\ \Sigma_{11} &= [\mathbf{K}(\mathbf{K} + \mathbf{\Sigma})^{-1}\mathbf{\Sigma}]_{1:N,1:N}. \end{aligned}$$

Similarly, the marginal posterior of the contaminated data follows the distribution

$$\begin{aligned} p(f_1(X)|\mathcal{D}^c_{m,s}) &= \mathcal{N}(f_1(X); \mu^c_1, \Sigma^c_{11}), \\ \mu^c_1 &= [\mathbf{m}_{\text{vec}}]_{1:N} + [\mathbf{K}(\mathbf{K} + \mathbf{\Sigma})^{-1}(\mathbf{y}^c_{\text{vec}} - \mathbf{m}_{\text{vec}})]_{1:N}, \\ \Sigma^c_{11} &= \Sigma_{11}, \end{aligned}$$

where $\mathbf{y}^c_{\text{vec}} = \left[\, y_{1,1}, \ldots, y_{m-1,s},\ y^c_{m,s},\ y_{m+1,s}, \ldots, y_{N,T} \,\right]^\top$. Note that the covariance matrices of both uncontaminated and contaminated posteriors are equal since they only depend on the inputs, and are therefore not affected by the contamination. Since both posteriors are Gaussian distributions, the KL divergence is available in closed form such that

$$\text{PIF}_{\text{MOGP}}(y^c_{m,s}, f_1, \mathcal{D}) = \frac{1}{2}\left[\text{Tr}((\Sigma^c_{11})^{-1}\Sigma_{11}) - N + (\mu^c_1 - \mu_1)^\top(\Sigma^c_{11})^{-1}(\mu^c_1 - \mu_1) + \log\left(\frac{\det\Sigma^c_{11}}{\det\Sigma_{11}}\right)\right].$$

Since $\Sigma^c_{11} = \Sigma_{11}$, it can be derived that

$$\begin{aligned} \text{Tr}((\Sigma^c_{11})^{-1}\Sigma_{11}) - N &= \text{Tr}(\Sigma^{-1}_{11}\Sigma_{11}) - N = \text{Tr}(I_N) - N = N - N = 0, \\ \log\left(\frac{\det\Sigma^c_{11}}{\det\Sigma_{11}}\right) &= \log\left(\frac{\det\Sigma_{11}}{\det\Sigma_{11}}\right) = \log(1) = 0. \end{aligned}$$

Therefore, the PIF is simplified to

$$\text{PIF}_{\text{MOGP}}(y^c_{m,s}, f_1, \mathcal{D}) = \frac{1}{2}(\mu^c_1 - \mu_1)^\top(\Sigma^c_{11})^{-1}(\mu^c_1 - \mu_1).$$

We derive the following expression for $\mu_1 - \mu^c_1$ by substituting the expressions above, and cancelling matching positive and negative terms.

$$\begin{aligned} &\mu_1 - \mu^c_1 \\ &= [\mathbf{m}_{\text{vec}}]_{1:N} + [\mathbf{K}(\mathbf{K} + \mathbf{\Sigma})^{-1}(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\text{vec}})]_{1:N} - [\mathbf{m}_{\text{vec}}]_{1:N} - [\mathbf{K}(\mathbf{K} + \mathbf{\Sigma})^{-1}(\mathbf{y}^c_{\text{vec}} - \mathbf{m}_{\text{vec}})]_{1:N} \\ &= [\mathbf{K}]_{1:N,1:NT}(\mathbf{K} + \mathbf{\Sigma})^{-1}(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\text{vec}}) - [\mathbf{K}]_{1:N,1:NT}(\mathbf{K} + \mathbf{\Sigma})^{-1}(\mathbf{y}^c_{\text{vec}} - \mathbf{m}_{\text{vec}}) \\ &= [\mathbf{K}]_{1:N,1:NT}(\mathbf{K} + \mathbf{\Sigma})^{-1}(\mathbf{y}_{\text{vec}} - \mathbf{y}^c_{\text{vec}}) \\ &= [\mathbf{K}]_{1:N,1:NT}\Omega(\mathbf{y}_{\text{vec}} - \mathbf{y}^c_{\text{vec}}), \end{aligned}$$

where we define $\Omega := (\mathbf{K} + \mathbf{\Sigma})^{-1}$ not depending on $\mathbf{y}$. Substituting the previously derived expression for $\mu_1 - \mu^c_1$ and using $(\Sigma^c_{11})^{-1} = [\mathbf{K}\Omega\mathbf{\Sigma}]^{-1}_{1:N,1:N}$ yields

$$\begin{aligned} \text{PIF}_{\text{MOGP}}(y^c_{m,s}, f_1, \mathcal{D}) &= \frac{1}{2}([\mathbf{K}]_{1:N,1:NT}\Omega(\mathbf{y}_{\text{vec}} - \mathbf{y}^c_{\text{vec}}))^\top[\mathbf{K}\Omega\mathbf{\Sigma}]^{-1}_{1:N,1:N}([\mathbf{K}]_{1:N,1:NT}\Omega(\mathbf{y}_{\text{vec}} - \mathbf{y}^c_{\text{vec}})) \\ &= \frac{1}{2}(\mathbf{y}_{\text{vec}} - \mathbf{y}^c_{\text{vec}})^\top\Omega[\mathbf{K}]_{1:NT,1:N}[\mathbf{K}\Omega\mathbf{\Sigma}]^{-1}_{1:N,1:N}[\mathbf{K}]_{1:N,1:NT}\Omega(\mathbf{y}_{\text{vec}} - \mathbf{y}^c_{\text{vec}}) \\ &= \frac{1}{2}(\mathbf{y}_{\text{vec}} - \mathbf{y}^c_{\text{vec}})^\top\Omega[\mathbf{K}]_{1:NT,1:N}\Psi[\mathbf{K}]_{1:N,1:NT}\Omega(\mathbf{y}_{\text{vec}} - \mathbf{y}^c_{\text{vec}}), \end{aligned}$$

where in the second equality we expand the transpose, and in the third equality we define $\Psi = [\mathbf{K}\Omega\boldsymbol{\Sigma}]^{-1}_{1:N,1:N}$, which is independent of the contamination. Since $\mathbf{y}_{\text{vec}}$ and $\mathbf{y}^c_{\text{vec}}$ are equal except for the $((s-1)N+m)$-th element, we obtain the desired expression

$$\begin{aligned}\text{PIF}_{\text{MOGP}}(y^c_{m,s}, f_1, \mathcal{D}) &= \frac{1}{2}\Big[\Omega[\mathbf{K}]_{1:NT,1:N}\Psi[\mathbf{K}]_{1:N,1:NT}\Omega\Big]_{((s-1)N+m),((s-1)N+m)}(y_{m,s}-y^c_{m,s})^2 \\ &= C_1(y_{m,s}-y^c_{m,s})^2,\end{aligned}$$

where $C_1 = \frac{1}{2}[\Omega[\mathbf{K}]_{1:NT,1:N}\Psi[\mathbf{K}]_{1:N,1:NT}\Omega]_{((s-1)N+m),((s-1)N+m)}$ does not depend on $y^c_{m,s}$. □

Consequently, MOGPs are not robust since $\text{PIF}_{\text{MOGP}}(y^c_{m,s}, f_t, \mathcal{D}) \to \infty$ as $|y^c_{m,s} - y_{m,s}| \to \infty$.

PIF for Independent Outputs in the MOGP We now consider the case where $s \neq t$, and the covariance matrix has a block-diagonal structure such that s and t belong to different blocks. In other words, the outputs s and t are independent of each other, and contamination in s therefore has no effect on the posterior influence function for t. More formally, let the set of outputs be partitioned into Q disjoint index blocks $\{G_1, \ldots, G_Q\}$ with $\bigcup_{q=1}^{Q} G_q = \{1, \ldots, T\}$. We say the MOGP output covariance matrix is *block-diagonal* w.r.t. this partition if for any $a \neq b$ and any $t \in G_a, s \in G_b$ the covariance matrix satisfies

$$\mathcal{K}_{s,t}(\mathbf{x}, \mathbf{x}) = \text{Cov}(f_s(\mathbf{x}), f_t(\mathbf{x})) = 0$$

Lemma B.5. *Let the MOGP be as in Lemma B.4. Assume the prior/joint covariance has a block-diagonal structure with respect to the partition $G_1, \ldots, G_Q$ defined above. If $s \neq t$ and $s \in G_a, t \in G_b$ for some $a \neq b$, it follows that $C_1 = 0$, and consequently*

$$\text{PIF}_{\text{MOGP}}(y^c_{m,s}, f_t, \mathcal{D}) = 0.$$

Proof. Again, without loss of generality, we prove the result for $t = 1$. Additionally, assume that the first block—which includes output $t = 1$—contains $R < s$ outputs, while the remaining $T - R$ outputs belong to the second (and final) block. We can then express the matrix $\mathbf{K} + \boldsymbol{\Sigma}$ as

$$\mathbf{K}+\boldsymbol{\Sigma} = \begin{bmatrix}[\mathbf{K}+\boldsymbol{\Sigma}]_{1:NR,1:NR} & [\mathbf{K}]_{1:NR,NR+1:NT} \\ [\mathbf{K}]_{NR+1:NT,1:NR} & [\mathbf{K}+\boldsymbol{\Sigma}]_{NR+1:NT,NR+1:NT}\end{bmatrix} = \begin{bmatrix}[\mathbf{K}+\boldsymbol{\Sigma}]_{1:NR,1:NR} & \mathbf{0} \\ \mathbf{0} & [\mathbf{K}+\boldsymbol{\Sigma}]_{NR+1:NT,NR+1:NT}\end{bmatrix}$$

$[\boldsymbol{\Sigma}]_{NR+1:NT,1:NR} = \mathbf{0}$ since $\boldsymbol{\Sigma}$ is a diagonal matrix, and $[\mathbf{K}]_{NR+1:NT,1:NR} = \mathbf{0}$ since $\mathbf{K}$ is block-diagonal. It is notationally cumbersome but straightforward to show, using block matrix inversion, that

$$\Omega := (\mathbf{K}+\boldsymbol{\Sigma})^{-1} = \begin{bmatrix}\Omega_{1:NR,1:NR} & \mathbf{0} \\ \mathbf{0} & \Omega_{NR+1:NT,NR+1:NT}\end{bmatrix}$$

for some block matrices $\Omega_{1:NR,1:NR} \in \mathbb{R}^{NR\times NR}$ and $\Omega_{NR+1:NT,NR+1:NT} \in \mathbb{R}^{N(T-R)\times N(T-R)}$. Additionally, the block matrix $[\mathbf{K}]_{1:N,1:NT}$ can be written such that

$$[\mathbf{K}]_{1:N,1:NT} = [[\mathbf{K}]_{1:N,1:NR} \quad [\mathbf{K}]_{1:N,NR+1:NT}] = [[\mathbf{K}]_{1:N,1:NR} \quad \mathbf{0}],$$

where $[\mathbf{K}]_{1:N,NR+1:NT} = \mathbf{0}$ since $\mathbf{K}$ is block-diagonal. Multiplying $[\mathbf{K}]_{1:N,1:NT}$ with Ω, it can be shown that

$$[\mathbf{K}]_{1:N,1:NT}\Omega = [[\mathbf{K}]_{1:N,1:NR}\Omega_{1:NR,1:NR} \quad \mathbf{0}].$$

This results in the following expression:

$$\Omega[\mathbf{K}]_{1:NT,1:N}\Psi[\mathbf{K}]_{1:N,1:NT}\Omega = \begin{bmatrix}\Omega_{1:NR,1:NR}[\mathbf{K}]_{1:NR,1:N}\Psi[\mathbf{K}]_{1:N,1:NR}\Omega_{1:NR,1:NR} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}.$$

We now note that, since $R < s$, the $((s-1)N+m)$-th diagonal element of this matrix equals zero, i.e. $C_1 = 0$. Therefore, $\text{PIF}_{\text{MOGP}}(y^c_{m,s}, f_1, \mathcal{D}) = 0$. □

PIF for R-MOGP

Proposition 3.2 *Suppose* $\mathbf{f} \sim \mathcal{GP}(\mathbf{m}, \mathcal{K})$, $\text{vec}(\mathcal{E}) \sim \mathcal{N}(0, \mathbf{\Sigma})$, *and* $W_i = \text{diag}(w_1(\mathbf{x}_i, \mathbf{y}_i), \ldots, w_T(\mathbf{x}_i, \mathbf{y}_i))$, *where* w_t *is defined as in* (7). *Then, there exists a constant* $C_2 > 0$ *independent of* $y^c_{m,s}$ *for which*

$$\max_{1 \leq t \leq T} \sup_{y^c_{m,s} \in \mathbb{R}} \text{PIF}(y^c_{m,s}, f_t, \mathcal{D}) \leq C_2.$$

Proof. To prove robustness of R-MOGPs, we make use of Proposition 3.2 in Altamirano et al. (2024), which states that the PIF over the *Gaussian* posterior over *all observations* is bounded under mild conditions for the weights. In our multi-output setting, we vectorise the outputs across all channels into a single observations vector. As a result, we can adapt this proposition to the multi-output setting:

Proposition B.6 (Altamirano et al., 2024)**.** *Suppose* $\mathbf{f} \sim \mathcal{GP}(\mathbf{m}, \mathcal{K})$ *and* $\text{vec}(\mathcal{E}) \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$*, and assume that for all* $t \in \{1, \ldots, T\}$, $\sup_{\mathbf{x},\mathbf{y}} w_t(\mathbf{x}, \mathbf{y}) < \infty$ *and* $\sup_{\mathbf{x},\mathbf{y}} |y_t \cdot w_t(\mathbf{x}, \mathbf{y})^2| < \infty$*. Then there exists a constant* $C_2 > 0$ *independent of* $y^c_{m,s}$ *such that the PIF for the* ***joint distribution*** *of R-MOGPs satisfies*

$$\text{PIF}_{\text{RMOGP}}(y^c_{m,s}, \mathcal{D}) \leq C_2.$$

To demonstrate the robustness of the *marginal distributions* of R-MOGPs, it remains to show that the *marginal* PIFs are bounded and that the assumptions on the weighting functions are satisfied. To show the former, we make use of the chain rule for the KL divergence (Cover and Thomas, 1991). This property states that the divergence between joint distributions is at least as large as the divergence between their marginal distributions. Formally, let $p(x, y)$ and $q(x, y)$ be joint probability densities over random variables $\tilde{x}$ and $\tilde{y}$. Then the KL divergence between these joint distributions decomposes as:

$$\begin{aligned} \text{KL}(p(x,y) \| q(x,y)) &= \text{KL}(p(x) \| q(x)) + \mathbb{E}_{\tilde{x} \sim p(\tilde{x})} \left[\text{KL}(p(y|\tilde{x}) \,\|\, q(y|\tilde{x})) \right] \\ &\geq \text{KL}(p(x) \| q(x)), \end{aligned}$$

since the divergence term is non-negative, i.e. $\text{KL}(p(y|\tilde{x}) \,\|\, q(y|\tilde{x})) \geq 0$. In other words, the KL divergence of the marginal distributions is less than or equal to the KL divergence of the joint distributions. Applying this result to R-MOGPs yields

$$\text{PIF}_{\text{RMOGP}}(y^c_{m,s}, f_t, \mathcal{D}) \leq \text{PIF}_{\text{RMOGP}}(y^c_{m,s}, \mathcal{D}) \leq C_2.$$

Thus, the marginal PIFs in R-MOGPs are bounded, which ensures their robustness. We now verify that the assumptions on the weights are satisfied. Specifically, we show that

$$\sup_{\mathbf{x},\mathbf{y}} w_t(\mathbf{x}, \mathbf{y}) < \infty \qquad \text{and} \qquad \sup_{\mathbf{x},\mathbf{y}} |y_t \cdot w_t(\mathbf{x}, \mathbf{y})^2| < \infty.$$

The first condition holds trivially since $w_t(\mathbf{x}, \mathbf{y}) \leq \beta_t = \frac{\sigma_t}{\sqrt{2}} < \infty$. To establish the second condition, we refer to the result in Appendix B.3 of Laplante et al. (2025) adapted to the multi-output setting, which states that $\sup_{\mathbf{x},\mathbf{y}} |y_t \cdot w_t(\mathbf{x}, \mathbf{y})^2| < \infty$ provided that $|\gamma_t(\mathbf{x}, \mathbf{y})| < \infty$ and $|c_t(\mathbf{x}, \mathbf{y})| < \infty$. Since $\gamma_t(\mathbf{x}, \mathbf{y})$ represents the mean of a Gaussian distribution, it is bounded. Finally, $c_t(\mathbf{x}, \mathbf{y})$ is also bounded since it is the quantile absolute deviation of a set of finite elements. Thus, the second condition also holds. □

B.4 Sparse Variational R-MOGPs

Let the MOGP be modelled using the ICM with R underlying latent functions such that $B = AA^\top$ for $A \in \mathbb{R}^{T \times R}$, and each latent function is approximated independently with a set of Q inducing points per latent function. We then define the matrix $U \in \mathbb{R}^{Q \times R}$ for the function evaluation over all inducing points with corresponding inducing location matrix for each latent function $Z_r \in \mathbb{R}^{Q \times d}$ for $r \in \{1, \ldots, R\}$. The sparse variational R-MOGP predictive is then given by:

$$\begin{aligned} p_W(f_\star | \mathbf{x}_\star, Y, X) &= \mathcal{N}(f_\star; \boldsymbol{\mu}^S_\star, \mathbf{\Sigma}^S_\star), \\ \boldsymbol{\mu}^S_\star &= \mathbf{m}_\star + \mathbf{k}^\top_{U\star} \mathbf{P}^{-1}_U \mathbf{K}^\top_{FU} \boldsymbol{\nu} \\ \mathbf{\Sigma}^S_\star &= \mathbf{k}_{\star\star} - \mathbf{k}^\top_{U\star} \mathbf{K}^{-1}_{UU} \mathbf{k}_{U\star} + \mathbf{k}^\top_{U\star} \mathbf{P}^{-1}_U \mathbf{k}_{U\star}, \end{aligned}$$

where $\boldsymbol{\nu} = \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{w},\text{vec}})$, $\mathbf{P}_U = \mathbf{K}_{UU} + \mathbf{K}_{FU}^{\top}\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}\mathbf{K}_{FU}$, $\mathbf{K}_{UU} = \text{diag}(\kappa(Z_1, Z_1), \ldots, \kappa(Z_R, Z_R)) \in \mathbb{R}^{QR\times QR}$ is a diagonal block matrix, and $\mathbf{K}_{FU} \in \mathbb{R}^{NT\times QR}$ is a block-matrix such that the t, r-th block of $\mathbf{K}_{FU}$ equals $a_{t,r}\kappa(X, Z_r)$ for $t \in \{1, \ldots, T\}$ and $r \in \{1, \ldots, R\}$. In addition, $\mathbf{k}_{U\star} \in \mathbb{R}^{QR\times T}$ is the block-matrix such that $[\mathbf{k}_{U\star}]_{r,t} = a_{t,r}\kappa(Z_r, \mathbf{x}_\star)$. The corresponding ELBO is given as:

$$\text{ELBO}(Z, \Sigma, \boldsymbol{\theta}) = -\frac{1}{2}\left(\text{Tr}\left(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{w}}^{-1/2}(\mathbf{K} - \mathbf{K}_{FU}\mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^{\top})\mathbf{J}_{\mathbf{w}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\right) + C(\Sigma, \boldsymbol{\theta})\right) + \frac{1}{2}\boldsymbol{\nu}^{\top}\mathbf{K}_{FU}\mathbf{P}_U^{-1}\mathbf{K}_{FU}^{\top}\boldsymbol{\nu} + \frac{1}{2}\log\left(\frac{\det(\mathbf{K}_{UU})}{\det(\mathbf{P}_U)}\right).$$

where $C(\Sigma, \boldsymbol{\theta})$ is a function that depends only on Σ and $\boldsymbol{\theta}$.

Proof. Without loss of generality, we derive the predictive for $\mathbf{m}_{\text{vec}} = 0$, and then extend it to an arbitrary prior mean. The ICM allows us to model the output as a linear combination of R latent functions $v_r(\mathbf{x})$ for $r \in \{1, \ldots, R\}$ such that

$$f_t(\mathbf{x}) = a_{t1}v_1(\mathbf{x}) + \cdots + a_{tR}v_R(\mathbf{x}),$$

where each latent function is modelled as an independent GP such that $v_r \sim \mathcal{GP}(0, \kappa)$. The inducing point evaluations in U are then given as $[U]_{qr} = v_r(\mathbf{z}_q^{(r)})$, where $\mathbf{z}_q^{(r)}$ is an inducing location such that $Z_r = [z_1^{(r)}, \ldots, z_Q^{(r)}]^{\top}$. We define $\mathbf{u}_{\text{vec}} = \text{vec}(U) \in \mathbb{R}^{RQ}$ by stacking its columns. The variational distribution is then given as:

$$q(\mathbf{f}_{\text{vec}}, \mathbf{u}_{\text{vec}}) = p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})q(\mathbf{u}_{\text{vec}}),$$

where $q(\mathbf{u}_{\text{vec}}) = \mathcal{N}(\mathbf{u}_{\text{vec}}; \boldsymbol{\mu}_U, \boldsymbol{\Sigma}_U)$. We now seek to approximate the exact posterior $p(\mathbf{f}_{\text{vec}}, \mathbf{u}_{\text{vec}}|\mathbf{y}_{\text{vec}})$ by this variational distribution by minimising the KL divergence between $q(\mathbf{f}_{\text{vec}}, \mathbf{u}_{\text{vec}})$ and $p(\mathbf{f}_{\text{vec}}, \mathbf{u}_{\text{vec}}|\mathbf{y}_{\text{vec}})$. This is equivalent to maximising ELBO:

$$\text{ELBO}(Z, \Sigma, \boldsymbol{\theta}) = \int \log\frac{\Psi(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}})p(\mathbf{u}_{\text{vec}})}{q(\mathbf{u}_{\text{vec}})}q(\mathbf{u}_{\text{vec}})d\mathbf{u}_{\text{vec}},$$

where

$$\Psi(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}}) = \exp\left(\int \log(p(\mathbf{y}_{\text{vec}}|\mathbf{f}_{\text{vec}}))p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})d\mathbf{f}_{\text{vec}}\right).$$

It is straightforward to show that the optimal variational distribution is given by $q(\mathbf{u}_{\text{vec}}) \propto \Psi(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}})p(\mathbf{u}_{\text{vec}})$ such that

$$q(\mathbf{u}_{\text{vec}}) = \frac{\Psi(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}})p(\mathbf{u}_{\text{vec}})}{\int \Psi(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}})p(\mathbf{u}_{\text{vec}})d\mathbf{u}_{\text{vec}}}.$$

This results in the following ELBO:

$$\text{ELBO}(Z, \Sigma, \boldsymbol{\theta}) = \log\left(\int \Psi(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}})p(\mathbf{u}_{\text{vec}})d\mathbf{u}_{\text{vec}}\right).$$

For the sparse variational R-MOGP, we replace the standard likelihood $p(\mathbf{y}_{\text{vec}}|\mathbf{f}_{\text{vec}})$ with $\exp(-N\mathcal{L}_W(F, Y, X))/((2\pi)^{NT/2}\text{det}(\boldsymbol{\Sigma})^{1/2})$ in $\Psi(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}})$. We divide by the constant $(2\pi)^{NT/2}\text{det}(\boldsymbol{\Sigma})^{1/2}$ as it is independent of $\mathbf{y}_{\text{vec}}$ and is used to regularise optimisation with respect to Σ. This results in the following optimal robust variational distribution:

$$q_W(\mathbf{u}_{\text{vec}}) = \frac{\Psi_W(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}})p(\mathbf{u}_{\text{vec}})}{\int \Psi_W(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}})p(\mathbf{u}_{\text{vec}})d\mathbf{u}_{\text{vec}}},$$

where

$$\begin{aligned}\log\Psi_W(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}}) &= \int \log\left(\exp(-N\mathcal{L}_W(F, Y, X))\right)p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})d\mathbf{f}_{\text{vec}} \\ &= \int \log\left(\exp\left(-\frac{1}{2}\left[\mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}\mathbf{f}_{\text{vec}} - 2\mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\nu} + C(\Sigma, \boldsymbol{\theta})\right]\right)\right)p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})d\mathbf{f}_{\text{vec}} \\ &= \int -\frac{1}{2}\left[\mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}\mathbf{f}_{\text{vec}} - 2\mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\nu} + C(\Sigma, \boldsymbol{\theta})\right]p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})d\mathbf{f}_{\text{vec}},\end{aligned}$$

where $\boldsymbol{\nu} = \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{W},\text{vec}})$ and $C(\Sigma, \boldsymbol{\theta}) = 2\sum_{i=1}^{N} c_i + NT\log(2\pi) + \log\det(\boldsymbol{\Sigma})$, where c_i is given in Appendix B.1. The conditional distribution of $\mathbf{f}_{\text{vec}}$ given $\mathbf{u}_{\text{vec}}$ is given as

$$
\begin{aligned}
p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}) &= \mathcal{N}(\mathbf{f}_{\text{vec}}; \boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}, \boldsymbol{\Sigma}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}),\\
\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}} &= \mathbf{K}_{FU}\mathbf{K}_{UU}^{-1}\mathbf{u}_{\text{vec}},\\
\boldsymbol{\Sigma}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}} &= \mathbf{K} - \mathbf{K}_{FU}\mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^{\top},
\end{aligned}
$$

where $\mathbf{K}_{UU} = \text{diag}(\kappa(Z_1, Z_1), \ldots, \kappa(Z_R, Z_R)) \in \mathbb{R}^{QR\times QR}$ is a diagonal block matrix, and $\mathbf{K}_{FU} \in \mathbb{R}^{NT\times QR}$ is a block matrix such that the t, r-th block of $\mathbf{K}_{FU}$ equals $a_{t,r}\kappa(X, Z_r)$. We can then write the above expression as:

$$
\begin{aligned}
&\log\Psi_W(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}})\\
&\quad= -\frac{1}{2}\int \mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\mathbf{f}_{\text{vec}}p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})d\mathbf{f}_{\text{vec}} + \int \mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\nu}p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})d\mathbf{f}_{\text{vec}} - \frac{1}{2}C(\Sigma, \boldsymbol{\theta})\\
&\quad= -\frac{1}{2}\int \mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\mathbf{f}_{\text{vec}}p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})d\mathbf{f}_{\text{vec}} + \mathbb{E}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}\left[\mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\nu}\right] - \frac{1}{2}C(\Sigma, \boldsymbol{\theta}).
\end{aligned}
$$

We now recall that the inner product between two vectors equals the trace of their outer product such that

$$
\mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{W}}^{-1}\mathbf{f}_{\text{vec}} = (\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{W}}^{-1/2}\mathbf{f}_{\text{vec}})^{\top}(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{W}}^{-1/2}\mathbf{f}_{\text{vec}}) = \text{Tr}(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{W}}^{-1/2}\mathbf{f}_{\text{vec}}\mathbf{f}_{\text{vec}}^{\top}\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}).
$$

Plugging this into above expression, we get:

$$
\begin{aligned}
&\log\Psi_W(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}})\\
&= -\frac{1}{2}\int \text{Tr}(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{W}}^{-1/2}\mathbf{f}_{\text{vec}}\mathbf{f}_{\text{vec}}^{\top}\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\Sigma}^{-1/2})p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})d\mathbf{f}_{\text{vec}} + \mathbb{E}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}\left[\mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\nu}\right] - \frac{1}{2}C(\Sigma, \boldsymbol{\theta})\\
&= -\frac{1}{2}\mathbb{E}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}\left[\text{Tr}(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{W}}^{-1/2}\mathbf{f}_{\text{vec}}\mathbf{f}_{\text{vec}}^{\top}\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\Sigma}^{-1/2})\right] + \mathbb{E}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}\left[\mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\nu}\right] - \frac{1}{2}C(\Sigma, \boldsymbol{\theta}).
\end{aligned}
$$

We can use that the expectation of the trace equals the trace of the expectation to get that

$$
\begin{aligned}
&\log\Psi_W(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}})\\
&= -\frac{1}{2}\text{Tr}\left(\mathbb{E}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}\left[\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{W}}^{-1/2}\mathbf{f}_{\text{vec}}\mathbf{f}_{\text{vec}}^{\top}\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\right]\right) + \mathbb{E}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}\left[\mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\nu}\right] - \frac{1}{2}C(\Sigma, \boldsymbol{\theta})\\
&= -\frac{1}{2}\text{Tr}\left(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{W}}^{-1/2}\mathbb{E}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}\left[\mathbf{f}_{\text{vec}}\mathbf{f}_{\text{vec}}^{\top}\right]\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\right) + \mathbb{E}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}\left[\mathbf{f}_{\text{vec}}^{\top}\boldsymbol{\nu}\right] - \frac{1}{2}C(\Sigma, \boldsymbol{\theta}).
\end{aligned}
$$

We now recall that

$$
\mathbb{E}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}[\mathbf{f}_{\text{vec}}\mathbf{f}_{\text{vec}}^{\top}] = \mathbb{V}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}[\mathbf{f}_{\text{vec}}] + \mathbb{E}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}[\mathbf{f}_{\text{vec}}]\mathbb{E}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}[\mathbf{f}_{\text{vec}}]^{\top}
$$

to get the following expression:

$$
\begin{aligned}
\log\Psi_W(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}}) = -\frac{1}{2}\text{Tr}&\left(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{W}}^{-1/2}\left(\mathbb{V}[\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}] + \mathbb{E}[\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}]\mathbb{E}[\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}]^{\top}\right)\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\right)\\
&+ \mathbb{E}\left[\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}\right]^{\top}\boldsymbol{\nu} - \frac{1}{2}C(\Sigma, \boldsymbol{\theta}).
\end{aligned}
$$

Now note that $\mathbb{V}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}[\mathbf{f}_{\text{vec}}]$ and $\mathbb{E}_{\mathbf{f}_{\text{vec}}\sim p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}})}[\mathbf{f}_{\text{vec}}]$ are known since $p(\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}) = \mathcal{N}(\mathbf{f}_{\text{vec}}; \boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}, \boldsymbol{\Sigma}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}})$. We can substitute these terms to get:

$$
\begin{aligned}
&\log\Psi_W(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}})\\
&= -\frac{1}{2}\text{Tr}\left(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{W}}^{-1/2}\left(\boldsymbol{\Sigma}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}} + \boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^{\top}\right)\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\right) + \boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^{\top}\boldsymbol{\nu} - \frac{1}{2}C(\Sigma, \boldsymbol{\theta})\\
&= -\frac{1}{2}\text{Tr}\left(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\Sigma}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\right)\\
&\qquad - \frac{1}{2}\text{Tr}\left(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^{\top}\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\right) + \boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^{\top}\boldsymbol{\nu} - \frac{1}{2}C(\Sigma, \boldsymbol{\theta}).
\end{aligned}
$$

We can then again use that trace of the outer product of a vector equals the inner product such that:

$$
\begin{aligned}
\log\Psi_W(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}}) = &-\frac{1}{2}\text{Tr}\left(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\Sigma}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}\mathbf{J}_{\mathbf{W}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\right)\\
&- \frac{1}{2}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^{\top}\mathbf{J}_{\mathbf{W}}^{-1}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}} + \boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^{\top}\boldsymbol{\nu} - \frac{1}{2}C(\Sigma, \boldsymbol{\theta}).
\end{aligned}
$$

To complete the proof, we plug Ψ_W into the optimal variational distribution:

$$
\begin{aligned}
&q_W(\mathbf{u}_{\text{vec}}) \\
&\propto \Psi_W(\mathbf{y}_{\text{vec}}, \mathbf{u}_{\text{vec}}) p(\mathbf{u}_{\text{vec}}) \\
&\propto \exp\left(-\frac{1}{2}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^\top \mathbf{J}_{\mathbf{w}}^{-1}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}} + \boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^\top \boldsymbol{\nu} - \frac{1}{2}\mathbf{u}_{\text{vec}}^\top \mathbf{K}_{UU}^{-1}\mathbf{u}_{\text{vec}}\right) \\
&= \exp\left(-\frac{1}{2}\mathbf{u}_{\text{vec}}^\top \mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^\top \mathbf{J}_{\mathbf{w}}^{-1}\boldsymbol{\Sigma}^{-1}\mathbf{K}_{FU}\mathbf{K}_{UU}^{-1}\mathbf{u}_{\text{vec}} + \mathbf{u}_{\text{vec}}^\top \mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^\top \boldsymbol{\nu} - \frac{1}{2}\mathbf{u}_{\text{vec}}^\top \mathbf{K}_{UU}^{-1}\mathbf{u}_{\text{vec}}\right) \\
&= \exp\left(-\frac{1}{2}\mathbf{u}_{\text{vec}}^\top \left(\mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^\top \mathbf{J}_{\mathbf{w}}^{-1}\boldsymbol{\Sigma}^{-1}\mathbf{K}_{FU}\mathbf{K}_{UU}^{-1} + \mathbf{K}_{UU}^{-1}\right)\mathbf{u}_{\text{vec}} + \mathbf{u}_{\text{vec}}^\top \mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^\top \boldsymbol{\nu}\right).
\end{aligned}
$$

By completing squares, we get:

$$
\begin{aligned}
q_W(\mathbf{u}_{\text{vec}}) &\propto \exp\left(-\frac{1}{2}(\mathbf{u}_{\text{vec}} - \boldsymbol{\mu}_U)^\top \boldsymbol{\Sigma}_U^{-1}(\mathbf{u}_{\text{vec}} - \boldsymbol{\mu}_U)\right), \\
\boldsymbol{\Sigma}_U &= (\mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^\top \mathbf{J}_{\mathbf{w}}^{-1}\boldsymbol{\Sigma}^{-1}\mathbf{K}_{FU}\mathbf{K}_{UU}^{-1} + \mathbf{K}_{UU}^{-1})^{-1} \\
&= \mathbf{K}_{UU}(\mathbf{K}_{UU} + \mathbf{K}_{FU}^\top \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}\mathbf{K}_{FU})^{-1}\mathbf{K}_{UU} \\
&= \mathbf{K}_{UU}\mathbf{P}_U^{-1}\mathbf{K}_{UU}, \\
\boldsymbol{\mu}_U &= \boldsymbol{\Sigma}_U \mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^\top \boldsymbol{\nu} \\
&= \mathbf{K}_{UU}\mathbf{P}_U^{-1}\mathbf{K}_{UU}\mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^\top \boldsymbol{\nu} \\
&= \mathbf{K}_{UU}\mathbf{P}_U^{-1}\mathbf{K}_{FU}^\top \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}(\mathbf{y}_{\text{vec}} - \mathbf{m}_{\mathbf{w},\text{vec}}),
\end{aligned}
$$

where we define $\mathbf{P}_U = \mathbf{K}_{UU} + \mathbf{K}_{FU}^\top \boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}\mathbf{K}_{FU}$. Finally, we obtain the posterior predictive distribution over $f_\star$ as follows:

$$
p_W(f_\star|\mathbf{x}_\star, Y, X) = \int p_W(f_\star|\mathbf{x}_\star, \mathbf{u}_{\text{vec}}) q_W(\mathbf{u}_{\text{vec}}) d\mathbf{u}_{\text{vec}}.
$$

Here, we can use the integral obtained in Appendix B.1 to get the following:

$$
\begin{aligned}
p_W(f_\star|\mathbf{x}_\star, Y, X) &= \mathcal{N}(f_\star; \boldsymbol{\mu}_\star^S, \boldsymbol{\Sigma}_\star^S), \\
\boldsymbol{\mu}_\star^S &= \mathbf{k}_{U\star}^\top \mathbf{K}_{UU}^{-1}\boldsymbol{\mu}_U, \\
&= \mathbf{k}_{U\star}^\top \mathbf{K}_{UU}^{-1}\mathbf{K}_{UU}\mathbf{P}_U^{-1}\mathbf{K}_{FU}^\top \boldsymbol{\nu} \\
&= \mathbf{k}_{U\star}^\top \mathbf{P}_U^{-1}\mathbf{K}_{FU}^\top \boldsymbol{\nu} \\
\boldsymbol{\Sigma}_\star^S &= \mathbf{k}_{\star\star} - \mathbf{k}_{U\star}^\top \mathbf{K}_{UU}^{-1}(\mathbf{K}_{UU} - \boldsymbol{\Sigma}_U)\mathbf{K}_{UU}^{-1}\mathbf{k}_{U\star} \\
&= \mathbf{k}_{\star\star} - \mathbf{k}_{U\star}^\top \mathbf{K}_{UU}^{-1}(\mathbf{K}_{UU} - \mathbf{K}_{UU}\mathbf{P}_U^{-1}\mathbf{K}_{UU})\mathbf{K}_{UU}^{-1}\mathbf{k}_{U\star} \\
&= \mathbf{k}_{\star\star} - \mathbf{k}_{U\star}^\top \mathbf{K}_{UU}^{-1}\mathbf{k}_{U\star} + \mathbf{k}_{U\star}^\top \mathbf{P}_U^{-1}\mathbf{k}_{U\star}.
\end{aligned}
$$

The ELBO is then obtained by substituting Ψ_W:

$$
\begin{aligned}
&\text{ELBO}(Z, \Sigma, \boldsymbol{\theta}) \\
&= \log\Bigg(\int \exp\Bigg(-\frac{1}{2}\text{Tr}\left(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{w}}^{-1/2}\boldsymbol{\Sigma}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}\mathbf{J}_{\mathbf{w}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\right) - \frac{1}{2}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^\top \mathbf{J}_{\mathbf{w}}^{-1}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}} \\
&+ \boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^\top \boldsymbol{\nu} - \frac{1}{2}C(\Sigma, \boldsymbol{\theta}) - \frac{1}{2}\mathbf{u}_{\text{vec}}^\top \mathbf{K}_{UU}^{-1}\mathbf{u}_{\text{vec}}\Bigg)(2\pi)^{-QR/2}\det(\mathbf{K}_{UU})^{-1/2} d\mathbf{u}_{\text{vec}}\Bigg).
\end{aligned}
$$

We can simplify as follows:

$$
\begin{aligned}
&\text{ELBO}(Z, \Sigma, \boldsymbol{\theta}) \\
&= -\frac{1}{2}\left(\text{Tr}\left(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{w}}^{-1/2}\boldsymbol{\Sigma}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}\mathbf{J}_{\mathbf{w}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\right) + C(\Sigma, \boldsymbol{\theta})\right) - \frac{QR}{2}\log(2\pi) - \frac{1}{2}\log(\det(\mathbf{K}_{UU})) \\
&+ \log\left(\int \exp\left(-\frac{1}{2}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^\top \mathbf{J}_{\mathbf{w}}^{-1}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}} + \boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^\top \boldsymbol{\nu} - \frac{1}{2}\mathbf{u}_{\text{vec}}^\top \mathbf{K}_{UU}^{-1}\mathbf{u}_{\text{vec}}\right) d\mathbf{u}_{\text{vec}}\right).
\end{aligned}
$$

We can integrate over $\mathbf{u}_{\text{vec}}$ in the latter term to get:

$$
\begin{aligned}
&\log\left(\int \exp\left(-\frac{1}{2}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^{\top}\mathbf{J}_{\mathbf{w}}^{-1}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}} + \boldsymbol{\mu}_{\mathbf{f}_{\text{vec}}|\mathbf{u}_{\text{vec}}}^{\top}\boldsymbol{\nu} - \frac{1}{2}\mathbf{u}_{\text{vec}}^{\top}\mathbf{K}_{UU}^{-1}\mathbf{u}_{\text{vec}}\right) d\mathbf{u}_{\text{vec}}\right) \\
&= \log\left(\int \exp\left(-\frac{1}{2}(\mathbf{u}_{\text{vec}}^{\top}\mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^{\top}\mathbf{J}_{\mathbf{w}}^{-1}\boldsymbol{\Sigma}^{-1}\mathbf{K}_{FU}\mathbf{K}_{UU}^{-1} + \mathbf{K}_{UU}^{-1})\mathbf{u}_{\text{vec}} + \mathbf{u}_{\text{vec}}^{\top}\mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^{\top}\boldsymbol{\nu}\right) d\mathbf{u}_{\text{vec}}\right) \\
&= \log\left(\frac{(2\pi)^{QR/2}\exp(\frac{1}{2}\boldsymbol{\nu}^{\top}\mathbf{K}_{FU}(\mathbf{K}_{UU} + \mathbf{K}_{FU}^{\top}\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}\mathbf{K}_{FU})^{-1}\mathbf{K}_{FU}^{\top}\boldsymbol{\nu})}{\sqrt{\det((\mathbf{K}_{UU}(\mathbf{K}_{UU} + \mathbf{K}_{FU}^{\top}\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}\mathbf{K}_{FU})^{-1}\mathbf{K}_{UU})^{-1})}}\right) \\
&= \frac{QR}{2}\log(2\pi) + \frac{1}{2}\boldsymbol{\nu}^{\top}\mathbf{K}_{FU}(\mathbf{K}_{UU} + \mathbf{K}_{FU}^{\top}\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}\mathbf{K}_{FU})^{-1}\mathbf{K}_{FU}^{\top}\boldsymbol{\nu} \\
&\qquad + \frac{1}{2}\log\left(\frac{1}{\det(\mathbf{K}_{UU}^{-1}(\mathbf{K}_{UU} + \mathbf{K}_{FU}^{\top}\boldsymbol{\Sigma}^{-1}\mathbf{J}_{\mathbf{w}}^{-1}\mathbf{K}_{FU})\mathbf{K}_{UU}^{-1})}\right) \\
&= \frac{QR}{2}\log(2\pi) + \frac{1}{2}\boldsymbol{\nu}^{\top}\mathbf{K}_{FU}\mathbf{P}_{U}^{-1}\mathbf{K}_{FU}^{\top}\boldsymbol{\nu} + \frac{1}{2}\log\left(\frac{\det(\mathbf{K}_{UU})^2}{\det(\mathbf{P}_U)}\right)
\end{aligned}
$$

This results in the final expression for the ELBO:

$$
\begin{aligned}
\text{ELBO}(Z, \Sigma, \boldsymbol{\theta}) = -\frac{1}{2}\Bigg(&\text{Tr}\left(\boldsymbol{\Sigma}^{-1/2}\mathbf{J}_{\mathbf{w}}^{-1/2}(\mathbf{K} - \mathbf{K}_{FU}\mathbf{K}_{UU}^{-1}\mathbf{K}_{FU}^{\top})\mathbf{J}_{\mathbf{w}}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\right) + C(\Sigma, \boldsymbol{\theta})\Bigg) \\
&+ \frac{1}{2}\boldsymbol{\nu}^{\top}\mathbf{K}_{FU}\mathbf{P}_{U}^{-1}\mathbf{K}_{FU}^{\top}\boldsymbol{\nu} + \frac{1}{2}\log\left(\frac{\det(\mathbf{K}_{UU})}{\det(\mathbf{P}_U)}\right).
\end{aligned}
$$

□

C Additional Experiments

The code to reproduce experiments is available at https://github.com/joshuarooijakkers/rmogp.

C.1 Implementation Details

For R-MOGPs, we use the w-LOO-CV objective as given in Section 3.3:

$$
\varphi_w(\Sigma, \boldsymbol{\theta}) = \sum_{i=1}^{N}\sum_{t=1}^{T}\left(\frac{w_{i,t}}{\beta_t}\right)^2 \log p_w(y_{i,t}|Y_{-(i,t)}, \Sigma, \boldsymbol{\theta}).
$$

We note, however, that computing the weights $w_{i,t}$ requires the hyperparameters $C_{-t,-t}^{(i)} = [\mathcal{K}(\mathbf{x}_i, \mathbf{x}_i) + \Sigma]_{-t,-t}$ and $C_{-t,t}^{(i)} = [\mathcal{K}(\mathbf{x}_i, \mathbf{x}_i)]_{-t,t}$, which we aim to learn. In principle, one could estimate the full objective jointly, but we found that doing so destabilises the optimisation. To address this, we adopt a common two-step procedure from the literature (Huber, 1981; Hampel et al., 1986): first, the weights are (robustly) estimated and then held fixed during the optimisation of the hyperparameters. Once the optimal parameters are selected, the weights are re-estimated for use during inference. Under the ICM, the hyperparameters $C_{-t,-t}^{(i)}$ and $C_{-t,t}^{(i)}$ are components of the covariance matrix C across outputs, and therefore we use the Fast Minimum Covariance Determinant (FastMCD) algorithm (Rousseeuw, 1984; Rousseeuw and Van Driessen, 1999) for weight estimation in the first step. This estimator robustly estimates the location and scatter of multivariate data by finding a subset of points whose covariance determinant is minimal.

In summary, to optimise the model parameters, we first estimate a robust covariance matrix using the FastMCD algorithm. Next, we compute the weights as described in Section 3.1 and keep them fixed throughout the optimisation. For a given set of hyperparameters, we evaluate the LOO-CV objective

for each $i \in \{1, \ldots, N\}, t \in \{1, \ldots, T\}$:

$$p_w(y_{i,t}|Y_{-(i,t)}, \boldsymbol{\theta}, \Sigma) = \mathcal{N}(\mu_{i,t}^{\text{RMOGP}}, (\sigma_{i,t}^{\text{RMOGP}})^2 + \sigma_t^2)$$

$$\mu_{i,t}^{\text{RMOGP}} = z_{i,t} + m_{i,t} - \frac{[(\mathbf{K} + \boldsymbol{\Sigma}\mathbf{J}_{\mathbf{w}})^{-1}\mathbf{z}_{\text{vec}}]_{(t-1)N+i}}{[(\mathbf{K} + \boldsymbol{\Sigma}\mathbf{J}_{\mathbf{w}})^{-1}]_{(t-1)N+i,(t-1)N+i}}$$

$$(\sigma_{i,t}^{\text{RMOGP}})^2 = \frac{1}{[(\mathbf{K} + \boldsymbol{\Sigma}\mathbf{J}_{\mathbf{w}})^{-1}]_{(t-1)N+i,(t-1)N+i}} - \frac{\sigma_t^4}{2} w_t(\mathbf{x}_i, \mathbf{y}_i)^{-2}.$$

We then compute the weighted LOO-CV objective as a sum over all i, t using weights $(w_{i,t}/\beta_t)^2$. The optimisation proceeds by iteratively updating the hyperparameters via the L-BFGS algorithm (Byrd et al., 1995) until convergence. Finally, the weights are re-estimated using the optimised covariance matrix.

C.2 Performance Metrics

To evaluate and compare model performance, we compute the root mean squared error (RMSE) across all observations and outputs:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\frac{1}{T_i}\sum_{t=1}^{T_i}(y_{i,t} - \hat{y}_{i,t})^2},$$

where $T_i \in \{1, \ldots, T\}$ denotes the number of output variables for observation $i \in \{1, \ldots, N\}$, $y_{i,t}$ is the true value, and $\hat{y}_{i,t}$ is the corresponding model prediction. The second performance metric we consider is the negative log predictive density (NLPD), which is given as

$$\text{NLPD} = -\frac{1}{N}\sum_{i=1}^{N}\frac{1}{T_i}\sum_{t=1}^{T_i}\log \mathcal{N}(y_{i,t}; \hat{y}_{i,t}, \hat{\sigma}_{i,t}^2),$$

where $\hat{\sigma}_{i,t}^2$ is the model's variance of its prediction.

C.3 Closed-form Conditional Expectations

The closed-form conditional expectations presented in Section 3.1 hold for general multi-output kernels. Most multi-output kernel constructions yield a covariance matrix $\mathbf{K} \in \mathbb{R}^{NT \times NT}$. For a given input $\mathbf{x}_i$, the sliced matrix $A^{(i)}$ is obtained by extracting entries $(i + kN, i + lN)$ for $k, l \in \{0, \ldots, T-1\}$ from $\mathbf{K}$. This is equivalent to evaluating the cross-output kernel at the same input, i.e. $\mathcal{K}(\mathbf{x}_i, \mathbf{x}_i)$. The corresponding covariance matrix is therefore $C^{(i)} = A^{(i)} + \Sigma$. We now illustrate this construction for three common multi-output kernel choices.

Intrinsic Coregionalisation Model (ICM) Under the ICM, the multi-output covariance is $\mathbf{K} = B \otimes \kappa(X, X)$, where $\kappa(X, X) \in \mathbb{R}^{N \times N}$. Applying the slicing procedure yields $[A^{(i)}]_{t,t'} = B_{t,t'}\kappa(\mathbf{x}_i, \mathbf{x}_i)$. Since $\kappa(\mathbf{x}_i, \mathbf{x}_i) = 1$ for normalised kernels, it follows that $A^{(i)} = B$ such that $C^{(i)} = B + \Sigma$.

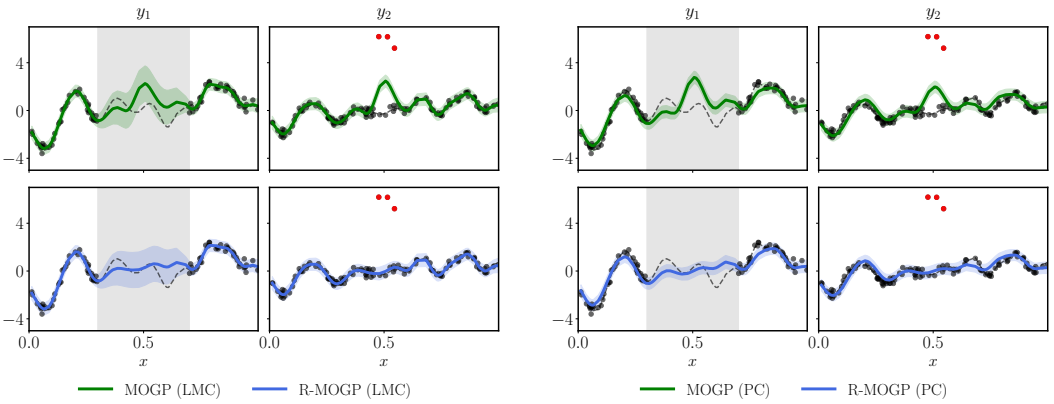
Linear Model of Coregionalisation (LMC) For the LMC, $\mathbf{K} = \sum_{g=1}^{G} B_g \otimes \kappa_g(X, X)$ for $g \in \{1, \ldots, G\}$. The sliced matrix becomes $[A^{(i)}]_{t,t'} = \sum_{g=1}^{G}[B_g]_{t,t'}\kappa_g(\mathbf{x}_i, \mathbf{x}_i) = \sum_{g=1}^{G}[B_g]_{t,t'}$, again using $\kappa_g(\mathbf{x}_i, \mathbf{x}_i) = 1$. Hence, $C^{(i)} = \sum_{g=1}^{G} B_g + \Sigma$.

Process Convolution (PC) The process convolution kernel is defined as $[\mathcal{K}(\mathbf{x}, \mathbf{x}')]_{t,t'} = \int G_t(\mathbf{x} - \mathbf{z})G_{t'}(\mathbf{x}' - \mathbf{z})d\mathbf{z}$ where G_t and $G_{t'}$ are smoothing kernels. For Gaussian MOGPs, $G_t(\mathbf{x} - \mathbf{z}) = v_t \exp(-\frac{1}{2}\lambda_t(\mathbf{x} - \mathbf{z})^2)$. The integral admits a closed form:

$$[\mathcal{K}(\mathbf{x}, \mathbf{x}')]_{t,t'} = \frac{v_t v_{t'}\sqrt{2\pi}}{\sqrt{\lambda_t + \lambda_{t'}}} \exp\left(-\frac{\lambda_t \lambda_{t'}}{2(\lambda_t + \lambda_{t'})}(\mathbf{x} - \mathbf{x}')^2\right).$$

Evaluating at $\mathbf{x} = \mathbf{x}' = \mathbf{x}_i$ gives $[A^{(i)}]_{t,t'} = v_t v_{t'} \sqrt{2\pi} / \sqrt{\lambda_t + \lambda_{t'}}$.

To illustrate that R-MOGP is also robust when using non-ICM multi-output kernels, we replicate the experiment from Figure 2, replacing the ICM kernel with the LMC and PC kernels. The resulting predictive distributions, shown in Figure 6, demonstrate that R-MOGP is likewise able to robustly impute the missing values under these alternative multi-output kernel specifications.



(a) Linear Model of Coregionalisation (LMC).

(b) Process Convolution (PC).

Figure 6: Synthetic imputation experiment using R-MOGPs based on the Linear Model of Coregionalisation (LMC) and Process Convolution (PC) frameworks.

C.4 Synthetic Multi-task Imputation Problem in Figure 2

We first uniformly at random sample $N = 120$ input locations ($d = 1$) over the interval [0,1]. The true latent functions are generated using an ICM with coregionalisation parameters $B_{11} = 2$, $B_{22} = 1$, and $B_{12} = 1.25$, and a squared exponential kernel with $\ell = 0.05$. Observations are obtained by adding Gaussian noise $\mathcal{N}(0, 0.05)$. Next, we contaminate $\epsilon_2 = 2.5\%$ of the observations in y_2 around $x = 0.5$ by replacing them with samples from $\mathcal{N}(6, 0.5)$. Finally, we remove data in y_1 where $0.3 < x < 0.7$.

In addition to the standard MOGP and our proposed R-MOGP, we include the t-MOGP and L-MOGP for a comprehensive comparison of robust methods (see Appendix C.10 for implementation details). All methods use prior mean $m_t(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} y_{i,t}$.

As illustrated in Figure 7, the predictive distributions demonstrate that R-MOGP, t-MOGP and L-MOGP effectively mitigate the influence of outliers. In contrast, the standard MOGP is significantly biased by the contaminated data, leading to poor predictive performance in both the corrupted regions and the missing data gap of y_1.

Next, we assess the reliability of uncertainty quantification for R-MOGP and MOGP using coverage plots for the same experiment, both with and without outliers, given in Figure 8. We increase the sample size to $N = 1000$ to obtain more stable coverage estimates. In the absence of outliers, the two methods achieve very similar coverage. In the presence of outliers, R-MOGP maintains coverage comparable to that in the uncorrupted setting, whereas standard MOGP exhibits substantially lower coverage.

C.5 Synthetic MOGP Problem from Section 4

We first uniformly sample $N = 100$ input locations ($d = 1$) over the interval $[-5, 5]$. Observations are then generated across $T = 3$ outputs from an ICM with a squared exponential kernel of lengthscale $\ell = 1$, and the coregionalisation matrix

$$B = \begin{bmatrix} 1.0 & 0.9 & 0.7 \\ 0.9 & 1.0 & 0.8 \\ 0.7 & 0.8 & 1.0 \end{bmatrix}.$$

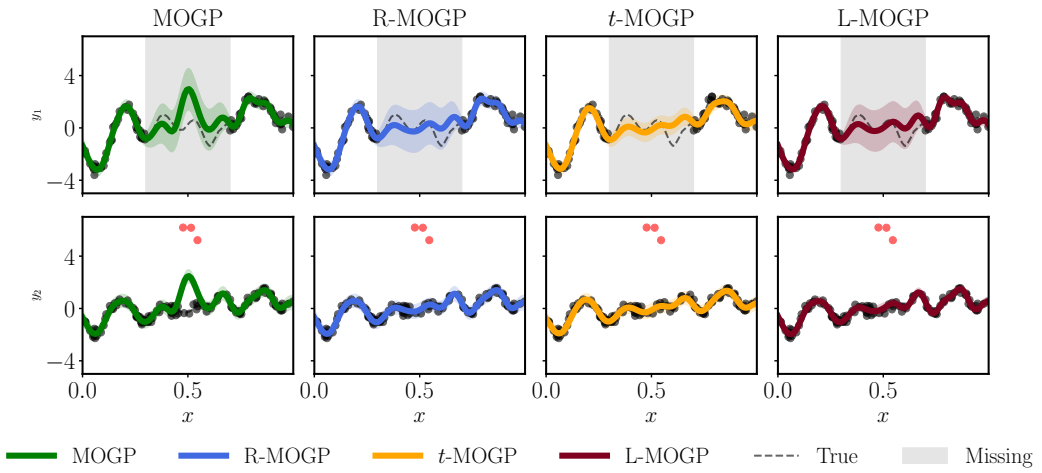


Figure 7: Synthetic imputation experiment for MOGP, R-MOGP, t-MOGP, and L-MOGP.

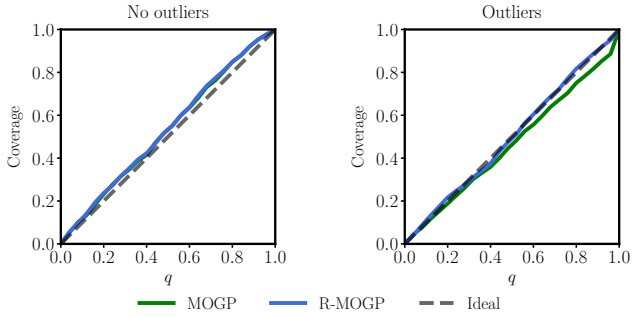


Figure 8: Coverage plots for MOGP and R-MOGP in the synthetic imputation experiment with $N = 1000$, with and without outliers. Coverage is the proportion of uncontaminated observations that fall within predictive intervals with nominal coverage q.

Gaussian noise $\varepsilon_{i,t} \sim \mathcal{N}(0, 0.1^2)$ is added to all outputs $t \in \{1, \ldots, 3\}$. This results in the dataset we use in the well-specified setting. In the outlier-contaminated setting, we introduce $\epsilon_1 = 10\%$ contamination by randomly selecting observations from the first output ($t = 1$) and perturbing them with outliers. Specifically, for each selected observation, we add a random draw from $U[2, 3]$ if $y_{i,t} < m_t(\mathbf{x}_i)$, and subtract it otherwise.

For the R-MOGP (w_{RMOGP}), we use the closed-form expression in Proposition 3.1 with the weights proposed in Section 3.1. For the R-MOGP (w_{RCGP}), we use the closed-form expression in Proposition 3.1 where we use the weight equation defined in (7). However, we now use that $\gamma_t(\mathbf{x}, \mathbf{y}) = m_t(\mathbf{x})$, and $c_t(\mathbf{x}, \mathbf{y})$ is chosen as the $(1 - \epsilon_t)$-th quantile of $\{|y_{i,t} - m_t(\mathbf{x}_i)|\}_{i=1}^N$. For all methods, the prior mean is taken as $m_t(\mathbf{x}) = 0$. Both R-MOGP variants use the optimisation objective described in Section 3.3, each with their respective weights. We repeat the experiment 20 times, each with a unique random seed.

C.6 Source of Robustness in R-MOGP

The R-MOGP posterior in Proposition 3.1 differs from the standard MOGP posterior in (5) in two respects: the identity matrix I_{NT} is replaced by the weight-dependent matrix $\mathbf{J}_{\mathbf{W}}$, and the prior mean $\mathbf{m}_{\text{vec}}$ is replaced by its weighted counterpart $\mathbf{m}_{\mathbf{W},\text{vec}}$. The matrix modification acts as a noise term, treating outliers as noisier observations, while the mean modification acts as a shrinkage term, trading a small increase in bias for a reduction in variance (Altamirano et al., 2024). In Figure 9, we isolate the effects of these two terms across varying values of ϵ_2. The noise term is essential to robustness: removing it causes R-MOGP to lose robustness. By contrast, R-MOGP is relatively insensitive to

changes in the shrinkage term. The results also show that poor calibration of ϵ_2 can compromise robustness.

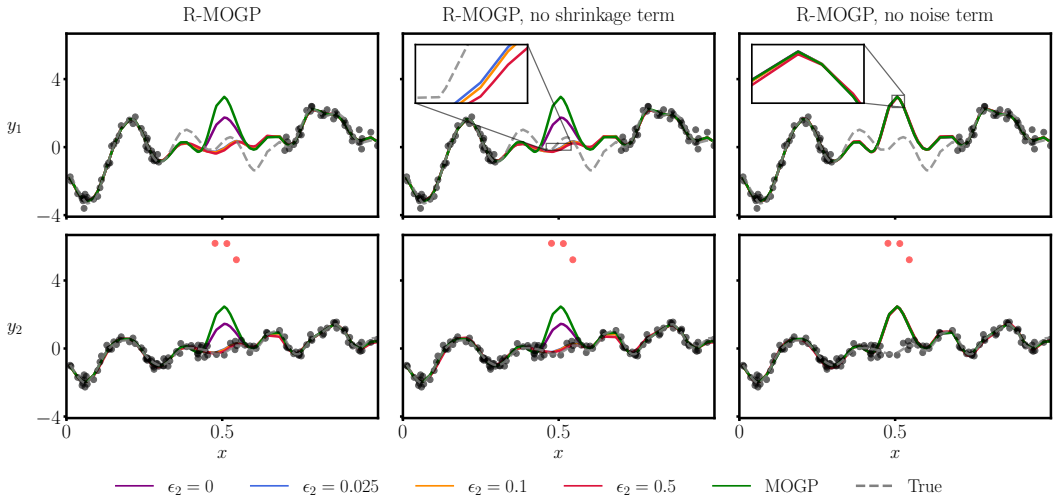


Figure 9: Posterior mean of R-MOGP (left), R-MOGP without shrinkage term (middle) and R-MOGP without noise term (right) for varying values of ϵ_2 on the synthetic imputation experiment.

C.7 Effect of Weights and Objectives on Robustness

In R-MOGP, robustness is achieved through both the choice of weight (Section 3.1) and the robust hyperparameter optimisation objective (Section 3.3). We assess the impact of these two factors by evaluating models under different weight schemes and optimisation objectives. To this end, we use the same synthetic dataset as in Appendix C.5. We compare MOGP, R-MOGP (w_{RCGP}), and R-MOGP (w_{RMOGP}) using various weightings in the LOO-CV objective: no weights (i.e. all equal to 1), linear weights, and squared weights. For MOGP, we adopt the initial R-MOGP weights. Results are presented in Table 3.

When using the unweighted objective, R-MOGP gains little over MOGP. This is expected: the unweighted LOO objective is essentially equivalent to maximising the standard log marginal likelihood, which is well known to be non-robust to outliers. Under such an objective, the hyperparameters are strongly influenced by the outliers, such that our proposed likelihood cannot express its intended robustness. Conversely, simply applying (squared) weighted optimisation to a standard MOGP does not yield robustness either: for prediction, the MOGP closed form is not robust in presence of model misspecification (as proven in Lemma B.4). Therefore, robustness arises only from the combination of (i) the robust R-MOGP formulation and (ii) a robust hyperparameter objective. Finally, in the absence of outliers, all methods perform equally well regardless of weighting. This is expected, as all observations receive equal weights, making the objectives effectively equivalent.

Table 3: Comparison of MOGP and R-MOGP (w_{RCGP}) and R-MOGP (w_{RMOGP}) under different LOO-CV weighting schemes (unweighted, weighted, squared weights). The results show that R-MOGP (w_{RMOGP}) combined with a squared weighted hyperparameter optimisation objective is most robust to outliers.

Outliers	**Method**	**Unweighted**		**Weighted**		**Squared**	
		RMSE	NLPD	RMSE	NLPD	RMSE	NLPD
✓	MOGP	0.20 ± 0.03	-0.37 ± 0.04	0.18 ± 0.03	-0.48 ± 0.05	0.18 ± 0.03	-0.52 ± 0.07
	R-MOGP (w_{RCGP})	0.19 ± 0.03	-0.38 ± 0.04	0.19 ± 0.04	-0.40 ± 0.04	0.18 ± 0.04	-0.44 ± 0.07
	R-MOGP (w_{RMOGP})	0.18 ± 0.04	-0.38 ± 0.04	0.13 ± 0.03	-0.53 ± 0.06	0.11 ± 0.01	-0.67 ± 0.08
✗	MOGP	0.093 ± 0.003	-0.95 ± 0.04	0.093 ± 0.004	-0.95 ± 0.04	0.093 ± 0.004	-0.95 ± 0.04
	R-MOGP (w_{RCGP})	0.093 ± 0.003	-0.95 ± 0.03	0.093 ± 0.004	-0.94 ± 0.04	0.093 ± 0.004	-0.94 ± 0.04
	R-MOGP (w_{RMOGP})	0.093 ± 0.004	-0.95 ± 0.04	0.093 ± 0.004	-0.95 ± 0.04	0.093 ± 0.004	-0.95 ± 0.04

C.8 Choice of the Weight Function

In Section 3.1 and Table 1, we discussed how the naive RCGP weight used in Altamirano et al. (2024) is suboptimal, as it is sensitive to prior mean misspecification. To illustrate this and demonstrate the improvement afforded by our proposed weight, we consider the setup in Figure 10. We first uniformly generate input values over the interval $[0, 1]$ and then sample $N = 120$ data points across $T = 2$ outputs from an ICM with $B_{11} = 2$, $B_{22} = 1$, $B_{12} = 1.25$, and a squared exponential kernel with lengthscale $\ell = 0.1$. Gaussian noise $\mathcal{N}(0, 0.05)$ is added to the observations. These hyperparameters are used for predictions across all models. For the first output ($t = 1$), we introduce outliers corresponding to $\epsilon_1 = 10\%$ of N at around $x = 0.5$, and the corresponding observations are replaced by samples from $\mathcal{N}(2.5, 0.5)$.

We generate predictions from three models: the standard MOGP, the R-MOGP with the RCGP weight w_{RCGP} from Altamirano et al. (2024), and the R-MOGP with our proposed weight w_{RMOGP} (see Section 3.1). Note that R-MOGP (w_{RCGP}) uses the same multi-output w_{RCGP} as detailed in Appendix C.5. Across all models, we use the prior mean $m_t(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} y_{i,t}$.

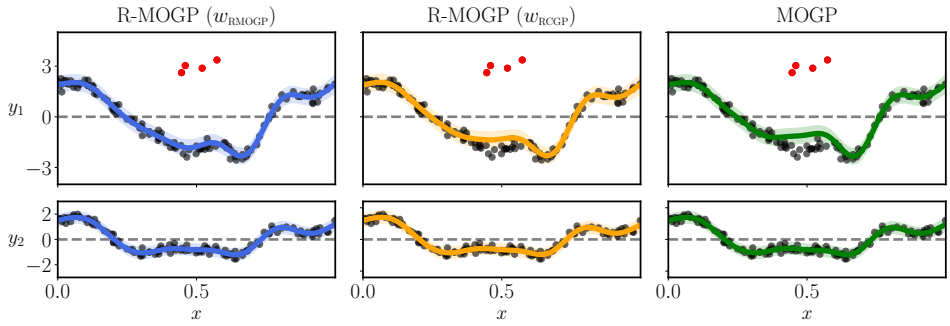


Figure 10: Effect of the weight function on the robustness of R-MOGP. R-MOGP (w_{RMOGP}) uses the weight function described in this paper, while the R-MOGP (w_{RCGP}) use the original RCGP weight with $\gamma_t(\mathbf{x}, \mathbf{y}) = m_t(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} y_{i,t}$. Among all methods, only R-MOGP (w_{RMOGP}) is robust to the contamination.

The plotted predictives show that both the standard MOGP and the R-MOGP with w_{RCGP} lack robustness: the former due to its inability to downweight observations, and the latter because outliers and data are at the same distance from the chosen prior, so both receive the same weight. In contrast, the R-MOGP with the proposed weight w_{RMOGP} shows robustness by effectively sharing information across outputs, since the centering function is taken to be the conditional expectation given the observations in the other output channel.

C.9 Performance Comparison for Multivariate Outliers

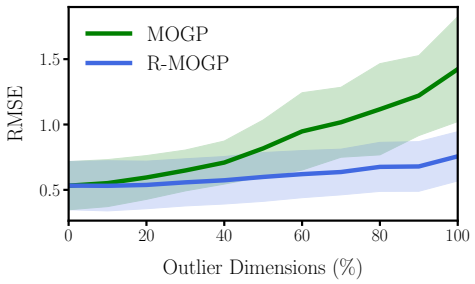
In many settings, outliers occur jointly, affecting multiple observed values for a given input (see, e.g. Leys et al., 2018). In this section, we analyse how the performance of both MOGP and R-MOGP changes as the number of jointly outlying observations in the observed vector increases. To ensure joint contamination, we introduce correlated outliers such that the Mahalanobis distance between the contaminated and uncontaminated points is large (set-up described below). This approach reflects realistic conditions in which outliers tend to be correlated (Rousseeuw and Hubert, 2011), and the Mahalanobis distance ensures that the joint deviation is large relative to the covariance structure.

Specifically, we first uniformly at random generate $N = 80$ input values over the interval $[0, 1]$. We then sample $T = 10$ outputs from an ICM with coregionalisation matrix B, where $B_{ij} = 1 - 0.1|i-j|$, and a squared exponential kernel with $\ell = 0.1$. Gaussian noise $\mathcal{N}(0, 0.05)$ is then added to each data point. These parameters are used for predictions in both models.

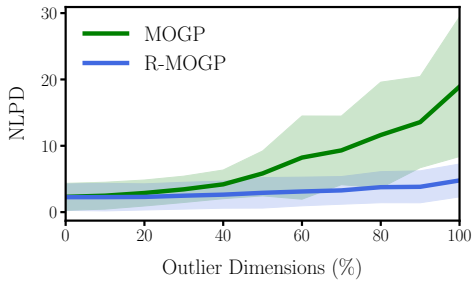
To generate the outliers, we uniformly at random select 10% of the input values for contamination. For a specified number of outputs $S \in \{1, \ldots, T\}$ to be jointly outlying, we add perturbations with large Mahalanobis distance, following Mardia et al. (2024). Specifically, for a given target Mahalanobis distance $\text{MD}_{\text{target}}$, and a base vector $\mathbf{v} \in \mathbb{R}^S$, we compute the Mahalanobis norm $q = \mathbf{v}^\top B^{-1}\mathbf{v}$.

We then determine the scaling factor $\alpha = \text{MD}_{\text{target}}/\sqrt{q}$, which yields the final outlier vector $\alpha\mathbf{v}$ with desired Mahalanobis distance $\text{MD}_{\text{target}}$. This perturbation is multiplied by either -1 or 1 with equal probability, before being added to the first S elements of the selected output vector. For each input, $\text{MD}_{\text{target}}$ is sampled from the univariate uniform distribution $U[10, 15]$, and the base vector is drawn from an S-variate uniform distribution $U[0.5, 1.5]$ to ensure correlation among the outliers. We set the prior mean as $m_t(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} y_{i,t}$ for $t \in \{1, \ldots, T\}$. The experiment is repeated for $S \in \{1, \ldots, T\}$, with 20 repetitions per S, each using a unique seed.

The results are displayed in Figure 11. As the dimensionality of the outliers increases, both RMSE and NLPD grow much faster for MOGP than for R-MOGP. Thus, R-MOGP remains more robust even when outliers occur in higher dimensions.



(a) Mean RMSE and one standard deviation across 20 random seeds versus the percentage of dimensions in which outliers occur.



(b) Mean NLPD and one standard deviation across 20 random seeds versus the percentage of dimensions in which outliers occur.

Figure 11: Performance of MOGP and R-MOGP as the number of outliers in the observed vector increases. We generate $N = 80$ data points from an ICM across $T = 10$ outputs. We then contaminate $\epsilon_s = 10\%$ of the observations with multivariate outliers for $s \in \{1, \ldots, S\}$ with $S \leq T$ increasing incrementally. The prior mean is $m_t(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} y_{i,t}$. As the percentage of outliers grows, the performance of the MOGP drops substantially faster than that of the R-MOGP.

C.10 Energy Efficiency Prediction

Dataset We obtain data from the ENERGY EFFICIENCY dataset from the UCI repository, which can be downloaded at `https://archive.ics.uci.edu/dataset/242/energy+efficiency`. The dataset was generated using building-energy simulation in ECOTECT for 12 prototype residential building shapes, systematically varied design parameters, and was designed to assess the energy efficiency of buildings based on various architectural characteristics. Each data point represents a building and consists of $d = 8$ covariates: relative compactness, surface area, wall area, roof area, overall height, orientation, glazing area and glazing area distribution. The goal is to predict $T = 2$ outputs: heating load ($t = 1$) and cooling load ($t = 2$).

We standardise both the input and output variables by subtracting their own mean and then dividing by their own standard deviation. The dataset comprises $N = 768$ data points, which we randomly split into a training set (75%) and a test set (25%) for model evaluation. This comprises the setup for the scenario with no outliers.

Outlier Scenarios We uniformly at random introduce outliers into $\epsilon_1 = 10\%$ of the heating load observations using the following contamination strategies, adapted from Altamirano et al. (2024):

- *Uniform outliers*: A specified proportion of data points (after standardisation) is uniformly at random chosen for contamination. This subset is then split evenly: for half the points, we add noise sampled from $z \sim U(6, 9)$; for the other half, we subtract it. Here, U represents the uniform distribution.
- *Asymmetric outliers*: We uniformly sample a fixed proportion of the dataset (after standardisation), as with uniform outliers. However, instead of splitting, we contaminate all selected points by adding $z \sim U(6, 9)$, introducing a directional bias in the outliers.

- *Student-t outliers*: A specified proportion of data points (after standardisation) is uniformly at random chosen for contamination. We then add noise sampled from $z \sim t_3(0, 5)$ to these data points. Here, $t_\nu(\mu, \sigma)$ represents the Student-t distribution with location parameter μ, scale parameter σ, and ν degrees of freedom.

Implementation The experiments were run on a 2025 HP ZBook Power G11 Mobile Workstation PC with an Intel(R) Core(TM) Ultra 7 155H CPU with 16GB memory.

We compare the MOGP and R-MOGP models against the t-MOGP and L-MOGP, which extend the MOGP by using a Student-t and Laplace likelihood, respectively. This likelihood breaks conjugacy, so the posterior and predictive distributions are no longer available in closed form and must instead be approximated. Our implementation is in `GPflow` (Matthews et al., 2017), which uses the variational inference method of Opper and Archambeau (2009) to estimate the posterior, predictive distribution, and hyperparameters. However, the package only provides a built-in implementation for scale parameters shared across outputs, which we adopt. To ensure fairness, we also fix $\sigma_t = \sigma$ across the MOGP and R-MOGP models. For all outlier scenarios (including the no-outlier case), we set $\nu = 3$, the default in `GPflow`. We found that larger values of ν consistently degrade performance in outlier-based scenarios.

For a fair comparison, parameter selection for both MOGP and R-MOGP is performed using the (w)-LOO-CV objective. The methods are implemented in `TensorFlow` (Abadi et al., 2016), the same underlying framework `GPflow` uses. We use a prior mean function with $m_t(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} y_{i,t}$ for $t \in \{1, 2\}$. Experiments are repeated 10 times with different train-test splits. The computation times illustrated in Figure 4 are based on these 40 runs (10 per outlier scenario).

Finally, we compare R-MOGP with several baseline machine learning methods to assess its robustness in a broader context. We include a standard MLP with three hidden layers of 64 neurons each and two regression heads corresponding to the $T = 2$ outputs. Since this MLP does not provide predictive uncertainty estimates, we also consider a Bayesian neural network with the same architecture, trained using variational inference. We further include XGBoost, training a separate model for each output. These comparisons should be interpreted in light of the different modelling assumptions and trade-offs associated with each method. Gaussian processes provide a continuous probabilistic model, while methods such as XGBoost do not directly provide predictive distributions. Moreover, the performance of each baseline depends on methodological and hyperparameter choices, such as tree depth and boosting parameters in XGBoost.

The results in Table 4 show that, in the absence of outliers, the MLP and XGBoost achieve lower RMSE than both MOGP and R-MOGP. Under all three outlier settings, however, R-MOGP achieves the lowest RMSE among the methods considered and the lowest NLPD among those providing predictive uncertainty estimates.

C.11 Navitoclax Dose-Response Modelling

We obtain data for the experimental drug Navitoclax from the Genomics of Drug Sensitivity in Cancer Database (Yang et al., 2013). Specifically, we used raw bulk data from the GDSC2 screening campaign (conducted around 2016–2017), available at `https://www.cancerrxgene.org/downloads/bulk_download`. Each record in the dataset contains the cell line identifier, drug name, dose, and corresponding viability measurement. For our analysis, we selected the cell lines P12-ICHIKAWA and ARH-77.

To ensure comparability across experiments, the viability scores were computed by normalising treated wells to plate-specific positive and negative controls using the protocol described in Vis et al. (2016) and implementation in the R package `gdscIC50` (Lightfoot et al., 2023). No batch correction was applied, and viability scores outside the interval $[0, 1]$ were retained without truncation. Technical replicates were averaged to yield a single `(cell line, drug, dose, viability)` observation for each unique combination.

For the MOGP, hyperparameters are selected by maximising the marginal likelihood, whereas for the R-MOGP, we use the w-LOO-CV objective described in Section 3.3. In addition, we include the t-MOGP and L-MOGP for a comprehensive comparison of robust methods (see Appendix C.10 for implementation details). In all models, we set the prior mean to $m_t(\mathbf{x}) = 0.5$ for $t \in \{1, 2\}$, reflecting a neutral prior belief halfway between cells being alive (1.0) and dead (0.0). The plots for

Table 4: RMSE and NLPD for MOGP, R-MOGP, and baseline machine learning methods on the ENERGY EFFICIENCY dataset under different outlier settings.

Outliers	Method	RMSE	NLPD
None	MOGP	0.12 (0.01)	−0.92 (0.08)
	R-MOGP	0.12 (0.01)	−0.88 (0.07)
	MLP	0.10 (0.004)	–
	BNN	0.21 (0.01)	0.18 (0.13)
	XGBoost	0.083 (0.01)	–
Uniform	MOGP	2.27 (0.29)	1.26 (0.06)
	R-MOGP	0.17 (0.02)	−0.25 (0.05)
	MLP	1.14 (0.13)	–
	BNN	0.36 (0.09)	1.29 (0.25)
	XGBoost	0.97 (0.04)	–
Asymmetric	MOGP	2.18 (0.18)	1.24 (0.06)
	R-MOGP	0.16 (0.02)	−0.31 (0.07)
	MLP	1.11 (0.10)	–
	BNN	0.68 (0.02)	1.98 (0.12)
	XGBoost	1.06 (0.07)	–
Student-t	MOGP	0.19 (0.02)	−0.03 (0.10)
	R-MOGP	0.14 (0.01)	−0.59 (0.04)
	MLP	0.23 (0.02)	–
	BNN	0.24 (0.02)	0.70 (0.11)
	XGBoost	0.19 (0.03)	–

each method are shown in Figure 12 and show that the MOGP predictions are clearly distorted by the outlier, while R-MOGP, t-MOGP and L-MOGP remain robust.

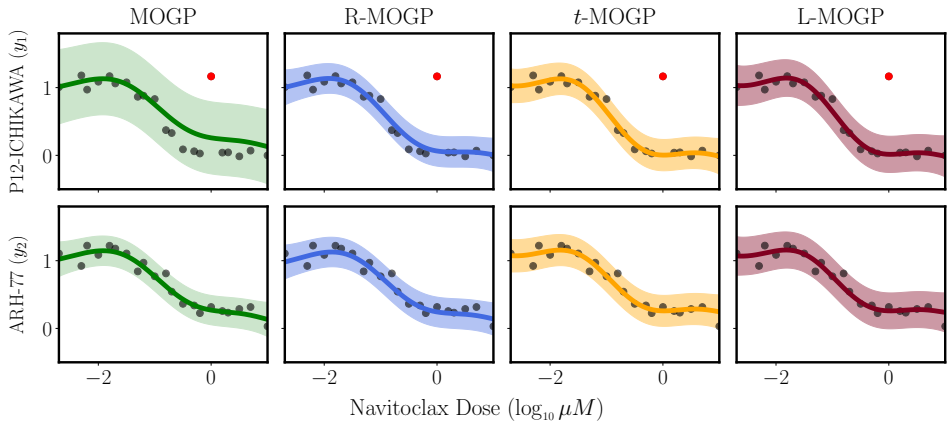


Figure 12: Navitoclax dose-response analysis with outlier in separate plots for MOGP, R-MOGP, t-MOGP and L-MOGP.

In addition, we quantitatively evaluate all methods using the in-sample RMSE and NLPD, reported in Table 5. The results confirm that MOGP breaks down in the presence of the outlier, whereas R-MOGP, t-MOGP, and L-MOGP approach the performance of MOGP without the outlier. Among these robust alternatives, only R-MOGP achieves a runtime of the same order as the conjugate MOGP.

C.12 Electricity Load Forecasting

We use data from Open Power System Data (Wiese et al., 2019), consisting of country-level electricity load time series from January 2010 to September 2020. The original hourly observations are

Table 5: Predictive performance and runtime of MOGP variants on the Navitoclax experiment.

Method	RMSE	NLPD	Runtime (s)
MOGP (no outliers)	0.066 (0.012)	−1.23 (0.11)	0.048
MOGP	0.170 (0.020)	−0.21 (0.05)	0.049
R-MOGP	0.081 (0.011)	−1.00 (0.07)	0.040
t-MOGP	0.067 (0.013)	−1.06 (0.06)	3.820
L-MOGP	0.068 (0.013)	−0.65 (0.73)	1.060

aggregated to daily frequency, yielding 3,795 data points. Our sample covers $T = 5$ Northern European countries: Denmark, Estonia, Finland, Norway, and Sweden.

We standardise both the input and output variables by subtracting their own mean and dividing by their own standard deviation. The dataset is randomly split into a training set (90%) and a test set (10%), yielding $N = 3{,}415$ data points. From this training set, we draw random subsamples of sizes [600, 1000, 1400, 1800, 2200, 2600, 3000] data points without replacement. For each subset, we train both SVMOGP and R-SVMOGP models.

To evaluate robustness, we contaminate each subset by replacing a fraction $\epsilon_1 = 0.05$ of the observations—randomly selected from the 40th to 60th percentile—with samples drawn from $\mathcal{N}(9, 0.2)$ for the Danish electricity load ($t = 1$).

Both SVMOGP and R-SVMOGP are trained with $R = 1$ latent process and $Q = \sqrt{N}$ inducing points per output. Model parameters are optimised using the L-BFGS algorithm, and are both implemented in `TensorFlow`. We then report total runtimes, including optimisation and prediction.

C.13 Robustness to Financial Anomalies

As an additional experiment, we analyse the to-be-announced (TBA) market. In this market, buyers and sellers trade mortgage-backed securities (MBS)—a collection of thousands of similar home loans—without specifying the exact underlying pool of mortgages until shortly before a defined settlement date. In the TBA market, transaction-level data are often highly correlated between securities as MBS tend to be constructed from similar pools of mortgages and are affected by the same economic factors (Boyarchenko et al., 2019). While the modelling problems in the TBA market are ideally suited to MOGPs (de Wolff et al., 2020), outliers are extremely common, particularly near the settlement date (Vickery and Wright, 2013). Some key reasons for this are that certain trades are priced much lower than the going rate because sellers can deliver slightly weaker collateral at the same price. Conversely, other trades are priced unusually high when specific collateral is scarce and commands a temporary premium.

We use transaction-level Fannie Mae Conventional Loan-backed MBS trading data, obtained from FINRA. It consists of $N = 139$ transactions observed on September 8 and 9, 2025, for $T = 3$ correlated uniform MBS coupons: 5% MBS, 5.5% MBS, and 6% MBS. We discretise the series at a 1-minute granularity and, when multiple trades occur within the same minute, we retain the last trade observed in that minute. In addition, we standardise both the input and output variables by subtracting their own mean and then dividing by their own standard deviation. We also set $\epsilon_t = 5/N$, expecting 5 contaminated data points per output. We fit both the MOGP and R-MOGP with a prior mean of $m_t(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} y_{i,t}$ for $t \in \{1, 2, 3\}$. The MOGP uses marginal likelihood for optimisation, and the R-MOGP uses the w-LOO-CV objective defined in Section 3.3. To reflect realistic trading conditions, we fix the noise variance in all models at a quarter tick, following common practice in liquid markets to avoid overestimating noise and to ensure stable, realistic predictions (Gonzalvez et al., 2019).

As can be seen in Figure 13, the data shows a clear anomalously high trade being visible in the MBS with 5.5% on September 8 around 4 pm. This outlier influences not only the predictions for the MBS with 5.5%, but also for the MBS with 6%, demonstrating the sensitivity of MOGP to such outliers across different output channels. In contrast, R-MOGP remains stable and unaffected in all three output channels.

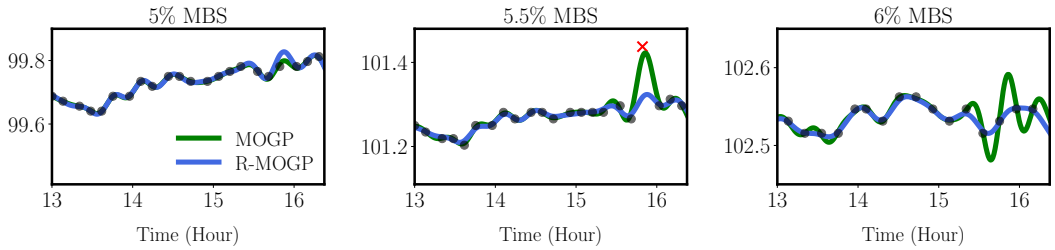


Figure 13: Robustness of MOGP and R-MOGP in MBS trades. The outlier at 4 pm in the 5.5% MBS distorts the MOGP predictions for that output and propagates errors to the 6% MBS. In contrast, R-MOGP is robust.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims in the abstract and introduction (e.g. the enumeration at the end) are conclusions from the subsequent sections.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We do not include a dedicated section on limitations; however, we discuss several limitations in the appendix (e.g. the case of multivariate outliers).

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: For each proposition and lemma, we start by stating the assumptions, which are also further elaborated on in the main text.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have included all necessary information to replicate the experimental results, with further details provided in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have included the data and code as supplementary material.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We have included all details on training, testing and optimising for each method to understand the experimental results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report standard deviations across multiple runs in the tables and include error bars in the figures.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide sufficient information on the compute resources in Section C.10.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research presented in this paper fully conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the potential societal impact of the cancer dose-response experiment in Section 4.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not pose risks for misuse.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The creators and owners of used code and data are properly cited in the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: New assets are well documented and provided in the supplementary material.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this paper does not involve LLMs as any important components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.