Robust and Conjugate Gaussian Process Regression

Matias Altamirano [1] **François-Xavier Briol** [1] **Jeremias Knoblauch** [1]

Abstract

To enable closed form conditioning, a common assumption in Gaussian process (GP) regression is independent and identically distributed Gaussian observation noise. This strong and simplistic assumption is often violated in practice, which leads to unreliable inferences and uncertainty quantification. Unfortunately, existing methods for robustifying GPs break closed-form conditioning, which makes them less attractive to practitioners and significantly more computationally expensive. In this paper, we demonstrate how to perform provably robust and conjugate Gaussian process (RCGP) regression at virtually no additional cost using generalised Bayesian inference. RCGP is particularly versatile as it enables exact conjugate closed form updates in all settings where standard GPs admit them. To demonstrate its strong empirical performance, we deploy RCGP for problems ranging from Bayesian optimisation to sparse variational Gaussian processes.

1. Introduction

GPs (Rasmussen & Williams, 2006) are one of the most widely used methods for Bayesian inference on latent functions, especially when uncertainty is required. They have numerous appealing properties, including that the prior is relatively interpretable and can be elicited through a choice of mean and covariance functions, as well as the fact that they have closed form posteriors under Gaussian likelihoods. Their convergence is also well understood, even under prior misspecification (Wynne et al., 2021). Thanks to these advantages, GPs have found applications in diverse problems including single- and multi-output regression (Bonilla et al., 2007; Moreno-Muñoz et al., 2018), emulation of expensive simulators (Santner et al., 2018), Bayesian optimisation (Shahriari et al., 2015; Garnett, 2021) and Bayesian deep learning (Damianou & Lawrence, 2013; Salimbeni et al., 2019; Dutordoir et al., 2020). Their use is enabled by a plethora of packages including `GPflow` (Matthews et al., 2017) `GPyTorch` (Gardner et al., 2018), `BoTorch` (Balandat et al., 2020), `ProbNum` (Wenger et al., 2021) and `emukit` (Paleyes et al., 2023).

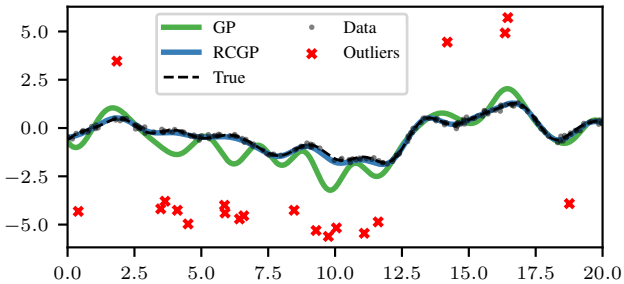

Figure 1. The posterior predictive mean of a GP (**green**) and the RCGP (**blue**) on a synthetic dataset where 10% of the data are uniformly generated outliers. Unlike the RCGP, the GP is adversely affected.

By far the most common use of GPs is in regression. Here, the observations correspond to noisy realisations from an unknown latent function that is assumed to be drawn from a GP prior. To obtain a conjugate GP posterior distribution on the latent function, the observation noise is usually assumed to be Gaussian. While assuming Gaussian observation noise makes the posterior tractable, it also makes inferences non-robust. In particular, Gaussian noise makes GPs highly susceptible to extreme values, heterogeneities, and outliers. This is illustrated in Figure 1 on a synthetic dataset corrupted with outliers: The standard GP is adversely affected, leading to considerable deviations between the inferred function and the ground truth. In many real-world applications and data sets, the presence of outliers is almost inevitable. They can occur for a variety of different reasons, including due to faulty measurements, broken sensors, extreme weather events, stock market sell-offs, or genetic mutations.

Existing Work The lack of robustness in GPs is a well-known fundamental challenge for their widespread application, and a number of methods have been proposed to address this. Broadly, these fall into two categories. The first replaces the Gaussian measurement error with more

[*]Equal contribution [1]Department of Statistical Science, University College London, London, United Kingdom. Correspondence to: Matias Altamirano <matias.altamirano.22@ucl.ac.uk>, François-Xavier Briol <f.briol@ucl.ac.uk>, Jeremias Knoblauch <j.knoblauch@ucl.ac.uk>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

heavy-tailed error distributions such as Student's t (Jylänki et al., 2011; Ranjan et al., 2016), Laplace (Kuss, 2006), Huber densities (Algikar & Mili, 2023), data-dependent noise (Goldberg et al., 1997), or mixture distributions (Naish-Guzman & Holden, 2007; Stegle et al., 2008; Daemi et al., 2019; Lu et al., 2023). Heavy tails allow these distributions to better accommodate outliers, rendering them more robust to corruptions. Their main limitation lies in their computational cost, as abandoning Gaussian noise nullifies one key advantage of GPs: conjugacy. As a consequence, these techniques rely on approximations via variational methods or Markov chain Monte Carlo. This decreases their accuracy while increasing computational costs. The second set of approaches consists in removing outlying observations before using a standard GP with Gaussian noise (Li et al., 2021; Park et al., 2022; Andrade & Takeda, 2023). While such approaches use conjugacy, it can be challenging to detect outliers in irregularly spaced data or higher dimensions. Outlier detection also tends to be computationally costly, and often requires estimating large numbers of parameters.

In this paper, we propose a new and third way to achieve robustness that uses generalised Bayesian inference (see e.g. Bissiri et al., 2016; Jewson et al., 2018; Knoblauch et al., 2022). In doing so, we significantly improve upon an earlier attempt in this direction due to Knoblauch (2019) that was applicable only for variational deep GPs, lacked closed form solutions, and was based on hyperparameters that were difficult to choose. In line with the ideas of generalised Bayesian methods, we will not modify the Gaussian noise model. Instead, we change how information is assimilated, and leverage robust loss functions instead of robust error models.

Contributions This paper proposes a novel robust and conjugate Gaussian process (RCGP) inspired by a generalised Bayesian inference scheme proposed in Altamirano et al. (2023). The posteriors rely on a generalised form of score matching (Hyvärinen, 2006; Barp et al., 2019), which effectively down-weights outlying observations. The resulting inference resolves the trade-off between robustness and computation inherent in existing methods: it is robust in the sense of Huber (1981) [Proposition 3.2] while retaining closed form solutions for both its posterior and posterior predictive [Proposition 3.1]. Additionally—and unlike other robust GPs—RCGPs can easily be plugged into various GP techniques such as sparse variational GPs (Titsias, 2009; Hensman et al., 2013) [Proposition 4.1], deep GPs (Damianou & Lawrence, 2013), multi-output GPs (Bonilla et al., 2007), and Bayesian optimisation (Shahriari et al., 2015) [Proposition 4.2]. Finally, even in settings where robustness is not required, our experiments show that RCGPs performs as well as standard GPs—raising the possibility that RCGPs may become a preferred default choice over GPs in the future.

The remainder of the paper reviews GPs and generalised Bayesian inference (Section 2), introduces RCGPs and proves their robustness (Section 3), and investigates their empirical performance and versatility for a range of experiments (Section 4).

2. Background

Our method applies the logic of generalised Bayesian posteriors to GPs. Here, we briefly explain the concepts relevant to understanding this interface.

Gaussian Processes Let $\mathbf{y} = (y_1, \dots, y_n)^\top$ denote n observations with covariates $\mathbf{x} = (x_1, \dots, x_n)^\top$, where $y_i \in \mathcal{Y} \subseteq \mathbb{R}$ and $x_i \in \mathcal{X} \subseteq \mathbb{R}^d$. While we take $\mathcal{Y} \subseteq \mathbb{R}$ for simplicity, our method can be straightforwardly generalised to multi-output regression (i.e., $\mathcal{Y} \subseteq \mathbb{R}^T$). We consider a regression setting where the noisy observations $\mathbf{y}$ come from a latent function $f : \mathcal{X} \to \mathbb{R}$:

$$y_i = f(x_i) + \varepsilon_i.$$

Here, $\boldsymbol{\varepsilon} = (\varepsilon_1, \dots, \varepsilon_n)^\top \in \mathbb{R}^n$ are independent observation errors. We place a GP prior on f, so that $f \sim \mathcal{GP}(m, k)$ with $m : \mathcal{X} \to \mathbb{R}$ and $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ being mean and kernel functions. These functions determine key properties in the draws from the GP such as differentiability, periodicity, long-range correlation or stationarity, and are parameterised by $\theta \in \Theta \subseteq \mathbb{R}^p$. Throughout, we write $\mathbf{f} = (f(x_1), \dots, f(x_n))^\top$, and use the fact that the GP prior implies the Gaussian prior $p(\mathbf{f}|\mathbf{x}) = \mathcal{N}(\mathbf{f}; \mathbf{m}, K)$, where K is the matrix with $K_{ij} = k(x_i, x_j)$ and $\mathbf{m} = (m(x_1), \dots, m(x_n))^\top$.

Finally, while there are various options for modelling observation error, almost all of them break conjugacy. The main exception is the choice $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \sigma^2 I_n)$ where I_n is an $n \times n$ identity matrix (or equivalently $p(\mathbf{y}|\mathbf{f}, \mathbf{x}) = \mathcal{N}(\mathbf{y}; \mathbf{f}, \sigma^2 I_n)$). This leads to the posterior

$$\begin{aligned} p(\mathbf{f}|\mathbf{y}, \mathbf{x}) &= \mathcal{N}(\mathbf{f}; \mu, \Sigma), \\ \mu &= \mathbf{m} + K(K + \sigma^2 I_n)^{-1}(\mathbf{y} - \mathbf{m}), \\ \Sigma &= K(K + \sigma^2 I_n)^{-1}\sigma^2 I_n. \end{aligned}$$

A key quantity of interest is then the posterior predictive over $f_\star$, the value $f(x_\star)$ at a new point $x_\star \in \mathcal{X}$:

$$\begin{aligned} p(f_\star|x_\star, \mathbf{x}, \mathbf{y}) &= \int_{\mathbb{R}^n} p(f_\star|x_\star, \mathbf{f}, \mathbf{x}, \mathbf{y}) p(\mathbf{f}|\mathbf{y}, \mathbf{x}) d\mathbf{f} \\ &= \mathcal{N}(f_\star; \mu_\star, \Sigma_\star), \\ \mu_\star &= m_\star + \mathbf{k}_\star^\top (K + \sigma^2 I_n)^{-1}(\mathbf{y} - \mathbf{m}), \\ \Sigma_\star &= k_{\star\star} - \mathbf{k}_\star^\top (K + \sigma^2 I_n)^{-1}\mathbf{k}_\star, \end{aligned}$$

for $\mathbf{k}_\star = (k(x_\star, x_1), \dots, k(x_\star, x_n))^\top$, $k_{\star\star} = k(x_\star, x_\star)$, $m_\star = m(x_\star)$. Though the posterior also depends on θ and

σ^2, we omit this for brevity. Commonly, they are chosen to maximise the marginal likelihood

$$p(\mathbf{y}|\mathbf{x},\theta,\sigma^2) = \int_{\mathbb{R}^n} p(\mathbf{f}|\mathbf{x},\theta,\sigma^2)p(\mathbf{y}|\mathbf{f},\mathbf{x},\theta,\sigma^2)d\mathbf{f} = \mathcal{N}(\mathbf{y};\mathbf{m},K+\sigma^2 I_n).$$

Alternatively, there are computationally efficient approaches for leave-one-out cross-validation, and computationally demanding Markov chain Monte Carlo methods for hierarchical Bayes (see Rasmussen & Williams, 2006, Section 5).

If $\mathbf{f}$ and ε are both modelled as Gaussians, GP regression is conjugate, so that posterior, posterior predictive, and marginal likelihood can all be obtained in closed forms. These operations have $O(n^3)$ computational and $O(n^2)$ storage cost, but are exact. For this reason, practitioners often model data using Gaussian errors—even when this assumption is wholly inappropriate and yields severe misspecification. Note that conjugacy also holds for GP interpolation—when ε is a Dirac measure at $(0,\dots,0)^\top$—in which case all formulae above remain correct for $\sigma = 0$. However, this assumes that the observations $\mathbf{y}$ are noise-free, which is even more susceptible to model misspecification than Gaussianity.

Generalised Bayesian Inference If a statistical model is misspecified so that the model cannot correctly describe the true data-generating mechanism, standard Bayesian updating is not the optimal way of integrating prior information with data (Zellner, 1988). Indeed, standard Bayes results in miscalibrated uncertainties and misleading inferences. In the parametric setting, a line of research has tackled this through generalised Bayesian methodology (see e.g. Grünwald, 2012; Hooker & Vidyashankar, 2014; Bissiri et al., 2016; Ghosh & Basu, 2016; Jewson et al., 2018; Miller & Dunson, 2018; Knoblauch et al., 2022; Miller, 2021; Fong et al., 2021; Jewson & Rossell, 2022; Matsubara et al., 2022b). Recently, generalised Bayesian posteriors have also been proposed for the non-parametric case (see e.g. Knoblauch, 2019; Wild et al., 2022). For regression, they take the form

$$p^L_\beta(\mathbf{f}|\mathbf{y},\mathbf{x}) \propto p(\mathbf{f}|\mathbf{x})\exp\big(-\beta n L_n(\mathbf{f},\mathbf{y},\mathbf{x})\big), \quad (1)$$

where $\propto$ denotes equality up to a multiplicative constant not depending on $\mathbf{f}$. The *learning rate* $\beta > 0$ is a scaling parameter that determines how quickly the posterior learns from data, and $L_n : \mathcal{Y}^n \times \mathcal{Y}^n \times \mathcal{X}^n \to \mathbb{R}$ is a loss connecting the data and the posited statistical model. Most choices for L_n are estimators of statistical divergences between the true data-generating process, p_0, and the model, such as the kernel Stein discrepancy (Matsubara et al., 2022b), β-divergences (Knoblauch et al., 2018), maximum mean discrepancies (Chérief-Abdellatif & Alquier, 2020), and Fisher divergences (Altamirano et al., 2023; Matsubara et al., 2022a).

The distributions in (1) are called *generalised* posteriors since they recover the standard Bayes posterior for $\beta = 1$ and $L_n(\mathbf{f},\mathbf{y},\mathbf{x}) = -\log p(\mathbf{y}|\mathbf{f},\mathbf{x})$. By instead choosing L_n to be a robust loss, generalised Bayesian inference has enhanced applications including filtering (Boustati et al., 2020; Duran-Martin et al., 2024), changepoint detection (Knoblauch et al., 2018; Altamirano et al., 2023), deep Gaussian processes (Knoblauch, 2019), doubly-intractable problems (Matsubara et al., 2022a;b) and Bayesian neural networks (Futami et al., 2018). In addition, generalised posteriors have been leveraged for computational efficiency. For instance, Matsubara et al. (2022b;a) used generalised posteriors for accelerated computation with unnormalised models in both continuous and discrete domains. Similarly, Schmon et al. (2020), Dellaporta et al. (2022), Pacchiardi & Dutta (2021), Legramanti et al. (2022), and Frazier et al. (2024) deployed them for simulation-based inference.

3. Methodology

We now present RCGPs in three steps. First we introduce our loss and explain how it ensures conjugacy. Second, we provide formal robustness guarantees. Finally, we show how to select hyperparameters.

The Loss Function Altamirano et al. (2023) present a posterior based on a generalised score matching loss (Hyvärinen, 2006; Lyu, 2009; Yu et al., 2022) due to Barp et al. (2019). The resulting posterior is provably robust, and conjugate for exponential family models. Importantly however, it is not applicable for regression settings and covariate-dependent models. To rectify this, we follow Xu et al. (2022) and leverage the tower property of expectations. Using this and denoting by $p_{0,x}$ the marginal distribution of the covariates, score matching losses in our setting lead to

$$\mathbb{E}_{X\sim p_{0,x}}\left[\mathbb{E}_{Y\sim p_0(\cdot|X)}\left[\|(s_{\text{model}} - s_{\text{truth}})(X,Y)\|_2^2\right]\right],$$

where $s_{\text{truth}}(x,y) = \nabla_y \log p_0(y|x)$ is the score function of the true data-generating conditional density, s_{model} the score function of our model. In the case of GP regression, this is $s_{\text{model}}(x,y) = \sigma^{-2}(f(x) - y)$. As in Altamirano et al. (2023), we instead use a weighted generalisation due to Barp et al. (2019):

$$\mathbb{E}_{X\sim p_{0,x}}\left[\mathbb{E}_{Y\sim p_0(\cdot|X)}\left[\|\big(w(s_{\text{model}} - s_{\text{truth}})\big)(X,Y))\|_2^2\right]\right]. \quad (2)$$

Here, $w : \mathcal{X}\times\mathcal{Y} \to \mathbb{R}\setminus\{0\}$ is a weighting function depending on both x and y, and we discuss how it should be chosen at the end of this section. Evaluating (2) would require the unknown score s_{truth}. Luckily, under mild smoothness and boundary conditions (Liu et al., 2022), we can use integration by parts to rewrite it up to a constant not depending on

f as

$$\mathbb{E}_{X\sim p_{0,x}}\left[\mathbb{E}_{Y\sim p_0(\cdot|X)}\left[\left((ws_{\text{model}})^2+2\nabla_y(w^2 s_{\text{model}})\right)(X,Y)\right]\right].$$

Importantly, this expression no longer depends on s_{truth}, and only features p_0 through an expectation. This leads to a natural estimator—and the proposed loss function—which is given by

$$L_n^w(\mathbf{f},\mathbf{y},\mathbf{x})=\frac{1}{n}\sum_{i=1}^{n}\left((ws_{\text{model}})^2+2\nabla_y(w^2 s_{\text{model}})\right)(x_i,y_i).$$

While we have motivated the loss using the marginal $p_{0,x}$ for simplicity, $p_{0,x}$ can be replaced with any other measure over $\mathbf{x}$. The active learning setting is a relevant example for this, and we use L_n^w in a Bayesian optimisation experiment in Section 4 to showcase this.

Finally, when the model is Gaussian, i.e. $p(\mathbf{y}|\mathbf{f},\mathbf{x})=\mathcal{N}(\mathbf{y};\mathbf{f},\sigma^2 I_n)$, this loss function becomes quadratic in f as follows:

$$L_n^w(\mathbf{f},\mathbf{y},\mathbf{x})=\mathbf{f}^\top A_n\mathbf{f}+b_n^\top\mathbf{f}+c_n$$

for some A_n, b_n, c_n in Appendix A.1. This follows the same idea as in Matsubara et al. (2022b); Altamirano et al. (2023), and will be crucial to obtain closed form solution for both the posterior and predictive.

Robust and Conjugate Gaussian Processes Based on L_n^w, we propose the RCGP posterior

$$p^w(\mathbf{f}|\mathbf{y},\mathbf{x})\propto p(\mathbf{f})\exp\{-nL_n^w(\mathbf{f},\mathbf{y},\mathbf{x})\},$$

where we absorb β into w. Considering the same setting as for the standard GP in Section 2, the RCGP posterior and its posterior predictive have closed forms. To state them, take $\operatorname{diag}(\mathbf{v})$ as the $d\times d$ diagonal matrix D so that $D_{ij}=0$ if $i\neq j$ and $D_{ii}=v_i$.

Proposition 3.1. *Suppose $f\sim\mathcal{GP}(m,k)$ and $\varepsilon\sim\mathcal{N}(0,I_n\sigma^2)$. Then, the RCGP posterior is*

$$\begin{aligned}p^w(\mathbf{f}|\mathbf{y},\mathbf{x})&=\mathcal{N}(\mathbf{f};\mu^R,\Sigma^R),\\ \mu^R&=\mathbf{m}+K\left(K+\sigma^2 J_{\mathbf{w}}\right)^{-1}(\mathbf{y}-\mathbf{m}_{\mathbf{w}}),\\ \Sigma^R&=K\left(K+\sigma^2 J_{\mathbf{w}}\right)^{-1}\sigma^2 J_{\mathbf{w}},\end{aligned}$$

for $\mathbf{w}=(w(x_1,y_1),\dots,w(x_n,y_n))^\top$, $\mathbf{m}_{\mathbf{w}}=\mathbf{m}+\sigma^2\nabla_y\log(\mathbf{w}^2)$ and $J_{\mathbf{w}}=\operatorname{diag}(\frac{\sigma^2}{2}\mathbf{w}^{-2})$. The RCGP's posterior predictive over $f_\star=f(x_\star)$ at $x^\star\in\mathcal{X}$ is

$$\begin{aligned}p^w(f_\star|x_\star,\mathbf{x},\mathbf{y})&=\int_{\mathbb{R}^n}p(f_\star|x_\star,\mathbf{f},\mathbf{x},\mathbf{y})p^w(\mathbf{f}|\mathbf{y},\mathbf{x})d\mathbf{f}\\ &=\mathcal{N}(f_\star;\mu_\star^R,\Sigma_\star^R),\\ \mu_\star^R&=m_\star+\mathbf{k}_\star^\top\left(K+\sigma^2 J_{\mathbf{w}}\right)^{-1}(\mathbf{y}-\mathbf{m}_{\mathbf{w}}),\\ \Sigma_\star^R&=k_{\star\star}-\mathbf{k}_\star^\top(K+\sigma^2 J_{\mathbf{w}})^{-1}\mathbf{k}_\star.\end{aligned}$$

Table 1. Existing methods as special cases of RCGPs

Method	$w(x,y)$
Standard GP	$\frac{\sigma}{\sqrt{2}}$
Heteroskedastic GP	$\frac{\sigma^2}{\sqrt{2}}\cdot r(x)^{-1}$
Robust GP	$\beta\cdot\left(1+\frac{(y-m(x))^2}{c^2}\right)^{-1/2}$

Throughout, exponents are applied entry-wise, and the proofs can be found in Appendix A.1. The distributions derived in the result have the same structure as their standard GP counterparts, but replace $\sigma^2 I_n$ with the "noise term" $\sigma^2 J_{\mathbf{w}}=\sigma^2\operatorname{diag}(\frac{\sigma^2}{2}\mathbf{w}^{-2})$, and $\mathbf{m}$ with the "shrinkage term" $\mathbf{m}_{\mathbf{w}}=\mathbf{m}+\sigma^2\nabla_y\log(\mathbf{w}^2)$. We interpret both terms after discussing how w should be chosen.

Similar to standard GP regression, RCGP regression is conjugate, has a computational cost of $O(n^3)$, and storage cost of $O(n^2)$. In addition, a variety of GP schemes fall into the proposed framework through a specific choice of w. For example, $w(x,y)=\frac{\sigma}{\sqrt{2}}$ recovers the standard GP, and $w(x,y)=\frac{\sigma^2}{\sqrt{2}}\cdot r(x)^{-1}$ a heteroskedastic GP with noise rate $r(x)$; see Table 1. Importantly, it would be flawed to simply interpret RCGPs as GPs with a different noise model: $\mathbf{w}$ depends directly on $\mathbf{y}$.

Finally, we note that RCGPs are in principle not suited to interpolation: the score s_{model} and thus the loss L_n^w are not defined if $\sigma^2=0$. However, it is common to add a "nugget" to the kernel for regularisation (Andrianakis & Challenor, 2012), which makes the problem equivalent to regression with very small $\sigma^2>0$. Consequently, RCGP regression is still applicable for interpolation problems whenever this regularisation is used—and can in fact strongly improve robustness in this setting.

Hyperparameter Selection To make RCGPs practically viable, we need to estimate θ and σ^2. A first idea would be to maximise the pseudo marginal likelihood $p^w(\mathbf{y}|\theta,\sigma^2)$—the RCGP's equivalent to the marginal likelihood, whose closed form is given in Appendix A.2. Unfortunately, this would be ill-posed: neither $\exp\{-nL^w(f,\mathbf{y},\mathbf{x})\}$ nor $p^w(\mathbf{y}|\theta,\sigma^2)$ are probability densities over $\mathbf{y}$. Hence, maximising $p^w(\mathbf{y}|\theta,\sigma^2)$ is like maximum likelihood estimation for an un-normalised density whose normaliser depends on the estimated parameter, leading to implausible hyperparameters and numerical issues—a well-known issue for generalised Bayesian methods (Jewson & Rossell, 2022). Instead of $p^w(\mathbf{y}|\theta,\sigma^2)$, we thus maximise the leave-one-out cross validation (LOO-CV) predictive posteriors via

$$\hat{\sigma}^2,\hat{\theta}=\arg\max_{\sigma^2,\theta}\left\{\sum_{i=1}^{n}\log p^w(y_i|\mathbf{x},\mathbf{y}_{-i},\theta,\sigma^2)\right\},$$

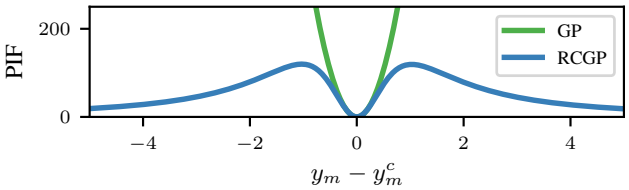

Figure 2. PIF_{GP} (**green**) and PIF_{RCGP} (**blue**) for the dataset in Figure 1. $\text{PIF}_{\text{GP}} \to \infty$ as $|y_m - y_m^c| \to \infty$, so that standard GPs are not robust. In contrast, PIF_{RCGP} is bounded, showing robustness of the RCGP.

where $\mathbf{y}_{-i} = (y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_n)$ (for details, see Section 5.2 Rasmussen & Williams, 2006). LOO-CV has been one of the primary methods for setting hyperparameters in Gaussian Processes (see e.g. Sundararajan & Keerthi, 1999; Bachoc, 2013b; Vehtari et al., 2016; Petit et al., 2020). A naive implementation of this objective function would require fitting the model n times, leading to a computational cost of $\mathcal{O}(n^4)$. Hence, we follow Sundararajan & Keerthi (1999) to obtain an analytical formulation that allows us to compute the LOO objective in $\mathcal{O}(n^3)$. By Proposition 3.1, $p^w(y_i|\mathbf{x}, \mathbf{y}_{-i}, \theta, \sigma^2) = \mathcal{N}(\mu_i^R, \sigma_i^R + \sigma^2)$ with

$$\mu_i^R = z_i + \mathbf{m}_i - [\left(K + \sigma^2 J_{\mathbf{w}}\right)^{-1} \mathbf{z}]_i [(K + \sigma^2 J_{\mathbf{w}})^{-1}]_{ii}^{-1},$$
$$\sigma_i^R = [(K + \sigma^2 J_{\mathbf{w}})^{-1}]_{ii}^{-1} - \frac{\sigma^4}{2} w(x_i, y_i)^{-2},$$

for $\mathbf{z} = \mathbf{y} - \mathbf{m}_{\mathbf{w}}$ and $\mathbf{z} = (z_1, \ldots z_n)$. The full derivation can be found in Appendix A.3. Crucially, we only need to compute $(K + \sigma^2 J_{\mathbf{w}})^{-1}$ once which leads to a computational cost of $\mathcal{O}(n^3)$, same as the standard marginal likelihood method.

Robustness RCGPs are not only computationally attractive, but also robust to outliers and non-Gaussian errors. While robustness for Bayesian methods can refer to a number of other aspects, including calibration (Grünwald, 2012; Huggins & Miller, 2023; Lyddon et al., 2019), adversarial robustness (Bogunovic et al., 2020; Kirschner & Krause, 2021), and robustness to misspecified priors (van Der Vaart & van Zanten, 2011; Bachoc, 2013a; Teckentrup, 2020; Wang et al., 2020; Karvonen, 2021; Bogunovic & Krause, 2021; Stephenson et al., 2022; Naslidnyk et al., 2023), we prove RCGP's robustness to misspecification in the error model. We do so using the classical framework of Huber (1981). To this end, we first define the contamination of a dataset $D = \{(x_i, y_i)\}_{i=1}^n$ by the datum y_m^c as $D_m^c = (D \setminus \{(x_m, y_m)\}) \cup \{(x_m, y_m^c)\}$ for some $m \in \{1, \ldots, n\}$. We quantify the impact of y_m^c on inference through the divergence between the contaminated and uncontaminated posteriors. As a function of $|y_m^c - y_m|$, this divergence is also sometimes called the *posterior influence function* (PIF) and was studied for parametric models in Ghosh & Basu (2016) and Matsubara et al. (2022b). To operationalise this, we consider the Kullback-Leibler (KL) divergence:

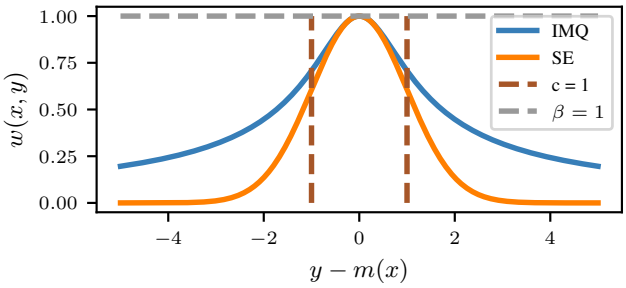

Figure 3. Comparing kernel-based w with the same hyperparameters: IMQ (**blue**) and Squared Exponential (SE) (**orange**). The dashed vertical lines indicate the soft threshold c past which a point is increasingly treated as an outlier. The SE down-weights observations more rapidly as they exceed c than the IMQ. The maximum possible weight for any observation is $\beta = 1$.

$$\text{PIF}(y_m^c, D) = \text{KL}\left(p_\beta^L(\mathbf{f}|D) \| p_\beta^L(\mathbf{f}|D_m^c)\right).$$

We call a posterior robust if $\sup_{y_m^c \in \mathcal{Y}} |\text{PIF}(y_m^c, D)| < \infty$, as this implies that even as $|y_m^c - y_m| \to \infty$, the contamination's effect on the posterior (as measured by the KL) is bounded. Choosing the KL is convenient as it allows closed form expressions for Gaussians, but we could in principle pick any other divergence with closed form expressions that is not uniformly bounded.

Proposition 3.2. *Suppose* $\mathbf{f} \sim \mathcal{GP}(m, k)$, $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, I_n \sigma^2)$, *and let* $C_k \in \mathbb{R}; k = 1, 2, 3$ *be constants independent of* y_m^c. *Then, GP regression has the PIF*

$$\text{PIF}_{\text{GP}}(y_m^c, D) = C_1 (y_m - y_m^c)^2,$$

and is not robust: $\text{PIF}_{GP}(y_m^c, f, D) \to \infty$ *as* $|y_m^c| \to \infty$. *In contrast, for RCGPs with* $\sup_{x,y} w(x, y) < \infty$,

$$\text{PIF}_{\text{RCGP}}(y_m^c, D) \leq C_2 (w(x_m, y_m^c)^2 y_m^c)^2 + C_3.$$

Thus, if $\sup_{x,y} \left\{y \cdot w(x, y)^2\right\} < \infty$, *RCGP is robust since* $\sup_{y_m^c} |\text{PIF}_{\text{RCGP}}(y_m^c, D)| < \infty$.

The proof is in Appendix A.4. The two conditions on w in Proposition 3.2 have clear interpretations: $\sup_{x,y} w(x, y) < \infty$ ensures that no observation has infinite weight—which would be antithetical to robustness. Further, $\sup_{x,y} \left\{y \cdot w(x, y)^2\right\} < \infty$ demonstrates that for robustness, w must down-weight observations at least at rate $1/y$ for $|y|$ large enough. Figure 2 shows the PIF for an RCGP with a weighting function that satisfies these conditions. Notably, the conditions on w in Proposition 3.2 can only hold if w depends on y. A consequence is that the w associated with heteroskedastic GPs in Table 1 does not lead to a bounded PIF—and is *not* robust in the sense defined above.

Choice of Weighting Function Many choices of w satisfy the conditions of Proposition 3.2. An ideal choice of w

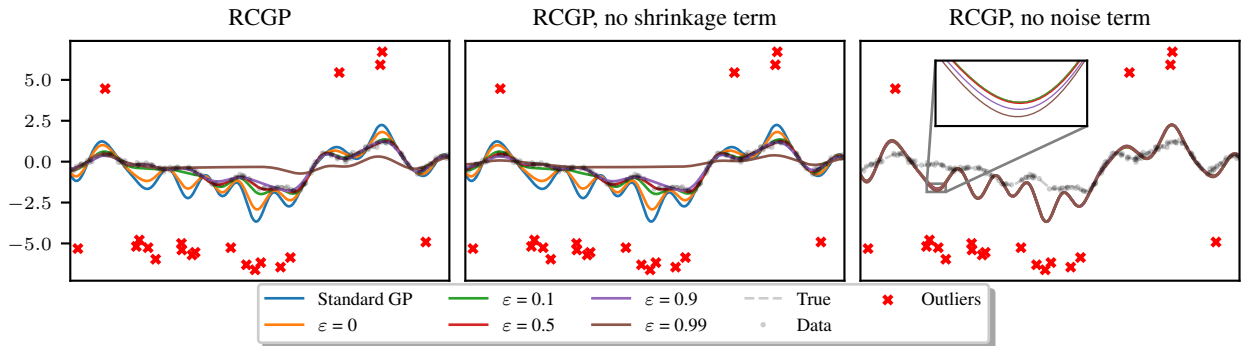

Figure 4. Posterior predictive mean for varying values of c obtained by adjusting ε using the quantile absolute deviation method proposed in Section 3, applied to a synthetic dataset with 10% uniformly generated outliers. **Left:** Full RCGP. **Center:** RCGP with no shrinkage term. **Right:** RCGP with no noise term.

attains its maximum close to reasonable data points, and decreases as outliers become more extreme. In the remainder, our measure of how outlying an observation y_i is will be $|y_i - m(x_i)|$. Additionally, w should be smooth to ensure its derivatives exist. Conveniently, this makes weighting functions constructed via infinitely differentiable radial kernels k as $w(x_i, y_i) = k(y_i, m(x_i))$ well-suited. We advocate for the Inverse Multi-Quadratic (IMQ) kernel: it has heavy tails, so that extreme observations are not weighted down too much; see Figure 3. It is given by

$$w_{\text{IMQ}}(x, y) = \beta\Big(1 + \frac{(y - m(x))^2}{c^2}\Big)^{-\frac{1}{2}},$$

with $\beta, c > 0$. Weighting functions chosen this way have two hyperparameters: the soft threshold c, and the learning rate β from (1) that we pulled into w. While it is in principle possible to pick both β and c jointly, we fix $\beta = \frac{\sigma}{\sqrt{2}}$ and choose only c. We do so since joint selection of β and c is numerically unstable due to near non-identifiability. While we could fix c and select β, this is difficult since learning rate selection is largely unresolved (see e.g. Lyddon et al., 2019; Syring & Martin, 2019; Bochkina, 2023; Wu & Martin, 2023; Frazier et al., 2023). Fixing $\beta = \frac{\sigma}{\sqrt{2}}$ and choosing c is much easier as c is interpretable. In particular, as $c \to \infty$, $w \to \frac{\sigma}{\sqrt{2}}$ so that the limiting setting recovers the standard GP (cf. Table 1). On the other hand, finite values $c < \infty$ can be interpreted as a soft outlier threshold: Points y_i for which $|y_i - f(x_i)| = c + \xi$ are increasingly treated as outliers the larger ξ becomes. This is illustrated in Figure 3, which depicts two kernels with $c = 1 = \beta$, and shows that the weights decrease rapidly to 0 once the threshold is exceeded.

While, in principle, it is possible to choose c by maximising the leave-one-out cross-validation predictive posterior, the performance is not optimal in practice. This is likely because maximising the predictive posterior for extreme observations tends to match/fit these outliers by *increasing* c, leading to a less robust method. See Appendix B.3 for further discussion. Therefore, we propose choosing c via the quantile absolute deviation around the prior mean as $c = Q_n(1 - \varepsilon)$, where $Q_n(1 - \varepsilon)$ is the $(1 - \varepsilon)$-th quantile of $\{|y_i - m(x_i)|\}_{i=1}^n$ for $\varepsilon \in [0, 1]$. As a default setting, we suggest $\varepsilon = 0.05$, which implies that we expect at most 5% of the data to be outliers.

Figure 4 shows an example of the impact of the choice of ε in the posterior where we observe that choosing $\varepsilon = 0.99$ will lead inferences that are *too* robust and treat nearly all points as outliers. Conversely, setting $\varepsilon = 0$ nearly reproduces the non-robust standard GP. Intermediate values exhibit a well-balanced and effective performance.

Crucially, the proposed weighting function depends on the prior mean $m(x_i)$. Hence, if the prior mean is not well specified, the method may discard observations that are not outliers or vice versa. (see Figure 7). Therefore, we emphasise the importance of carefully selecting the prior mean, as this choice could impact the performance of our method. Note that this consideration is crucial not only for our approach, but also for the standard GP (see e.g. De Ath et al., 2020; Hwang et al., 2023)

Interpretation of RCGP Terms Having discussed our choice of w, we are now ready to interpret the new terms arising in Proposition 3.1. For the RCGP, the noise term $\sigma^2 J_{\mathbf{w}} = \sigma^2 I_n(\mathbf{1} + (\mathbf{y} - \mathbf{m})^2/c^2)$ replaces the standard GP term $\sigma^2 I_n$. Here, the exponents have been applied element-wise and $\mathbf{1} = (1, \ldots 1)^\top \in \mathbb{R}^n$. Functionally, $\sigma^2 J_{\mathbf{w}}$ treats outliers as though they were noisier than other observations. Note however that the weight for y_i depends on y_i itself—and thus *cannot* be interpreted as coming from a different or heteroskedastic noise model on $\boldsymbol{\varepsilon}$. Conceptually, the term treats information as unreliable whenever it comes from observations y_i that deviate very extremely from the prior mean $m(x_i)$. This not only ensures robustness, but also has computational benefits: in particular, the term tends to improve conditioning, making it more numerically stable to

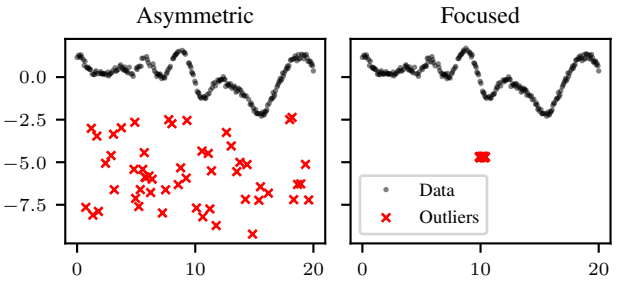

Figure 5. Considered contamination regimes are asymmetric (left) and focused (right).

invert $K+\sigma^2 J_{\mathbf{w}}$ than $K+\sigma^2 I_n$. This is especially relevant when σ is small (e.g. for interpolation) and n large, in which case numerical conditioning is typically a significant issue for GPs (Andrianakis & Challenor, 2012).

Next, $\mathbf{y}-\mathbf{m_w} = \mathbf{y}-\mathbf{m}-\sigma^2\left[\frac{2(\mathbf{y}-\mathbf{m})}{c^2\mathbf{1}+(\mathbf{y}-\mathbf{m})^2}\right]$ replaces the standard GP's $\mathbf{y}-\mathbf{m}$ for RCGPs. Unlike the noise term, the i-th shrinkage term $[\mathbf{m_w}]_i$ is *not* monotonically increasing in $|y_i - m(x_i)|$. Rather, it is increasing as $|y_i - m(x_i)| \uparrow c$, peaks at σ^2/c for $|y_i - m(x_i)| = c$, and then decreases monotonically as $|y_i - m(x_i)|$ increases further. The effect is most obvious for $\mathbf{m}=0$, for which $\mathbf{y}-\mathbf{m_w} = \mathbf{y}(\mathbf{1} - 2\sigma^2[c^2\mathbf{1}+\mathbf{y}^2]^{-1})$. Generally, shrinkage is associated with trading off a slight bias for a reduction in variance (e.g. Von Luxburg & Schölkopf, 2011). This also applies here: relative to the GP, the RCGP's posterior mean will exhibit smaller variance at the expense of a slight bias towards m.

Figure 4 shows the impact of each term in the predictive posterior. It is clear that the main term providing robustness to our method is the noise term.

Table 2. Average test set mean absolute error and standard deviation (in brackets) for 50 train–test splits.

	GP	RCGP	t-GP	m-GP
		No Outliers		
Synthetic	**0.09 (0.00)**	**0.09 (0.00)**	**0.09 (0.00)**	0.33 (0.00)
Boston	**0.19 (0.01)**	**0.19 (0.01)**	**0.19 (0.01)**	0.28 (0.00)
Energy	0.03 (0.00)	**0.02 (0.00)**	0.03 (0.00)	0.61 (0.00)
Yacht	0.02 (0.01)	0.02 (0.01)	**0.01 (0.00)**	0.33 (0.00)
		Focused Outliers		
Synthetic	0.19 (0.00)	**0.15 (0.00)**	0.18 (0.00)	0.23 (0.00)
Boston	0.23 (0.06)	**0.22 (0.01)**	0.27 (0.00)	0.27 (0.00)
Energy	0.03 (0.04)	**0.02 (0.00)**	0.03 (0.05)	0.24 (0.00)
Yacht	0.26 (0.15)	**0.10 (0.14)**	0.20 (0.04)	0.24 (0.00)
		Asymmetric Outliers		
Synthetic	1.14 (0.00)	0.63 (0.00)	1.06 (0.00)	**0.61 (0.00)**
Boston	0.63 (0.02)	**0.49 (0.00)**	0.52 (0.00)	0.52 (0.00)
Energy	0.54 (0.02)	0.44 (0.04)	0.42 (0.02)	**0.41 (0.00)**
Yacht	0.54 (0.06)	**0.35 (0.02)**	0.41 (0.00)	0.40 (0.00)
		Uniform		
Synthetic	0.34 (0.00)	**0.21 (0.00)**	0.30 (0.00)	0.27 (0.00)
Boston	0.53 (0.13)	0.51 (0.12)	0.30 (0.01)	**0.24 (0.00)**
Energy	0.25 (0.03)	0.24 (0.03)	**0.23 (0.05)**	0.23 (0.00)
Yacht	0.36 (0.16)	0.29 (0.07)	**0.16 (0.00)**	0.34 (0.00)

Table 3. Average clock time in seconds and its standard deviation (in brackets) across 50 repetitions

	GP	RCGP	t-GP	m-GP
Synthetic	1.5 (0.1)	1.2 (0.0)	2.2 (0.0)	3.0 (0.0)
Boston	1.9 (0.5)	5.1 (0.9)	30.7 (6.1)	16.7 (1.7)
Energy	3.8 (0.9)	4.6 (2.0)	34.0 (11)	33.8 (0.3)
Yacht	1.6 (0.3)	2.1 (0.2)	5.6 (0.7)	4.5 (0.4)

4. Experiments

The code is available at `https://github.com/maltamiranomontero/RCGP`. The extensive literature on robust GPs makes it impossible to compare RCGPs to all competitors. Thus, we choose two representative and popular approaches: a GP with heavy tailed Student's t errors (Jylänki et al., 2011, "t-GP"), and a GP directly modelling outliers via mixture distributions (Lu et al., 2023, "m-GP"). We do not compare against outlier-removal methods: they complement—rather than compete with—RCGPs. Throughout, we picked w, c and β as proposed in Section 3. All hyperparameters are selected via L-BFGS.

Benchmarking We assess our method on four benchmark datasets, including the synthetic problem in Figure 1 ($d=1, n=300$), the `Boston` dataset ($d=13, n=506$), and two datasets from the UCI repository: `Energy` ($d=8, n=768$) and `Yacht` ($d=6, n=308$). These allow us to compare the performances of robust GP methods in a range of dimensions and number of data points. For each dataset, we consider four settings: the original dataset (i.e. no outliers), focused outliers (i.e. outliers clustered in x- and y- space), asymmetric outliers (i.e. observations corrupted by strictly negative shifts) and uniform outliers (both positive and negative shifts at random); see Figures 1 and 5 for illustrations. In each case, 10% of observations are perturbed to become outliers. Results are provided in Tables 2 and 3, and a full description of the datasets and outlier generation process is provided in Appendix B.3.

As expected, standard GPs outperform robust methods in absolute mean error when there are no outliers. That being said, RCGPs can easily compete in this setting—indicating that there are no clear drawbacks to using RCGPs as a default. For example, on both the `Energy` and `Boston` datasets, the performance of RCGPs in the setting with no outliers is comparable to that of GPs. Further, RCGPs tend to outperform their competitors in the presence of focused and asymmetric outliers. This happens as the competitors'

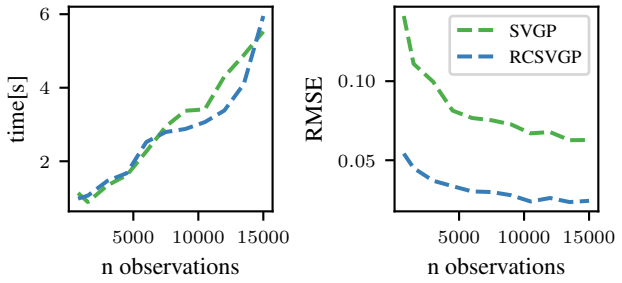

Figure 6. Average clock time (left) and root mean square error (right) for SVGP (green) and RCSVGP (blue) on the synthetic dataset with 10% uniform outliers and $m = \sqrt{n}$ inducing points across 10 repetitions.

posited noise models are symmetric, but the outlier generation is not. This phenomenon demonstrates the advantage of using a generalised Bayesian approach as opposed to a more refined noised model. In the uniform outlier setting , t-GPs outperform all alternatives. This is to be expected: Student's t errors are a reasonable approximation to that error generating process. Though t-GPs perform even better, RCGPs still significantly outperform GPs. Further, while RCGPs and GPs have comparable computational cost, t-GPs and m-GPs are significantly more computationally expensive. To demonstrate this, Table 3 compares the cost of training (including hyperparameter selection) of each method. The slowdown of t-GPs and m-GPs is due to needing variational approximations, and especially noticeable for datasets with larger n. Both GPs and RCGPs have runtimes of $O(n^3)$, though there are minor numerical differences due to the adaptive stopping rule of L-BFGS for hyperparameter optimisation.

Sparse Variational Gaussian Processes (SVGPs) The $O(n^3)$ cost of GPs is prohibitive for large n. A popular remedy are SVGPs (Titsias, 2009), which reduce the cost to $O(nm^2)$ with $m \ll n$ inducing points $\mathbf{u} = (u_1, \ldots, u_m)^\top$. RCGPs are amenable to this type of inference. The resulting robust conjugate SVGP (RCSVGP) is derived below.

Proposition 4.1. *For $f \sim \mathcal{GP}(m, k)$, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$, the RCSVGP posterior is $f \sim \mathcal{GP}(\widetilde{\mu}, \widetilde{\Sigma})$, where*

$$\begin{aligned}
\widetilde{\mu}(x) &= \phi_{\mathbf{u}}(x)^\top \mu_{\mathbf{u}}, \\
\widetilde{\Sigma}(x, x') &= k(x, x') - \phi_{\mathbf{u}}(x)^\top \left(K_{\mathbf{uu}} - \Sigma_{\mathbf{u}}\right) \phi_u(x'), \\
\mu_{\mathbf{u}} &= \mathbf{m} + K_{\mathbf{uu}} P_{\mathbf{u}}^{-1} K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} (\mathbf{y} - m_{\mathbf{w}}), \\
\Sigma_{\mathbf{u}} &= K_{\mathbf{uu}} P_{\mathbf{u}}^{-1} K_{\mathbf{uu}},
\end{aligned}$$

for $P_{\mathbf{u}} = \left(K_{\mathbf{uu}} + K_{\mathbf{u}}^\top \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}\right)$, $[K_{\mathbf{uu}}]_{ij} = k(u_i, u_j)$, $[K_{\mathbf{u}}]_{ij} = k(u_i, x_j)$, $[k_{\mathbf{u}}(x)]_i = k(u_i, x)$, and $\phi_{\mathbf{u}}(x) = K_{\mathbf{uu}}^{-1} k_{\mathbf{u}}(x)$.

$$\begin{aligned}
J(\mathbf{u}, \theta, \sigma^2) = \frac{1}{2} \nu^\top K_{\mathbf{u}}^\top Q_{\mathbf{u}}^{-1} K_{\mathbf{u}} \nu + \frac{1}{2} \log \left(\frac{\det \left(K_{\mathbf{uu}}\right)^2}{\det \left(Q_{\mathbf{u}}\right)} \right) \\
+ C(\sigma^2) - \operatorname{Tr} \left(\sigma^{-2} J_{\mathbf{w}}^{-\frac{1}{2}} (K - K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1} K_{\mathbf{u}}) J_{\mathbf{w}}^{-\frac{1}{2}} \right),
\end{aligned}$$

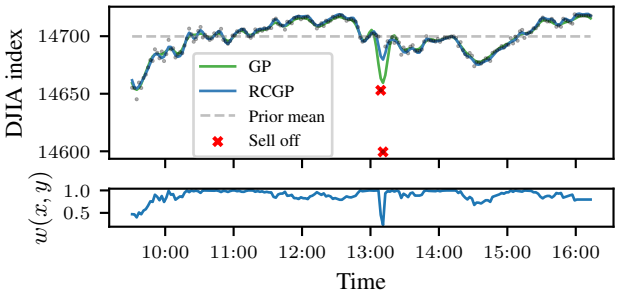

Figure 7. Top: GP (green) and RCGP (blue) regression on the DJIA with $m(x) = \frac{1}{n} \sum_{i=1}^n y_i$. While the GP overfits to outliers around 1pm, RCGP is robust. *Bottom:* w_{IMQ} with β, c chosen as proposed in Section 3.

where $Q_{\mathbf{u}} = K_{\mathbf{uu}} + K_{\mathbf{u}}^\top \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}$, $\nu = \sigma^{-2} J_{\mathbf{w}}^{-1} (\mathbf{y} - \mathbf{m}_{\mathbf{w}})$, and $C(\sigma^2)$ is a function that only depends on σ^2.

See Appendix A.5 for a proof and details. A numerical comparison between SVGPs and RCSVGPs for the synthetic example with 10% uniform outliers is presented in Figure 6. While RCSVGPs and SVGPs have similar runtimes, RCSVGPs make far more accurate predictions. In Appendix B.4, we show that this generalises to the other datasets and outlier types.

Twitter Flash Crash To illustrate the practical utility of RCGPs, we next analyse the Dow Jones Industrial Average (DJIA) index on the 17/04/2013. On this day, the Associated Press' Twitter account was hacked and falsely tweeted that explosions at the White House had injured the president. This lead to a rapid sell-off in American stock markets within seconds, quickly followed by an equally fast bounce-back. This resulted in a few moments after 1pm that day where the DJIA did not accurately reflect the USA's economic realities. In this sense, it is reasonable to regard the resulting group of extreme observations as outliers. Figure 7 depicts the raw data as well as a GP and RCGP fit. The figure not only illustrates the robustness of RCGPs, but also sheds further light on relevant trade-offs when choosing w. As the bottom panel shows, most points have weights ≈ 1. There are two exceptions to this: one of these is the group of points past 1pm where the crash occurred. Here, the observations are down-weighted almost all the way to zero. This is desirable, and exactly what we would expect RCGP to do. Perhaps more surprisingly, weights are also significantly smaller than 1 before 10am. This is not obviously desirable, and due to the choice of prior mean. In particular, recall that $w(x_i, y_i)$ is small if $|y_i - m(x_i)|$ is large. For Figure 7, the constant function $m(x) = \frac{1}{n} \sum_{i=1}^n y_i$ is chosen as the prior mean. Relative to this prior mean, the first observations are outliers. While this is not ideal, it is worth noting that this behaviour can be easily remedied by choosing m more carefully. For example, one can fit a simple parametric model for m, as is extremely common in the literature. See

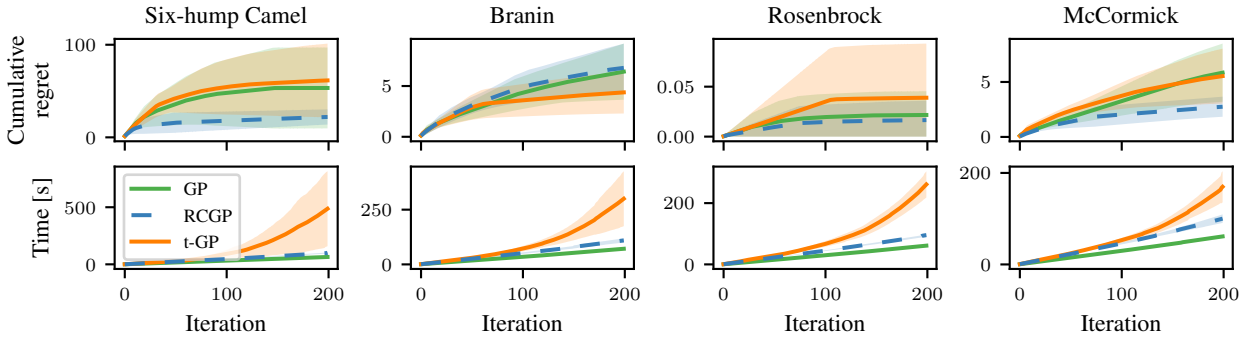

Figure 8. Mean cumulative regret (top) and clock time (bottom) for BO with GP (green), RCGP (blue) and t-GP (orange) with 20% asymmetric outliers and UCB acquisition function over 10 realisations.

Appendix B.1 for a further discussion.

Bayesian Optimisation One of the most successful applications of GPs has been Bayesian optimisation (BO) (Garnett, 2021). In BO, a data-dependent acquisition function $a : \mathcal{X} \to \mathbb{R}$ based on the posterior GP characterises desirable regions of $\mathcal{X}$ to choose the next—often noisy—function evaluations of f from. Here, misspecification of the noise model is a significant concern: it leads to poor acquisition functions; see e.g. Martinez-Cantin et al. (2018); Bogunovic et al. (2020); Kirschner & Krause (2021). Using RCGPs instead of GPs leads to a BO algorithm that is naturally robust to such misspecification. We illustrate this using the upper-confidence bound (UCB) and probability of improvement (PI) acquisition functions.

Proposition 4.2. *Let $f \sim \mathcal{GP}(m, k)$, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. The UCB and PI acquisition functions for RCGPs are*

$$a_{\text{UCB-RCGP}}(x_\star) = \mu_\star^R + \lambda(\Sigma_\star^R)^{1/2}$$
$$a_{\text{PI-RCGP}}(x_\star) = \Phi\Big(\big(\mu_\star^R - f(x_{max})\big)(\Sigma_\star^R)^{-1/2}\Big)$$

where Φ is the cdf of a standard normal, and x_{max} is the best solution we have so far.

Here, we have re-used the notations of Proposition 3.1, and full derivations are available in Appendix A.6. The only paper explicitly studying BO in the presence of outliers is Martinez-Cantin et al. (2018), which uses a t-GP. We thus compare RCGPs to GPs and t-GPs on the classical six-hump camel, Branin, McCormick and Rosenbrock functions (see Appendix B.5). Figure 8 shows the results if each new function evaluation has a 20% chance of being contaminated by an asymmetric outlier generated as in Figure 5, and the BO uses UCB as the acquisition function. In terms of cumulative regret, RCGPs outperform GPs. While t-GPs can match this, they take orders of magnitude longer to run. The results remain the same if the PI acquisition function is used (See Appendix B.5). It is notable that even without outliers, RCGPs match or outperform GPs (see Appendix B.5). This is likely because the GP priors on f are misspecified, and robustness to misspecified error models may help in this setting.

5. Conclusion

A major issue with existing GP regression techniques is that they are *either* robust *or* conjugate—never both. RCGPs solve this issue through generalised Bayesian inference, and demonstrate that robustness does not require prohibitive computational cost. Intriguingly, our experiments also indicate that there is no clear disadvantage from using RCGPs in the absence of misspecification—raising the possibility that RCGPs may become a preferred default choice over GPs in the future.

Limitations and future work One limitation of our work is its dependency on a well-specified prior mean through the weighting function, as illustrated in Figure 7. Despite this, the weighting function offers flexibility for practitioners to select one that best suits their specific problem. For example, a practitioner concerned only with outliers in one direction could choose a weighting function that penalises deviations only in that direction. Additionally, we provide guidelines on selecting a weighting function to ensure that RCGP remains provably outlier robust (see Proposition 3.2).

Another limitation of our method is the inability to obtain a posterior over the kernel hyperparameters due to the use of generalised Bayes. However, it is important to emphasise that the primary focus of our method is on speed and robustness rather than implementing a fully Bayesian procedure. Full Bayesian approaches that account for kernel hyperparameter uncertainty with standard GPs are significantly slower, so they are not our primary comparison. Instead, we focus on conjugacy, which is generally not achievable with a prior on hyperparameters.

Lastly, the flexibility and conjugacy of RCGPs means that there is significant scope to extend their use to settings beyond those presented here—including multi-output GPs (Bonilla et al., 2007; Altamirano & Tobar, 2022), linear-time GPs (Hartikainen & Sarkka, 2010), GPs with derivative (Morris et al., 1993; Wu et al., 2017) or integral data (Yousefi et al., 2019; Tanaka et al., 2019), deep GPs (Damianou & Lawrence, 2013), transformed GPs (Maroñas et al., 2021), and probabilistic numerics (Hennig et al., 2022).

Acknowledgements

FXB and JK were supported by the EPSRC grant [EP/Y011805/1].

Impact statement

This paper presents work whose goal is to advance the field of statistical Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Algikar, P. and Mili, L. Robust Gaussian process regression with Huber likelihood. *arXiv:2301.07858*, 2023.

Altamirano, M. and Tobar, F. Nonstationary multi-output gaussian processes via harmonizable spectral mixtures. In *International Conference on Artificial Intelligence and Statistics*, pp. 3204–3218. PMLR, 2022.

Altamirano, M., Briol, F.-X., and Knoblauch, J. Robust and scalable Bayesian online changepoint detection. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 642–663. PMLR, 2023.

Andrade, D. and Takeda, A. Robust Gaussian process regression with the trimmed marginal likelihood. In *Uncertainty in Artificial Intelligence*, pp. 67–76, 2023.

Andrianakis, I. and Challenor, P. The effect of the nugget on Gaussian process emulators of computer models. *Computational Statistics and Data Analysis*, 56(12):4215–4228, 2012.

Bachoc, F. Cross validation and maximum likelihood estimations of hyper-parameters of Gaussian processes with model misspecification. *Computational Statistics and Data Analysis*, 66:55–69, 2013a.

Bachoc, F. Cross validation and maximum likelihood estimations of hyper-parameters of gaussian processes with model misspecification. *Computational Statistics & Data Analysis*, 66:55–69, 2013b.

Balandat, M., Karrer, B., Jiang, D. R., Daulton, S., Letham, B., Wilson, A. G., and Bakshy, E. BoTorch: A framework for efficient Monte-Carlo Bayesian optimization. In *Advances in Neural Information Processing Systems 33*, 2020. URL http://arxiv.org/abs/1910.06403.

Barp, A., Briol, F.-X., Duncan, A., Girolami, M., and Mackey, L. Minimum Stein discrepancy estimators. In *Advances in Neural Information Processing Systems*, pp. 12964–12976, 2019.

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

Bochkina, N. Bernstein-von Mises theorem and misspecified models: a review. In *Foundations of Modern Statistics*, pp. 355–380, 2023.

Bogunovic, I. and Krause, A. Misspecified Gaussian process bandit optimization. *Advances in Neural Information Processing Systems*, 34:3004–3015, 2021.

Bogunovic, I., Krause, A., and Scarlett, J. Corruption-tolerant Gaussian process bandit optimization. In *International Conference on Artificial Intelligence and Statistics*, pp. 1071–1081, 2020.

Bonilla, E. V., Chai, K., and Williams, C. Multi-task Gaussian process prediction. *Advances in Neural Information Processing Systems*, 20, 2007.

Boustati, A., Akyildiz, O. D., Damoulas, T., and Johansen, A. Generalised Bayesian filtering via sequential Monte Carlo. In *Advances in Neural Information Processing Systems*, pp. 418–429, 2020.

Chérief-Abdellatif, B.-E. and Alquier, P. MMD-Bayes: Robust Bayesian estimation via maximum mean discrepancy. In *Symposium on Advances in Approximate Bayesian Inference*, 2020.

Daemi, A., Alipouri, Y., and Huang, B. Identification of robust Gaussian process regression with noisy input using EM algorithm. *Chemometrics and Intelligent Laboratory Systems*, 191:1–11, 2019.

Damianou, A. and Lawrence, N. D. Deep Gaussian processes. In *Artificial intelligence and statistics*, pp. 207–215. PMLR, 2013.

De Ath, G., Fieldsend, J. E., and Everson, R. M. What do you mean? the role of the mean function in bayesian optimisation. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, pp. 1623–1631, 2020.

Dellaporta, C., Knoblauch, J., Damoulas, T., and Briol, F.-X. Robust Bayesian inference for simulator-based models via the MMD posterior bootstrap. In *International Conference on Artificial Intelligence and Statistics*, pp. 943–970, 2022.

Duran-Martin, G., Altamirano, M., Shestopaloff, A. Y., Knoblauch, J., Jones, M., Briol, F.-X., and Murphy, K. Outlier-robust Kalman filtering through generalised Bayes. *arXiv:2403.05646*, 2024.

Dutordoir, V., Wilk, M., Artemev, A., and Hensman, J. Bayesian image classification with deep convolutional Gaussian processes. In *International Conference on Artificial Intelligence and Statistics*, pp. 1529–1539. PMLR, 2020.

Fong, E., Holmes, C., and Walker, S. G. Martingale posterior distributions. *arXiv:2103.15671*, 2021.

Frazier, D. T., Kohn, R., Drovandi, C., and Gunawan, D. Reliable Bayesian inference in misspecified models. *arXiv:2302.06031*, 2023.

Frazier, D. T., Knoblauch, J., and Drovandi, C. The impact of loss estimation on gibbs measures. *arXiv preprint arXiv:2404.15649*, 2024.

Futami, F., Sato, I., and Sugiyama, M. Variational inference based on robust divergences. In *International Conference on Artificial Intelligence and Statistics*, pp. 813–822, 2018.

Gardner, J. R., Pleiss, G., Bindel, D., Weinberger, K. Q., and Wilson, A. GPyTorch: Blackbox matrix-matrix Gaussian Pprocess inference with GPU acceleration. In *Advances in Neural Information Processing Systems*, pp. 7587–7597, 2018.

Garnett, R. *Bayesian Optimization*. Cambridge University Press, 2021.

Ghosh, A. and Basu, A. Robust Bayes estimation using the density power divergence. *Annals of the Institute of Statistical Mathematics*, 68(2):413–437, 2016.

Goldberg, P., Williams, C., and Bishop, C. Regression with input-dependent noise: A Gaussian process treatment. *Advances in Neural Information Processing Systems*, 10, 1997.

Grünwald, P. The safe Bayesian. In *International Conference on Algorithmic Learning Theory*, pp. 169–183, 2012.

Hartikainen, J. and Sarkka, S. Kalman filtering and smoothing solutions to temporal Gaussian process regression models. In *2010 IEEE International Workshop on Machine Learning for Signal Processing*, pp. 379–384, 2010. ISBN 9781424478774. URL https://users.aalto.fi/$\sim$ssarkka/pub/gp-ts-kfrts.pdf.

Hennig, P., Osborne, M. A., and Kersting, H. *Probabilistic Numerics: Computation as Machine Learning*. Cambridge University Press, 2022.

Hensman, J., Fusi, N., and Lawrence, N. D. Gaussian processes for big data. *arXiv preprint arXiv:1309.6835*, 2013.

Hooker, G. and Vidyashankar, A. Bayesian model robustness via disparities. *TEST*, 23(3):556–584, 2014.

Huber, P. J. Robust statistics. *Wiley Series in Probability and Mathematical Statistics*, 1981.

Huggins, J. H. and Miller, J. W. Reproducible model selection using bagged posteriors. *Bayesian Analysis*, 18(1): 79–104, 2023.

Hwang, S.-g., L'Huillier, B., Keeley, R. E., Jee, M. J., and Shafieloo, A. How to use gp: effects of the mean function and hyperparameter selection on gaussian process regression. *Journal of Cosmology and Astroparticle Physics*, 2023(02):014, 2023.

Hyvärinen, A. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6:695–708, 2006.

Jewson, J. and Rossell, D. 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.

Jewson, J., Smith, J. Q., and Holmes, C. Principled Bayesian minimum divergence inference. *Entropy*, 20(6):442, 2018.

Jylänki, P., Vanhatalo, J., and Vehtari, A. Robust Gaussian process regression with a student-t likelihood. *Journal of Machine Learning Research*, 12(11), 2011.

Karvonen, T. Estimation of the scale parameter for a misspecified Gaussian process model. *arXiv:2110.02810*, 2021.

Kirschner, J. and Krause, A. Bias-robust Bayesian optimization via dueling bandits. In *International Conference on Machine Learning*, pp. 5595–5605, 2021.

Knoblauch, J. Robust deep Gaussian processes. *arXiv preprint arXiv:1904.02303*, 2019.

Knoblauch, J., Jewson, J. E., and Damoulas, T. Doubly robust Bayesian inference for non-stationary streaming data with β-divergences. In *Advances in Neural Information Processing Systems*, pp. 64–75, 2018.

Knoblauch, J., Jewson, J., and Damoulas, T. An optimization-centric view on Bayes' rule: Reviewing and generalizing variational inference. *Journal of Machine Learning Research*, 23(132):1–109, 2022.

Kuss, M. *Gaussian process models for robust regression, classification, and reinforcement learning*. PhD thesis, echnische Universität Darmstadt Darmstadt, Germany, 2006.

Legramanti, S., Durante, D., and Alquier, P. Concentration and robustness of discrepancy-based ABC via Rademacher complexity. *arXiv:2206.06991*, 2022.

Li, Z.-Z., Li, L., and Shao, Z. Robust Gaussian process regression based on iterative trimming. *Astronomy and Computing*, 36:100483, 2021.

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

Lu, Y., Ma, J., Fang, L., Tian, X., and Jiang, J. Robust and scalable Gaussian process regression and its applications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21950–21959, 2023.

Lyddon, S. P., Holmes, C. C., and Walker, S. G. General Bayesian updating and the loss-likelihood bootstrap. *Biometrika*, 106(2):465–478, 2019.

Lyu, S. Interpretation and generalization of score matching. In *Conference on Uncertainty in Artificial Intelligence*, pp. 359–366, 2009.

Maroñas, J., Hamelijnck, O., Knoblauch, J., and Damoulas, T. Transforming gaussian processes with normalizing flows. In *International Conference on Artificial Intelligence and Statistics*, pp. 1081–1089. PMLR, 2021.

Martinez-Cantin, R., Tee, K., and McCourt, M. Practical Bayesian optimization in the presence of outliers. *International Conference on Artificial Intelligence and Statistics*, pp. 1722–1731, 2018.

Matsubara, T., Knoblauch, J., Briol, F.-X., Oates, C., et al. Generalised Bayesian inference for discrete intractable likelihood. *arXiv:2206.08420*, 2022a.

Matsubara, T., Knoblauch, J., Briol, F.-X., and Oates, C. J. Robust generalised Bayesian inference for intractable likelihoods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 84(3):997–1022, 2022b.

Matthews, A. G. d. G., van der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrá, P., Ghahramani, Z., and Hensman, J. GPflow: A Gaussian process library using TensorFlow. *Journal of Machine Learning Research*, 18(40):1–6, apr 2017. URL http://jmlr.org/papers/v18/16-537.html.

Miller, J. W. Asymptotic normality, concentration, and coverage of generalized posteriors. *Journal of Machine Learning Research*, 22(1):7598–7650, 2021.

Miller, J. W. and Dunson, D. B. Robust bayesian inference via coarsening. *Journal of the American Statistical Association*, 2018.

Moreno-Muñoz, P., Artés, A., and Alvarez, M. Heterogeneous multi-output Gaussian process prediction. *Advances in Neural Nnformation Processing Systems*, 31, 2018.

Morris, M. D., Mitchell, T. J., and Ylvisaker, D. Bayesian design and analysis of computer experiments: Use of derivatives in surface prediction. *Technometrics*, 35(3): 243–255, 1993.

Naish-Guzman, A. and Holden, S. Robust regression with twinned Gaussian processes. *Advances in Neural Information Processing Systems*, 20, 2007.

Naslidnyk, M., Kanagawa, M., Karvonen, T., and Mahsereci, M. Comparing scale parameter estimators for Gaussian process regression Cross validation and maximum likelihood. *arXiv:2307.07466*, 2023.

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

Pacchiardi, L. and Dutta, R. Generalized Bayesian likelihood-free inference using scoring rules estimators. *arXiv:2104.03889*, 2021.

Paleyes, A., Mahsereci, M., and Lawrence, N. D. Emukit: A Python toolkit for decision making under uncertainty. *Proceedings of the Python in Science Conference*, 2023.

Park, C., Borth, D. J., Wilson, N. S., Hunter, C. N., and Friedersdorf, F. J. Robust Gaussian process regression with a bias model. *Pattern Recognition*, 124:108444, 2022.

Petit, S., Bect, J., Da Veiga, S., Feliot, P., and Vazquez, E. Towards new cross-validation-based estimators for gaussian process regression: efficient adjoint computation of gradients. *arXiv preprint arXiv:2002.11543*, 2020.

Ranjan, R., Huang, B., and Fatehi, A. Robust Gaussian process modeling using EM algorithm. *Journal of Process Control*, 42:125–136, 2016.

Rasmussen, C. E. and Williams, C. K. *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006.

Salimbeni, H., Dutordoir, V., Hensman, J., and Deisenroth, M. Deep Gaussian processes with importance-weighted variational inference. In *International Conference on Machine Learning*, pp. 5589–5598. PMLR, 2019.

Santner, T. J., Williams, B. J., and Notz, W. I. *The Design and Analysis of Computer Experiments*. Springer, 2nd edition, 2018. ISBN 9781493988457.

Schmon, S. M., Cannon, P. W., and Knoblauch, J. Generalized posteriors in approximate Bayesian computation. In *Symposium on Advances in Approximate Bayesian Inference*, 2020.

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and De Freitas, N. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015.

Stegle, O., Fallert, S. V., MacKay, D. J., and Brage, S. Gaussian process robust regression for noisy heart rate data. *IEEE Transactions on Biomedical Engineering*, 55 (9):2143–2151, 2008.

Stephenson, W. T., Ghosh, S., Nguyen, T. D., Yurochkin, M., Deshpande, S. K., and Broderick, T. Measuring the robustness of Gaussian processes to kernel choice. In *International Conference on Artificial Intelligence and Statistics, PMLR*, volume 151, pp. 3308–3331, 2022.

Sundararajan, S. and Keerthi, S. Predictive app roaches for choosing hyperparameters in gaussian processes. *Advances in neural information processing systems*, 12, 1999.

Syring, N. and Martin, R. Calibrating general posterior credible regions. *Biometrika*, 106(2):479–486, 2019.

Tanaka, Y., Tanaka, T., Iwata, T., Kurashima, T., Okawa, M., Akagi, Y., and Toda, H. Spatially aggregated Gaussian processes with multivariate areal outputs. In *Advances in Neural Information Processing Systems*, 2019.

Teckentrup, A. L. Convergence of Gaussian process regression with estimated hyper-parameters and applications in Bayesian inverse problems. *SIAM-ASA Journal on Uncertainty Quantification*, 8(4):1310–1337, 2020. ISSN 21662525. doi: 10.1137/19M1284816.

Titsias, M. Variational learning of inducing variables in sparse Gaussian processes. In *Artificial intelligence and statistics*, pp. 567–574. PMLR, 2009.

van Der Vaart, A. and van Zanten, H. Information rates of nonparametric Gaussian process methods. *Journal of Machine Learning Research*, 12:2095–2119, 2011.

Vehtari, A., Mononen, T., Tolvanen, V., Sivula, T., and Winther, O. Bayesian leave-one-out cross-validation approximations for gaussian latent variable models. *Journal of Machine Learning Research*, 17(103):1–38, 2016.

Von Luxburg, U. and Schölkopf, B. Statistical learning theory: Models, concepts, and results. In *Handbook of the History of Logic*, volume 10, pp. 651–706. Elsevier, 2011.

Wang, W., Tuo, R., and Wu, C. F. J. On prediction properties of kriging: uniform error bounds and robustness. *Journal of the American Statistical Association*, 115(530):920–930, 2020.

Wenger, J., Krämer, N., Pförtner, M., Schmidt, J., Bosch, N., Effenberger, N., Zenn, J., Gessner, A., Karvonen, T., Briol, F.-X., Mahsereci, M., and Hennig, P. ProbNum: Probabilistic numerics in Python. *arXiv:2112.02100*, 2021. URL http://arxiv.org/abs/2112.02100.

Wild, V. D., Hu, R., and Sejdinovic, D. Generalized variational inference in function spaces: Gaussian measures meet Bayesian deep learning. *Advances in Neural Information Processing Systems*, 35:3716–3730, 2022.

Wu, J., Poloczek, M., Gordon Wilson, A., and Frazier, P. I. Bayesian optimization with gradients. In *Advances in Neural Information Processing Systems*, 2017.

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

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

Xu, J., Scealy, J. L., Wood, A. T., and Zou, T. Generalized score matching for regression. *arXiv preprint arXiv:2203.09864*, 2022.

Yousefi, F., Smith, M. T., and Álvarez, M. A. Multi-task learning for aggregated data using Gaussian processes. In *Advances in Neural Information Processing Systems*, 2019.

Yu, S., Drton, M., and Shojaie, A. Generalized score matching for general domains. *Information and Inference*, 11(2):739–780, 2022. ISSN 20498772. doi: 10.1093/imaiai/iaaa041.

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

Robust and Conjugate Gaussian Process Regression: Supplementary Materials

Our supplementary material is structured as follows. In Appendix A, we provide the proofs of all our theoretical results, as well as additional results which complement those in the main text. In Appendix B, we provide additional details on our numerical experiments.

Notation

In this section, we recap the notation used throughout the paper.

- If $g : \mathcal{X} \times \mathcal{Y} \subseteq \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}$, then $\nabla_y g = \dfrac{\partial g}{\partial y}$ is the partial derivative of g with respect to y.
- $X \sim \mathcal{N}(\mu, \Sigma)$ denotes that X is has a multivariate Gaussian distribution with mean μ and covariance Σ. Furthermore, $p(x) = \mathcal{N}(x; \mu, \Sigma)$ denotes that p is the density of this multivariate Gaussian distribution.
- For some d-dimensional vector $\mathbf{v} = (v_1, \ldots, v_d)^\top$, $\mathrm{diag}(\mathbf{v})$ is the $d \times d$ diagonal matrix D so that $D_{ij} = 0$ if $i \neq j$ and $D_{ii} = v_i$. In addition, exponents are applied entry-wise; e.g. $\mathbf{v}^2 = (v_1^2, \ldots, v_d^2)^\top$.
- Tr denotes the trace operator, which for some $d \times d$ matrix A is given by: $\mathrm{Tr}(A) = \sum_{i=1}^d A_{ii}$.
- A positive-definite kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a symmetric function, i.e. $k(x, x') = k(x', x)$ for any x, x', that is positive-definite, i.e. $\forall \{x_1, ..., x_n\} \subset \mathcal{X}$, $\forall \{c_1, ..., c_n\} \subset \mathbb{R}$ and $\forall n \in \mathbb{N}$: $\sum_{i=1}^n \sum_{j=1}^n c_i c_j k(x_i, x_j) \geq 0$, which equivalent that for any n points the $n \times n$ matrix given by $k(x, x)$ is positive-semidefinite. According to the Moore-Aronszajn theorem, every positive-definite kernel is also a reproducing kernel, and according to Loeve's theorem, any reproducing kernel is the covariance function of a second-order stochastic process and vice-versa.
- Let $X \sim p$ be a random variable distributed according to the probability density p. We define the expectation and the variance of the random variable X as

$$\mathbb{E}_{X \sim p}[X] = \int_{\mathbb{R}^n} x p(x) dx, \qquad \mathbb{V}_{X \sim p}[X] = \int_{\mathbb{R}^n} (x - \mathbb{E}_{X \sim p}[X])(x - \mathbb{E}_{X \sim p}[X])^\top p(x) dx$$

A. Proofs of Theoretical Results

A.1. Proof of Proposition 3.1

Recall the loss function defined in Section 3:

$$L_n^w(\mathbf{f}, \mathbf{y}, \mathbf{x}) = \frac{1}{n} \sum_{i=1}^n \left((w s_{\text{model}})^2 + 2 \nabla_y (w^2 s_{\text{model}}) \right)(x_i, y_i).$$

Here, $s_{\text{model}} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is the score function, and $w : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is the weighting function. Proposition 3.1 gives the posterior and posterior predictive distributions for RCGPs, and we now derive these one-by-one.

Posterior Distribution Firstly, we show that the RCGP posterior has density $p^w(\mathbf{f}|\mathbf{y}, \mathbf{x}) = \mathcal{N}(\mathbf{f}; \mu^R, \Sigma^R)$,

$$\mu^R = \mathbf{m} + K \left(K + \sigma^2 J_{\mathbf{w}} \right)^{-1} (\mathbf{y} - \mathbf{m}_{\mathbf{w}}), \qquad \Sigma^R = K \left(K + \sigma^2 J_{\mathbf{w}} \right)^{-1} \sigma^2 J_{\mathbf{w}},$$

where $\mathbf{w} = (w(x_1, y_1), \ldots, w(x_n, y_n))^\top$, $\mathbf{m}_{\mathbf{w}} = \mathbf{m} + \sigma^2 \nabla_y \log(\mathbf{w}^2)$, $J_{\mathbf{w}} = \mathrm{diag}(\frac{\sigma^2}{2} \mathbf{w}^{-2})$.

Proof. Firstly, it is worth noting that for $s_{\text{model}}(x,y) = (f(x)-y)\sigma^{-2}$, the loss function $L_n^{\widetilde{w}}$ constructed with the weight function w can be written as

$$
\begin{aligned}
L_n^{w}(\mathbf{f},\mathbf{y},\mathbf{x}) &= \frac{1}{n}\sum_{i=1}^{n}\frac{w(x_i,y_i)^2(f(x_i)-y_i)^2}{\sigma^4} + 2\nabla_y\left(\frac{w(x_i,y_i)^2(f(x_i)-y_i)}{\sigma^2}\right)\\
&= \frac{1}{n}\sum_{i=1}^{n}\frac{w(x_i,y_i)^2(f(x_i)^2-2f(x_i)y_i+y_i^2)}{\sigma^4} + 2\nabla_y\left(\frac{w(x_i,y_i)^2(f(x_i)-y_i)}{\sigma^2}\right)\\
&= \frac{1}{n}\sum_{i=1}^{n}\frac{w(x_i,y_i)^2(f(x_i)^2-2f(x_i)y_i)}{\sigma^4} + 2\nabla_y\left(\frac{w(x_i,y_i)^2 f(x_i)}{\sigma^2}\right) + \frac{w(x_i,y_i)^2y_i^2}{\sigma^4} - 2\nabla_y\left(\frac{w(x_i,y_i)^2 y_i}{\sigma^2}\right).
\end{aligned}
$$

Let us consider $\widetilde{w}(x,y) = \sqrt{2\sigma^{-2}}w(x,y)$, then

$$
\begin{aligned}
L_n^{\widetilde{w}}(\mathbf{f},\mathbf{y},\mathbf{x}) &= \frac{1}{n}\sum_{i=1}^{n}\frac{\widetilde{w}(x_i,y_i)^2(f(x_i)^2-2f(x_i)y_i)}{2\sigma^2} + f(x_i)\nabla_y\widetilde{w}(x_i,y_i)^2 + \frac{\widetilde{w}(x_i,y_i)^2y_i^2}{2\sigma^2} - \nabla_y(\widetilde{w}(x_i,y_i)^2y_i)\\
&= \frac{1}{2n}\left(\mathbf{f}^\top\sigma^{-2}\operatorname{diag}(\widetilde{\mathbf{w}}^2)\mathbf{f} - 2\mathbf{f}^\top\left(\sigma^{-2}\operatorname{diag}(\widetilde{\mathbf{w}}^2)\mathbf{y} - \nabla_y\widetilde{\mathbf{w}}^2\right) + \mathbf{y}^\top\sigma^{-2}\operatorname{diag}\widetilde{\mathbf{w}}^2)\mathbf{y} - 2\nabla_y(\mathbf{y}^\top\widetilde{\mathbf{w}}^2)\right)\\
&= \frac{1}{2n}\left(\mathbf{f}^\top\sigma^{-2}\operatorname{diag}(\widetilde{\mathbf{w}}^2)\mathbf{f} - 2\mathbf{f}^\top\sigma^{-2}\operatorname{diag}(\widetilde{\mathbf{w}}^2)\left(\mathbf{y} - \sigma^2\nabla_y\log(\widetilde{\mathbf{w}}^2)\right) + \mathbf{y}^\top\sigma^{-2}\operatorname{diag}(\widetilde{\mathbf{w}}^2)\mathbf{y} - 2\nabla_y(\mathbf{y}^\top\widetilde{\mathbf{w}}^2)\right)\\
&= \frac{1}{2n}\left(\mathbf{f}^\top\sigma^{-2}J_{\mathbf{w}}^{-1}\mathbf{f} - 2\mathbf{f}^\top\sigma^{-2}J_{\mathbf{w}}^{-1}(\mathbf{y}-\mathbf{m_w}+\mathbf{m}) + C(\mathbf{x},\mathbf{y},\sigma^2)\right)
\end{aligned}
$$

Where we use the fact that $\nabla_y\widetilde{\mathbf{w}}^2 = \sigma^{-2}\operatorname{diag}(\widetilde{\mathbf{w}}^2)\times\sigma^2\nabla_y\log(\widetilde{\mathbf{w}}^2)$, and

$$
\begin{aligned}
\mathbf{m_w} &= \mathbf{m} + \sigma^2\nabla_y\log(\widetilde{\mathbf{w}}^2) = \mathbf{m} + \sigma^2\nabla_y\log(\mathbf{w}^2)\\
J_{\mathbf{w}} &= \operatorname{diag}(\widetilde{\mathbf{w}}^{-2}) = \operatorname{diag}(\frac{\sigma^2}{2}\mathbf{w}^{-2})\\
C(\mathbf{x},\mathbf{y},\sigma^2) &= \mathbf{y}^\top\sigma^{-2}\operatorname{diag}(\widetilde{\mathbf{w}}^2)\mathbf{y} - 2\nabla_y\mathbf{y}^\top\widetilde{\mathbf{w}}^2 = \mathbf{y}^\top\sigma^{-2}\operatorname{diag}(2\sigma^{-2}\mathbf{w}^2)\mathbf{y} - 4\sigma^2\nabla_y\mathbf{y}^\top\mathbf{w}^2
\end{aligned}
$$

One remark is that $C(\mathbf{x},\mathbf{y},\sigma^2)$ does not depend on $\mathbf{f}$. Thus, it will not have an impact on the posterior.

Now, we can calculate the density of the generalised posterior of $\mathbf{f}$ using the loss function defined before as follows

$$
\begin{aligned}
p_w(\mathbf{f}|\mathbf{y}) &\propto p(\mathbf{f})\exp\{-nL_n^w(\mathbf{f},\mathbf{y},\mathbf{x})\}\\
&\propto \exp\left(-\frac{1}{2}(\mathbf{f}-\mathbf{m})^\top K^{-1}(\mathbf{f}-\mathbf{m})\right)\exp\left(-\frac{1}{2}\left(\mathbf{f}^\top\sigma^{-2}J_{\mathbf{w}}^{-1}\mathbf{f} - 2\mathbf{f}^\top\sigma^{-2}J_{\mathbf{w}}^{-1}(\mathbf{y}-\mathbf{m_w}+\mathbf{m})\right)\right)\\
&= \exp\left(-\frac{1}{2}\left((\mathbf{f}-\mathbf{m})^\top K^{-1}(\mathbf{f}-\mathbf{m}) + \mathbf{f}^\top\sigma^{-2}J_{\mathbf{w}}^{-1}\mathbf{f} - 2\mathbf{f}^\top\sigma^{-2}J_{\mathbf{w}}^{-1}(\mathbf{y}-\mathbf{m_w}+\mathbf{m})\right)\right)\\
&\propto \exp\left(-\frac{1}{2}\left(\mathbf{f}^\top K^{-1}\mathbf{f} - 2\mathbf{f}^\top K^{-1}\mathbf{m} + \mathbf{f}^\top\sigma^{-2}J_{\mathbf{w}}^{-1}\mathbf{f} - 2\mathbf{f}^\top\sigma^{-2}J_{\mathbf{w}}^{-1}(\mathbf{y}-\mathbf{m_w}+\mathbf{m})\right)\right)\\
&\propto \exp\left(-\frac{1}{2}\left(\mathbf{f}^\top(K^{-1}+\sigma^{-2}J_{\mathbf{w}}^{-1})\mathbf{f} - 2\mathbf{f}^\top\left((K^{-1}+\sigma^{-2}J_{\mathbf{w}}^{-1})\mathbf{m} + \sigma^{-2}J_{\mathbf{w}}^{-1}(\mathbf{y}-\mathbf{m_w})\right)\right)\right).
\end{aligned}
$$

By completing squares, the posterior has the form

$$
\begin{aligned}
p_w(\mathbf{f}|\mathbf{y}) &\propto \exp\left(-\frac{1}{2}\left((\mathbf{f}-\mu_R)^\top\Sigma_R^{-1}(\mathbf{f}-\mu_R)\right)\right),\\
\Sigma_R &= (K^{-1}+\sigma^{-2}J_{\mathbf{w}}^{-1})^{-1} = K\left(K+\sigma^2J_{\mathbf{w}}\right)^{-1}\sigma^2J_{\mathbf{w}},\\
\mu_R &= \Sigma_R\left((K^{-1}+\sigma^{-2}J_{\mathbf{w}}^{-1})\mathbf{m} + \sigma^{-2}J_{\mathbf{w}}^{-1}(\mathbf{y}-\mathbf{m_w})\right) = \mathbf{m} + K\left(K+\sigma^2J_{\mathbf{w}}\right)^{-1}(\mathbf{y}-m_{\mathbf{w}}),
\end{aligned}
$$

where we use the fact that for two invertible matrices A, B, we have $(A^{-1}+B^{-1})^{-1} = A(A+B)^{-1}B$. One remark is that Σ_R is positive semidefinite, since is the inverse of a sum of positive semidefinite matrices. □

Predictive distribution The RCGP posterior predictive distribution over $f_\star = f(x_\star)$ at $x^\star \in \mathcal{X}$ is a multivariate Gaussian distribution with density

$$\begin{aligned} p^w(f_\star|x_\star, \mathbf{x}, \mathbf{y}) &= \int_{\mathbb{R}^n} p(f_\star|x_\star, \mathbf{f}, \mathbf{x}, \mathbf{y}) p^w(\mathbf{f}|\mathbf{y}, \mathbf{x}) d\mathbf{f} \mathcal{N}(f_\star; \mu_\star^R, \Sigma_\star^R), \\ \mu_\star^R &= m_\star + \mathbf{k}_\star^\top \left(K + \sigma^2 J_\mathbf{w}\right)^{-1} (\mathbf{y} - \mathbf{m}_\mathbf{w}), \\ \Sigma_\star^R &= k_{\star\star} - \mathbf{k}_\star^\top (K + \sigma^2 J_\mathbf{w})^{-1} \mathbf{k}_\star. \end{aligned}$$

Proof. We first derive the predictive for $m(x) = 0$ and then extend it to an arbitrary prior mean m. In order to compute the predictive, we first need the conditional density $p(f_\star|x_\star, \mathbf{f}, \mathbf{x}, \mathbf{y})$. Using the fact that f is a mean-zero GP, the joint distribution of $\mathbf{f}$ and $f_\star$ is

$$\begin{pmatrix} \mathbf{f} \\ f_\star \end{pmatrix} \sim \mathcal{N}\left(0, \begin{pmatrix} K & \mathbf{k}_\star \\ \mathbf{k}_\star^\top & k_{\star\star} \end{pmatrix}\right)$$

where $\mathbf{k}_\star = (k(x_\star, x_1), ..., k(x_\star, x_n))^\top$, and $k_{\star,\star} = k(x_\star, x_\star)$. Therefore, the density of the conditional distribution of a multivariate normal is well known and has the form $p(f_\star|x_\star, \mathbf{f}, \mathbf{x}, \mathbf{y}) = \mathcal{N}(f_\star; \mu, \Sigma)$ where

$$\begin{aligned} \mu &= \mathbf{k}_\star^\top K^{-1} \mathbf{f}, \\ \Sigma &= k_{\star\star} - \mathbf{k}_\star^\top K^{-1} \mathbf{k}_\star. \end{aligned}$$

Let us define $a = K^{-1}\mathbf{k}_\star$, then $\mu = a^\top \mathbf{f}$, and we can write the density of the predictive distribution as follows

$$\begin{aligned} p^w(f_\star|x_\star, \mathbf{x}, \mathbf{y}) &= \int_{\mathbb{R}^n} p(f_\star|x_\star, \mathbf{f}, \mathbf{x}, \mathbf{y}) p^w(\mathbf{f}|\mathbf{y}, \mathbf{x}) d\mathbf{f}. \\ &\propto \int_{\mathbb{R}^n} \exp\left(-\frac{1}{2}\left((f_\star - a^\top \mathbf{f})^\top \Sigma^{-1} (f_\star - a^\top \mathbf{f}) + (\mathbf{f} - \mu^R)^\top (\Sigma^R)^{-1} (\mathbf{f} - \mu^R)\right)\right) d\mathbf{f} \\ &\propto \int_{\mathbb{R}^n} \exp\left(-\frac{1}{2}\left(f_\star \Sigma^{-1} f_\star - 2\mathbf{f}^\top a \Sigma^{-1} f_\star + \mathbf{f}^\top a \Sigma^{-1} a^\top \mathbf{f} + \mathbf{f}^\top (\Sigma^R)^{-1} \mathbf{f} - 2\mathbf{f}^\top (\Sigma^R)^{-1} \mu^R\right)\right) d\mathbf{f} \\ &\propto \exp\left(-\frac{1}{2} f_\star \Sigma^{-1} f_\star\right) \int_{\mathbb{R}^n} \exp\left(-\frac{1}{2}\left(\mathbf{f}^\top ((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)\mathbf{f} - 2\mathbf{f}^\top \left(a\Sigma^{-1} f_\star + (\Sigma^R)^{-1}\mu^R\right)\right)\right) d\mathbf{f}. \end{aligned}$$

where the steps follow from basic arithmetic and taking out all terms which do not depend on $\mathbf{f}$, and $\propto$ indicates we do not consider the normalisation constants. Integrating over $\mathbf{f}$, we get

$$\begin{aligned} &p^w(f_\star|x_\star, \mathbf{x}, \mathbf{y}) \\ &\propto \exp\left(-\frac{1}{2}\left(f_\star \Sigma^{-1} f_\star - (a\Sigma^{-1} f_\star + (\Sigma^R)^{-1}\mu^R)^\top ((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)^{-1} (a\Sigma^{-1} f_\star + (\Sigma^R)^{-1}\mu^R)\right)\right) \\ &\propto \exp\left(-\frac{1}{2}\left(f^{\star\top} (\Sigma^{-1} - \Sigma^{-1} a^\top ((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)^{-1} a\Sigma^{-1}) f_\star - 2f^{\star\top} \Sigma^{-1} a^\top ((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)^{-1} (\Sigma^R)^{-1}\mu^R\right)\right). \end{aligned}$$

Therefore, by completing squares, we obtain $p(f_\star|x_\star, \mathbf{x}, \mathbf{y}) = \mathcal{N}(f_\star; \mu_\star^R, \Sigma_\star^R)$, where,

$$\begin{aligned} \mu_\star^R &= (1 - a^\top ((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)^{-1} a\Sigma^{-1})^{-1} \Sigma\Sigma^{-1} a^\top ((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)^{-1} (\Sigma^R)^{-1}\mu^R, \\ \Sigma_\star^R &= (1 - a^\top ((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)^{-1} a\Sigma^{-1})^{-1} \Sigma. \end{aligned}$$

Now, we expand the terms by arithmetic rules for matrix-vector multiplication, to obtain the final expressions:

$$\begin{aligned} \mu_\star^R &= (1 - a^\top ((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)^{-1} a\Sigma^{-1})^{-1} \Sigma\Sigma^{-1} a^\top ((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)^{-1} (\Sigma^R)^{-1}\mu^R \\ &= (1 - a^\top ((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)^{-1} a\Sigma^{-1})^{-1} (((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)(a^\top)^{-1})(\Sigma^R)^{-1}\mu^R \\ &= (((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)(a^\top)^{-1} - a\Sigma^{-1})^{-1} (\Sigma^R)^{-1}\mu^R \\ &= ((\Sigma^R)^{-1}(a^\top)^{-1} + a\Sigma^{-1} - a\Sigma^{-1})^{-1} (\Sigma^R)^{-1}\mu^R \\ &= ((\Sigma^R)^{-1}(a^\top)^{-1})^{-1} (\Sigma^R)^{-1}\mu^R \\ &= a^\top \mu^R \\ &= \mathbf{k}_\star^\top \left(K + \sigma^2 J_\mathbf{w}\right)^{-1} (\mathbf{y} - m_\mathbf{w}). \end{aligned}$$

The covariance follows the form:

$$\Sigma_\star^R = (\Sigma^{-1} - \Sigma^{-1}a^\top((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)^{-1}a\Sigma^{-1})^{-1},$$

we use the Woodbury matrix identity on the term $((\Sigma^R)^{-1} + a\Sigma^{-1}a^\top)^{-1}$ and by arithmetic rules for matrix-vector multiplication we obtain:

$$\begin{aligned}
\Sigma_\star^R &= (\Sigma^{-1} - \Sigma^{-1}a^\top(\Sigma^R - \Sigma^R a(\Sigma + a^\top\Sigma^R a)^{-1}a^\top\Sigma^R)a\Sigma^{-1})^{-1} \\
&= (\Sigma^{-1} - \Sigma^{-1}a^\top(\Sigma^R a(\Sigma + a^\top\Sigma^R a)^{-1}((\Sigma + a^\top\Sigma^R a)a^{-1} - a^\top\Sigma^R))a\Sigma^{-1})^{-1} \\
&= (\Sigma^{-1} - \Sigma^{-1}a^\top(\Sigma^R a(\Sigma + a^\top\Sigma^R a)^{-1}(\Sigma a^{-1} + a^\top\Sigma^R - a^\top\Sigma^R)a\Sigma^{-1})^{-1} \\
&= (\Sigma^{-1} - \Sigma^{-1}a^\top(\Sigma^R a(\Sigma + a^\top\Sigma^R a)^{-1}\Sigma a^{-1}a\Sigma^{-1})^{-1} \\
&= (\Sigma^{-1} - \Sigma^{-1}a^\top\Sigma^R a(\Sigma + a^\top\Sigma^R a)^{-1})^{-1} \\
&= (\Sigma + a^\top\Sigma^R a)(\Sigma^{-1}(\Sigma + a^\top\Sigma^R a) - \Sigma^{-1}a^\top\Sigma^R a)^{-1} \\
&= \Sigma + a^\top\Sigma^R a.
\end{aligned}$$

Finally, we replace Σ^R and use the Woodbury matrix identity on it, to get:

$$\begin{aligned}
\Sigma_\star^R &= \Sigma + \mathbf{k}_\star^\top K^{-1}(K^{-1} + \sigma^{-2}J_\mathbf{w}^{-1})^{-1}K^{-1}\mathbf{k}_\star \\
&= \Sigma + \mathbf{k}_\star^\top K^{-1}(K - K(K + \sigma^2 J_\mathbf{w})^{-1}K)K^{-1}\mathbf{k}_\star \\
&= \Sigma + \mathbf{k}_\star^\top K^{-1}\mathbf{k}_\star - \mathbf{k}_\star^\top(K + \sigma^2 J_\mathbf{w})^{-1}\mathbf{k}_\star \\
&= k_{\star\star} - \mathbf{k}_\star^\top(K + \sigma^2 J_\mathbf{w})^{-1}\mathbf{k}_\star.
\end{aligned}$$

If the prior mean function $m : \mathcal{X} \to \mathbb{R}$ is non-zero, we only need a minor modification of the derivation above. This involves recognising that if we have a function $f \sim \mathcal{GP}(m, k)$, then $f' = f - m \sim f \sim \mathcal{GP}(0, k)$. Therefore, when we have observed values of f, we can adjust them by subtracting the corresponding prior mean function values, thus obtaining observations of f'. We then perform inference on f', and once we have the posterior on this function, we can simply add the prior mean back to the posterior mean to obtain the posterior estimate for f. In other words, for a general prior mean m the density of the predictive posterior will be:

$$\begin{aligned}
p^w(f_\star|x_\star, \mathbf{x}, \mathbf{y}) &= \mathcal{N}(f_\star; \mu_\star^R, \Sigma_\star^R), \\
\mu_\star^R &= m_\star + \mathbf{k}_\star^\top\left(K + \sigma^2 J_\mathbf{w}\right)^{-1}(\mathbf{y} - \mathbf{m}_\mathbf{w}), \\
\Sigma_\star^R &= k_{\star\star} - \mathbf{k}_\star^\top(K + \sigma^2 J_\mathbf{w})^{-1}\mathbf{k}_\star.
\end{aligned}$$

□

A.2. Pseudo Marginal Likelihood for RCGPs

While maximising the pseudo marginal likelihood for RCGP $p^w(\mathbf{y}|\theta, \sigma^2)$ is ill-posed, it is available in closed form given below.

Proposition A.1. *The pseudo marginal likelihood for RCGPs takes the form*

$$\begin{aligned}
&p^w(\mathbf{y} \mid \mathbf{x}, \theta, \sigma^2) \\
&\quad = \frac{1}{\sqrt{|K||K^{-1} + \sigma^{-2}J_\mathbf{w}^{-1}|}} \exp\left(\frac{1}{2}(\mathbf{y} - m_\mathbf{w})^\top\sigma^{-2}J_\mathbf{w}^{-1}(K^{-1} + \sigma^{-2}J_\mathbf{w}^{-1})^{-1}\sigma^{-2}J_\mathbf{w}^{-1}(\mathbf{y} - m_\mathbf{w}) - C(\mathbf{x}, \mathbf{y}, \sigma^2)\right).
\end{aligned}$$

Proof.

$$\begin{aligned}
p^w(\mathbf{y}|\mathbf{x}, \theta, \sigma^2) &= \int_{\mathbb{R}^n} p(\mathbf{f}|\mathbf{x}, \theta, \sigma^2)\exp\{-nL_n^w(\mathbf{f}, \mathbf{y}, \mathbf{x})\}d\mathbf{f} \\
&= \frac{1}{\sqrt{(2\pi)^n|K|}}\int_{\mathbb{R}^n}\exp\left(-\frac{1}{2}\mathbf{f}^\top K^{-1}\mathbf{f} - \frac{1}{2}\mathbf{f}^\top\sigma^{-2}J_\mathbf{w}^{-1}\mathbf{f} + \mathbf{f}^\top\sigma^{-2}J_\mathbf{w}^{-1}(\mathbf{y} - m_\mathbf{w}) - C(\mathbf{x}, \mathbf{y}, \sigma^2)\right)d\mathbf{f} \\
&= \frac{1}{\sqrt{(2\pi)^n|K|}}\int_{\mathbb{R}^n}\exp\left(-\frac{1}{2}\mathbf{f}^\top(K^{-1} + \sigma^{-2}J_\mathbf{w}^{-1})\mathbf{f} + \mathbf{f}^\top\sigma^{-2}J_\mathbf{w}^{-1}(\mathbf{y} - m_\mathbf{w}) - C(\mathbf{x}, \mathbf{y}, \sigma^2)\right)d\mathbf{f}.
\end{aligned}$$

Integrating over $\mathbf{f}$ now yields

$$
\begin{aligned}
&p^w(\mathbf{y}|\mathbf{x},\theta,\sigma^2)\\
&= \frac{1}{\sqrt{(2\pi)^n|K|}}\frac{\sqrt{(2\pi)^n}}{\sqrt{|K^{-1}+\sigma^{-2}J_{\mathbf{w}}^{-1}|}}\exp\left(\frac{1}{2}(\mathbf{y}-m_{\mathbf{w}})^\top\sigma^{-2}J_{\mathbf{w}}^{-1}(K^{-1}+\sigma^{-2}J_{\mathbf{w}}^{-1})^{-1}\sigma^{-2}J_{\mathbf{w}}^{-1}(\mathbf{y}-m_{\mathbf{w}})-C(\mathbf{x},\mathbf{y},\sigma^2)\right)\\
&= \frac{1}{\sqrt{|K||K^{-1}+\sigma^{-2}J_{\mathbf{w}}^{-1}|}}\exp\left(\frac{1}{2}(\mathbf{y}-m_{\mathbf{w}})^\top\sigma^{-2}J_{\mathbf{w}}^{-1}(K^{-1}+\sigma^{-2}J_{\mathbf{w}}^{-1})^{-1}\sigma^{-2}J_{\mathbf{w}}^{-1}(\mathbf{y}-m_{\mathbf{w}})-C(\mathbf{x},\mathbf{y},\sigma^2)\right).
\end{aligned}
$$

□

A.3. Leave-one-out Cross-validation Predictive

The hyperparameters obtained by the leave-one-out cross-validation predictive posteriors are

$$
\hat{\sigma}^2,\hat{\theta} = \arg\max_{\sigma^2,\theta}\left\{\sum_{i=1}^n \log p^w(y_i|\mathbf{x},\mathbf{y}_{-i},\theta,\sigma^2)\right\},
$$

where $\mathbf{y}_{-i}=(y_1,\ldots,y_{i-1},y_{i+1},\ldots,y_n)$. By Proposition 3.1, $p^w(y_i|\mathbf{x},\mathbf{y}_{-i},\theta,\sigma^2)=\mathcal{N}(\mu_i^R,\sigma_i^R+\sigma^2)$ with

$$
\begin{aligned}
\mu_i^R &= z_i+\mathbf{m}_i-\left[\left(K+\sigma^2J_{\mathbf{w}}\right)^{-1}\mathbf{z}\right]_i[(K+\sigma^2J_{\mathbf{w}})^{-1}]_{ii}^{-1},\\
\sigma_i^R &= [(K+\sigma^2J_{\mathbf{w}})^{-1}]_{ii}^{-1}-\frac{\sigma^4}{2}w(x_i,y_i)^{-2},
\end{aligned}
$$

for $\mathbf{z}=\mathbf{y}-\mathbf{m}_{\mathbf{w}}$ and $\mathbf{z}=(z_1,\ldots z_n)$.

Proof. Without loss of generality, we will derive the predictive for $i=n$, which can be extended for an arbitrary $i\in\{1,...,n\}$ using a permutation matrix. Let $p^w(y_n|\mathbf{x},\mathbf{y}_{-n},\theta,\sigma^2)=\mathcal{N}(\mu_n^R,\sigma_n^R+\sigma^2)$ with

$$
\begin{aligned}
\mu_n^R &= \mathbf{m}_n+K_{1:n-1;n}^\top[K+\sigma^2J_{\mathbf{w}^c}]_{1:n-1;1:n-1}^{-1}\mathbf{z}_{1:n-1},\\
\Sigma_\star^R &= k_{nn}-K_{1:n-1;n}^\top[K+\sigma^2J_{\mathbf{w}^c}]_{1:n-1;1:n-1}^{-1}K_{1:n-1;n}.
\end{aligned}
$$

where $[K+\sigma^2J_{\mathbf{w}}]_{1:n-1;1:n-1}$ denotes the submatrix formed from rows $\{1,...,n-1\}$ and columns $\{1,...,n-1\}$, $K_{1:n-1;n}$ denotes the submatrix formed from n^{th} row and columns $\{1,...,n-1\}$, and k_{nn} denotes the element from n^{th} row and n^{th} columns. Now, we observe that we can write the matrix $K+\sigma^2J_{\mathbf{w}^c}$ as

$$
K+\sigma^2J_{\mathbf{w}} = \begin{pmatrix}[K+\sigma^2J_{\mathbf{w}}]_{1:n-1;1:n-1} & K_{n;1:n-1}\\ K_{1:n-1;n} & K_{nn}+\frac{\sigma^4}{2}w(x_n,y_n)^{-2}\end{pmatrix} = \begin{pmatrix}A & B\\ C & D\end{pmatrix}
$$

where we use A, B, C and D for clarity in the derivation. Applying block matrix inversion it is now notationally cumbersome, but it is easy to show that.

$$
(K+\sigma^2J_{\mathbf{w}})^{-1} = \begin{pmatrix}A^{-1}+A^{-1}B\left(D-CA^{-1}B\right)^{-1}CA^{-1} & -A^{-1}B\left(D-CA^{-1}B\right)^{-1}\\ -\left(D-CA^{-1}B\right)^{-1}CA^{-1} & \left(D-CA^{-1}B\right)^{-1}\end{pmatrix}.
$$

Therefore, we have that

$$
\begin{aligned}
[(K+\sigma^2J_{\mathbf{w}})^{-1}]_{nn} &= \left(D-CA^{-1}B\right)^{-1}\\
&= \left(K_{nn}+\frac{\sigma^4}{2}w(x_n,y_n)^{-2}-K_{1:n-1;n}^\top[K+\sigma^2J_{\mathbf{w}}]_{1:n-1;1:n-1}^{-1}K_{1:n-1;n}\right)^{-1}.
\end{aligned}
$$

This leads to

$$
K_{nn}-K_{1:n-1;n}^\top[K+\sigma^2J_{\mathbf{w}}]_{1:n-1;1:n-1}^{-1}K_{1:n-1;n} = \frac{1}{[(K+\sigma^2J_{\mathbf{w}})^{-1}]_{nn}}-\frac{\sigma^4}{2}w(x_n,y_n)^{-2},
$$

which gives us the desired variance:

$$\sigma_n^R = [(K + \sigma^2 J_{\mathbf{w}})^{-1}]_{nn}^{-1} - \frac{\sigma^4}{2} w(x_n, y_n)^{-2},$$

Now, for the mean, we use again the block matrix inversion to get

$$(K + \sigma^2 J_{\mathbf{w}})^{-1}\mathbf{z} = \begin{pmatrix} A^{-1} + A^{-1}B\left(D - CA^{-1}B\right)^{-1} CA^{-1} & -A^{-1}B\left(D - CA^{-1}B\right)^{-1} \\ -\left(D - CA^{-1}B\right)^{-1} CA^{-1} & \left(D - CA^{-1}B\right)^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{z}_{1:n-1} \\ z_n \end{pmatrix}.$$

Therefore, we have that

$$\begin{aligned} [(K + \sigma^2 J_{\mathbf{w}})^{-1}\mathbf{z}]_n &= -\left(D - CA^{-1}B\right)^{-1} CA^{-1}\mathbf{z}_{1:n-1} + \left(D - CA^{-1}B\right)^{-1} z_n \\ &= \left(D - CA^{-1}B\right)^{-1} (z_n - CA^{-1}\mathbf{z}_{1:n-1}). \end{aligned}$$

Noting that $\left(D - CA^{-1}B\right)^{-1} = [(K + \sigma^2 J_{\mathbf{w}^c})^{-1}]_{nn}$, and replacing A and C by their values we obtain

$$[(K + \sigma^2 J_{\mathbf{w}})^{-1}\mathbf{z}]_n = [(K + \sigma^2 J_{\mathbf{w}})^{-1}]_{nn}(z_n - K_{1:n-1;n}^\top [K + \sigma^2 J_{\mathbf{w}}]_{1:n-1;1:n-1}^{-1}\mathbf{z}_{1:n-1}).$$

Finally, rearranging terms, we note that

$$K_{1:n-1;n}^\top [K + \sigma^2 J_{\mathbf{w}}]_{1:n-1;1:n-1}^{-1}\mathbf{z}_{1:n-1} = z_n - [\left(K + \sigma^2 J_{\mathbf{w}}\right)^{-1}\mathbf{z}]_n [(K + \sigma^2 J_{\mathbf{w}})^{-1}]_{nn}^{-1}$$

Leading to the desired mean function:

$$\mu_n^R = z_n + \mathbf{m}_n - [\left(K + \sigma^2 J_{\mathbf{w}}\right)^{-1}\mathbf{z}]_n [(K + \sigma^2 J_{\mathbf{w}})^{-1}]_{nn}^{-1},$$

□

A.4. Proof of Proposition 3.2

First, define the contamination of the dataset $D = \{(x_i, y_i)\}_{i=1}^n$ by the datum y_m^c as $D_m^c = (D \setminus \{(x_m, y_m)\}) \cup \{(x_m, y_m^c)\}$ for some $m \in \{1, \ldots, n\}$. Let $\mathbf{y} = (y_1, ..., y_n)^\top$ and $\mathbf{y}^c = (y_1, ..., y_{m-1}, y_m^c, y_{m+1}, ..., y_n)^\top$.

PIF for the standard GP GP regression has the PIF for some constant $C_1 \in \mathbb{R}$.

$$\mathrm{PIF}_{\mathrm{GP}}(y_m^c, D) = C_1(y_m - y_m^c)^2,$$

and is not robust: $\mathrm{PIF}_{\mathrm{GP}}(y_m^c, f, D) \to \infty$ as $|y_m^c| \to \infty$.

Proof. Let $p(\mathbf{f}|D) = \mathcal{N}(\mathbf{f}; \mu, \Sigma)$ and $p(\mathbf{f}|D_m^c) = \mathcal{N}(\mathbf{f}; \mu^c, \Sigma_c)$ the uncontaminated and contaminated standard GP posterior respectively. Here,

$$\begin{aligned} \mu &= \mathbf{m} + K(K + \sigma^2 I_n)^{-1}(\mathbf{y} - \mathbf{m}) & \mu_c &= \mathbf{m} + K(K + \sigma^2 I_n)^{-1}(\mathbf{y}^c - \mathbf{m}) \\ \Sigma &= K(K + \sigma^2 I_n)^{-1}\sigma^2 I_n & \Sigma_c &= K(K + \sigma^2 I_n)^{-1}\sigma^2 I_n. \end{aligned}$$

Therefore, the PIF has the form

$$\mathrm{PIF}_{\mathrm{GP}}(y_m^c, D) = \frac{1}{2}\left(\mathrm{Tr}\left(\Sigma_c^{-1}\Sigma\right) - n + (\mu_c - \mu)^\mathsf{T}\left(\Sigma_c\right)^{-1}(\mu_c - \mu) + \ln\left(\frac{\det \Sigma_c}{\det \Sigma}\right)\right).$$

We observe that $\Sigma_c = \Sigma$ since they do not depend on y, so that

$$\begin{aligned} &\mathrm{Tr}\left(\Sigma_c^{-1}\Sigma\right) - n = \mathrm{Tr}\left(I_n\right) - n = n - n = 0, \\ &\ln\left(\frac{\det(\Sigma_c)}{\det(\Sigma)}\right) = \ln\left(\det(\Sigma_c)\det(\Sigma^{-1})\right) = \ln\left(\det(\Sigma_c\Sigma^{-1})\right) = \ln\left(\det(I_n)\right) = 0. \end{aligned}$$

This finally leads to the PIF

$$\mathrm{PIF}_{\mathrm{GP}}(y_m^c, D) = \frac{1}{2}\left((\mu_c - \mu)^\mathsf{T} (\Sigma_c)^{-1} (\mu_c - \mu)\right).$$

We now notice that the term $\mu_c - \mu$ can be written as

$$\begin{aligned}\mu_c - \mu &= (\mathbf{m} + K(K+\sigma^2 I_n)^{-1}(\mathbf{y}-\mathbf{m})) - (\mathbf{m} + K(K+\sigma^2 I_n)^{-1}(\mathbf{y}_c - \mathbf{m})) \\ &= K(K+\sigma^2 I_n)^{-1}(\mathbf{y}-\mathbf{y}^c).\end{aligned}$$

Substituting the relevant expressions for $\mu_c - \mu$ and Σ_c above, we find

$$\begin{aligned}\mathrm{PIF}_{\mathrm{GP}}(y_m^c, D) &= \frac{1}{2}\left((\mathbf{y}-\mathbf{y}^c)^\mathsf{T}(K+\sigma^2 I_n)^{-1}K(K(K+\sigma^2 I_n)^{-1}\sigma^2 I_n)^{-1}K(K+\sigma^2 I_n)^{-1}(\mathbf{y}-\mathbf{y}^c)\right). \\ &= \frac{1}{2}\left((\mathbf{y}-\mathbf{y}^c)^\mathsf{T}(K+\sigma^2 I_n)^{-1}K\sigma^{-2}I_n(\mathbf{y}-\mathbf{y}^c)\right).\end{aligned}$$

Finally, since $\mathbf{y}$ and $\mathbf{y}^c$ are the same except for the mth element, the above expression is equal to

$$\mathrm{PIF}_{\mathrm{GP}}(y_m^c, D) = \frac{1}{2}\left(\left[(K+\sigma^2 I_n)^{-1}K\sigma^{-2}I_n\right]_{mm}(y_m - y_m^c)^2\right).$$

□

PIF for the RCGP For RCGPs with $\sup_{x,y} w(x,y) < \infty$, it holds that

$$\mathrm{PIF}_{\mathrm{RCGP}}(y_m^c, D) \leq C_1(w(x_m, y_m^c)^2 y_m^c)^2 + C_2,$$

for some constants $C_1, C_2 \in \mathbb{R}$. Thus, if $\sup_{x,y}\left\{y \cdot w(x,y)^2\right\} < \infty$, RCGP is robust since $\sup_{y_m^c} |\mathrm{PIF}_{\mathrm{RCGP}}(y_m^c, D)| < \infty$.

Proof. Without loss of generality, we will prove the bound for $m = n$, which can be extended for an arbitrary $m \in \{1, \ldots, n\}$ using a permutation matrix. Let $p^w(\mathbf{f}|D) = \mathcal{N}(\mathbf{f}; \mu^R, \Sigma^R)$ and $p^w(\mathbf{f}|D_m^c) = \mathcal{N}(\mathbf{f}; \mu_R^c, \Sigma_R^c)$ denote the uncontaminated and contaminated standard posterior respectively. Here,

$$\begin{aligned}\mu^R &= \mathbf{m} + K\left(K+\sigma^2 J_{\mathbf{w}}\right)^{-1}(\mathbf{y} - m_{\mathbf{w}}) & \mu_c^R &= \mathbf{m} + K\left(K+\sigma^2 J_{\mathbf{w}^c}\right)^{-1}(\mathbf{y} - m_{\mathbf{w}^c}) \\ \Sigma^R &= K\left(K+\sigma^2 J_{\mathbf{w}}\right)^{-1}\sigma^2 J_{\mathbf{w}} & \Sigma_c^R &= K\left(K+\sigma^2 J_{\mathbf{w}^c}\right)^{-1}\sigma^2 J_{\mathbf{w}^c}.\end{aligned}$$

Here $\mathbf{w}^c = (w(x_1, y_1), ..., w(x_n, y_n^c))^\top$. Therefore, the PIF has the form

$$\mathrm{PIF}_{\mathrm{RCGP}}(y_m^c, D) = \frac{1}{2}\left(\underbrace{\mathrm{Tr}\left((\Sigma_c^R)^{-1}\Sigma^R\right) - n}_{(1)} + \underbrace{\left(\mu_c^R - \mu^R\right)^\mathsf{T}(\Sigma_c^R)^{-1}\left(\mu_c^R - \mu^R\right)}_{(2)} + \underbrace{\ln\left(\frac{\det\Sigma_c^R}{\det\Sigma^R}\right)}_{(3)}\right).$$

Now, we will get a bound for each term in the PIF. The first term can be bound as

$$\begin{aligned}(1) = \mathrm{Tr}\left((\Sigma_c^R)^{-1}\Sigma^R\right) - n &= \mathrm{Tr}\left(\sigma^{-2}J_{\mathbf{w}^c}^{-1}\left(K+\sigma^2 J_{\mathbf{w}^c}\right)\left(K+\sigma^2 J_{\mathbf{w}}\right)^{-1}\sigma^2 J_{\mathbf{w}}\right) - n \\ &\leq \mathrm{Tr}\left(\sigma^{-2}J_{\mathbf{w}^c}^{-1}\left(K+\sigma^2 J_{\mathbf{w}^c}\right)\right)\mathrm{Tr}\left(\left(K+\sigma^2 J_{\mathbf{w}}\right)^{-1}\sigma^2 J_{\mathbf{w}}\right) - n,\end{aligned}$$

where we use the fact that for two positive semidefinite matrices A, B, it holds that $\mathrm{Tr}(AB) \leq \mathrm{Tr}(A)\,\mathrm{Tr}(B)$. Observing that $\left(K+\sigma^2 J_{\mathbf{w}}\right)^{-1}\sigma^2 J_{\mathbf{w}}$ does not depend on the contamination, we can now write $\widetilde{C}_1 = \mathrm{Tr}(\left(K+\sigma^2 J_{\mathbf{w}}\right)^{-1}\sigma^2 J_{\mathbf{w}})$, so that by using the arithmetic rules of traces, we obtain

$$\begin{aligned}(1) &\leq \mathrm{Tr}\left(\sigma^{-2}J_{\mathbf{w}^c}^{-1}\left(K+\sigma^2 J_{\mathbf{w}^c}\right)\right)\widetilde{C}_1 - n \\ &= \mathrm{Tr}\left(\sigma^{-2}J_{\mathbf{w}^c}^{-1}K + I_n\right)\widetilde{C}_1 - n \\ &= (\mathrm{Tr}\left(\sigma^{-2}J_{\mathbf{w}^c}^{-1}K\right) + n)\widetilde{C}_1 - n \\ &= \left(\sum_{i=1}^n \sigma^{-2}w^2(x_i, y_y)K_{ii} + n\right)\widetilde{C}_1 - n.\end{aligned}$$

Finally, since $\sup_{x,y} w(x,y) < \infty$, the entire expression can be bounded by a constant $\widetilde{C}_2$ that does not depend on the contamination y_n^c as

$$(1) \leq \left(\sigma^{-2} \sup_{x,y} w(x,y) \sum_{i=1}^{n} K_{ii} + n\right) \widetilde{C}_1 - n = \widetilde{C}_2.$$

Next, we tackle the second term by noting that

$$(2) = \left(\mu_c^R - \mu^R\right)^{\mathsf{T}} (\Sigma_c^R)^{-1} \left(\mu_c^R - \mu^R\right) \leq \lambda_{\max}((\Sigma_c^R)^{-1}) \|\mu_c^R - \mu^R\|_2^2 \leq \lambda_{\max}((\Sigma_c^R)^{-1}) \|\mu_c^R - \mu^R\|_1^2,$$

where $\lambda_{\max}((\Sigma_c^R)^{-1})$ is the maximum eigenvalue of $(\Sigma_c^R)^{-1}$. Then, expanding $\lambda_{\max}((\Sigma_c^R)^{-1})$ and using Weyl's inequality, we get:

$$\lambda_{\max}((\Sigma_c^R)^{-1}) = \lambda_{\max}(K^{-1} + \sigma^{-2} J_{\mathbf{w}^c}^{-1}) \leq \lambda_{\max}(K^{-1}) + \lambda_{\max}(\sigma^{-2} J_{\mathbf{w}^c}^{-1})).$$

Since $J_{\mathbf{w}^c}^{-1} = \mathrm{diag}((\mathbf{w}^c)^2)$, and $\sup_{x,y} w(x,y) < \infty$, it holds that $\lambda_{\max}(\sigma^{-2} J_{\mathbf{w}^c}^{-1})) = \widetilde{C}_3 < +\infty$, so that we have

$$\lambda_{\max}((\Sigma_c^R)^{-1}) = \lambda_{\max}(K^{-1}) + \widetilde{C}_3 = \widetilde{C}_4.$$

We replace this in the expression for (2) to obtain

$$\begin{aligned}
(2) &\leq \widetilde{C}_4 \|\mu_c^R - \mu^R\|_1^2 \\
&= \widetilde{C}_4 \|\mathbf{m} + K\left(K + \sigma^2 J_{\mathbf{w}^c}\right)^{-1} (\mathbf{y} - m_{\mathbf{w}^c}) - \mathbf{m} - K\left(K + \sigma^2 J_{\mathbf{w}}\right)^{-1} (\mathbf{y} - m_{\mathbf{w}})\|_1^2 \\
&= \widetilde{C}_4 \|K(\left(K + \sigma^2 J_{\mathbf{w}^c}\right)^{-1} (\mathbf{y} - m_{\mathbf{w}^c}) - \left(K + \sigma^2 J_{\mathbf{w}}\right)^{-1} (\mathbf{y} - m_{\mathbf{w}}))\|_1^2.
\end{aligned}$$

Applying Cauchy-Schwartz we obtain:

$$(2) \leq \widetilde{C}_4 \|K\|_F \| \left(K + \sigma^2 J_{\mathbf{w}^c}\right)^{-1} (\mathbf{y} - m_{\mathbf{w}^c}) - \left(K + \sigma^2 J_{\mathbf{w}}\right)^{-1} (\mathbf{y} - m_{\mathbf{w}})\|_1^2,$$

where $\|.\|_F$ denotes the Frobenius norm. Now, let's write $K + \sigma^2 J_{\mathbf{w}^c}$ as the block matrix

$$\begin{aligned}
K + \sigma^2 J_{\mathbf{w}^c} &= \begin{pmatrix} [K + \sigma^2 J_{\mathbf{w}^c}]_{1:n-1;1:n-1} & [K + \sigma^2 J_{\mathbf{w}^c}]_{n;1:n-1} \\ [K + \sigma^2 J_{\mathbf{w}^c}]_{1:n-1;n} & [K + \sigma^2 J_{\mathbf{w}^c}]_{nn} \end{pmatrix} \\
&= \begin{pmatrix} [K + \sigma^2 J_{\mathbf{w}^c}]_{1:n-1;1:n-1} & K_{n;1:n-1} \\ K_{1:n-1;n} & K_{nn} + \sigma^2 w(x_n, y_n^c)^{-2} \end{pmatrix},
\end{aligned}$$

where $[K + \sigma^2 J_{\mathbf{w}^c}]_{1:n-1;1:n-1}$ denotes the submatrix formed from rows $\{1, ..., n-1\}$ and columns $\{1, ..., n-1\}$, and $K_{1:n-1;n}$ denotes the submatrix formed from n^{th} row and columns $\{1, ..., n-1\}$. The second equality holds since $J_{\mathbf{w}^c}$ is diagonal. Because we assumed that the contamination is in the nth term, $[K + \sigma^2 J_{\mathbf{w}^c}]_{1:n-1;1:n-1} = [K + \sigma^2 J_{\mathbf{w}}]_{1:n-1;1:n-1}$, so that

$$K + \sigma^2 J_{\mathbf{w}^c} = \begin{pmatrix} [K + \sigma^2 J_{\mathbf{w}}]_{1:n-1;1:n-1} & K_{n;1:n-1} \\ K_{1:n-1;n} & K_{nn} + \sigma^2 w(x_n, y_n^c)^{-2} \end{pmatrix} = \begin{pmatrix} A & B \\ C & q(y_n^c) \end{pmatrix}$$

where we use A, B, and C for clarity in the derivation and to emphasise that these submatrices do not depend on the contamination and therefore can be treated as constants. Applying block matrix inversion, it is now notationally cumbersome but easy to show that

$$(K + \sigma^2 J_{\mathbf{w}^c})^{-1} = \begin{pmatrix} A^{-1} + A^{-1} B q(y_n^c)^{-1} C A^{-1} & -A^{-1} B q(y_n^c)^{-1} \\ -q(y_n^c)^{-1} C A^{-1} & q(y_n^c)^{-1} \end{pmatrix}.$$

We can now use this to rewrite the matrix-vector product

$$\begin{aligned}
&\left(K + \sigma^2 J_{\mathbf{w}^c}\right)^{-1} (\mathbf{y} - m_{\mathbf{w}^c}) \\
&= \begin{pmatrix} A^{-1} + A^{-1} B q(y_n^c)^{-1} C A^{-1} & -A^{-1} B q(y_n^c)^{-1} \\ -q(y_n^c)^{-1} C A^{-1} & q(y_n^c)^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{y}_{1:n-1} \\ y_n^c \end{pmatrix} \\
&= \begin{pmatrix} (A^{-1} + A^{-1} B q(y_n^c)^{-1} C A^{-1}) \mathbf{y}_{1:n-1} - A^{-1} B q(y_n^c)^{-1} y_n^c \\ -q(y_n^c)^{-1} C A^{-1} \mathbf{y}_{1:n-1} + q(y_n^c)^{-1} y_n^c \end{pmatrix},
\end{aligned}$$

where $\mathbf{y}_{1:n-1} = (y_1, ..., y_{n-1})^\top$. Replicating the same steps for the uncontaminated terms, we similarly obtain:

$$\begin{aligned}&\left(K + \sigma^2 J_{\mathbf{w}}\right)^{-1}(\mathbf{y} - m_{\mathbf{w}})\\ &= \begin{pmatrix}(A^{-1} + A^{-1}Bq(y_n)^{-1}CA^{-1})\mathbf{y}_{1:n-1} - A^{-1}Bq(y_n)^{-1}y_n\\ -q(y_n)^{-1}CA^{-1}\mathbf{y}_{1:n-1} + q(y_n)^{-1}y_n\end{pmatrix}.\end{aligned}$$

Now, we can write

$$\begin{aligned}&\sum_{i=1}^{n-1}\left|\left(\left(K + \sigma^2 J_{\mathbf{w}^c}\right)^{-1}(\mathbf{y} - m_{\mathbf{w}^c}) - \left(K + \sigma^2 J_{\mathbf{w}}\right)^{-1}(\mathbf{y} - m_{\mathbf{w}})\right)_i\right|\\ &= \sum_{i=1}^{n-1}\left|\left(A^{-1}Bq(y_n^c)^{-1}CA^{-1}\mathbf{y}_{1:n-1} - A^{-1}Bq(y_n^c)^{-1}y_n^c - A^{-1}Bq(y_n)^{-1}CA^{-1}\mathbf{y}_{1:n-1} + A^{-1}Bq(y_n)^{-1}y_n\right)_i\right|.\end{aligned}$$

Using the triangle inequality, we can bound this as

$$\begin{aligned}&\leq \sum_{i=1}^{n-1}\left|\left(A^{-1}Bq(y_n^c)^{-1}CA^{-1}\mathbf{y}_{1:n-1}\right)_i\right| + \left|\left(A^{-1}Bq(y_n^c)^{-1}y_n^c\right)_i\right|\\ &\qquad + \underbrace{\left|\left(A^{-1}Bq(y_n)^{-1}CA^{-1}\mathbf{y}_{1:n-1}\right)_i\right| + \left|\left(A^{-1}Bq(y_n)^{-1}y_n\right|\right.}_{\widetilde{C}_5}\\ &= \widetilde{C}_5 + \sum_{i=1}^{n-1}\left|q(y_n^c)^{-1}\right|\left|\left(A^{-1}BCA^{-1}\mathbf{y}_{1:n-1}\right)_i\right| + \left|q(y_n^c)^{-1}y_n^c\right|\left|\left(A^{-1}B\right)_i\right|\\ &= \widetilde{C}_5 + \left|q(y_n^c)^{-1}\right|\sum_{i=1}^{n-1}\left|\left(A^{-1}BCA^{-1}\mathbf{y}_{1:n-1}\right)_i\right| + \left|q(y_n^c)^{-1}y_n^c\right|\sum_{i=1}^{n-1}\left|\left(A^{-1}B\right)_i\right|\end{aligned}$$

where $\widetilde{C}_5$ are all terms that do not depend on the contamination. Now, we can observe that:

$$\left|q(y_n^c)^{-1}\right| = \frac{1}{\left|K_{nn} + \sigma^2 w(x_n, y_n^c)^{-2}\right|} = \frac{\sigma^{-2}w(x_n, y_n^c)^2}{\left|K_{nn}\sigma^{-2}w(x_n, y_n^c)^2 + 1\right|} \leq \sigma^{-2}w(x_n, y_n^c)^2,$$

where the last inequality holds because $K_{nn}\sigma^{-2}w(x_n, y_n^c)^2 > 0$. Therefore, since $\sup_{x,y} w(x, y) < \infty$, we can write

$$\begin{aligned}&\widetilde{C}_5 + \left|q(y_n^c)^{-1}\right|\sum_{i=1}^{n-1}\left|\left(A^{-1}BCA^{-1}\mathbf{y}_{1:n-1}\right)_i\right| + \left|q(y_n^c)^{-1}y_n^c\right|\sum_{i=1}^{n-1}\left|\left(A^{-1}B\right)_i\right|\\ &\leq \widetilde{C}_5 + \left|q(y_n^c)^{-1}\right|\sum_{i=1}^{n-1}\left|\left(A^{-1}BCA^{-1}\mathbf{y}_{1:n-1}\right)_i\right| + \left|q(y_n^c)^{-1}y_n^c\right|\sum_{i=1}^{n-1}\left|\left(A^{-1}B\right)_i\right|\\ &\leq \widetilde{C}_6 + \left|w(x_n, y_n^c)^2 y_n^c\right|\widetilde{C}_7\end{aligned}$$

For $\widetilde{C}_6 = \widetilde{C}_5 + \sigma^{-2}\sup_{x,y} w(x, y)^2 \sum_{i=1}^{n-1}\left|\left(A^{-1}BCA^{-1}\mathbf{y}_{1:n-1}\right)_i\right|$, and $\widetilde{C}_7 = \sigma^{-2}\left|\sum_{i=1}^{n-1}\left|\left(A^{-1}B\right)_i\right|\right.$. Now, similarly for the nth term we have

$$\begin{aligned}&\left|\left(K + \sigma^2 J_{\mathbf{w}^c}\right)^{-1}(\mathbf{y} - m_{\mathbf{w}^c}) - (K + \sigma^2 J_{\mathbf{w}})^{-1}(\mathbf{y} - m_{\mathbf{w}})\right)_n\right|\\ &\qquad = \left| - q(y_n^c)^{-1}CA^{-1}\mathbf{y}_{1:n-1} + q(y_n^c)^{-1}y_n^c + q(y_n)^{-1}CA^{-1}\mathbf{y}_{1:n-1} - q(y_n)^{-1}y_n\right|\\ &\qquad \leq \left|q(y_n^c)^{-1}CA^{-1}\mathbf{y}_{1:n-1}\right| + \left|q(y_n^c)^{-1}y_n^c\right| + \left|q(y_n)^{-1}CA^{-1}\mathbf{y}_{1:n-1}\right| + \left|q(y_n)^{-1}y_n\right|\\ &\qquad \leq \widetilde{C}_8 + \widetilde{C}_9\left|w(x_n, y_n^c)^2 y_n^c\right|,\end{aligned}$$

for $\widetilde{C}_8 = \sigma^{-2}\sup_{x,y} w(x,y)^2\left|CA^{-1}\mathbf{y}_{1:n-1}\right| + \left|q(y_n)^{-1}CA^{-1}\mathbf{y}_{1:n-1}\right| + \left|q(y_n)^{-1}y_n\right|$ and $\widetilde{C}_9 = \sigma^{-2}$. Putting both expressions together, we obtain:

$$\begin{aligned}(2) &\leq \widetilde{C}_4\|K\|_F\|\left(K+\sigma^2 J_{\mathbf{w}^c}\right)^{-1}(\mathbf{y}-m_{\mathbf{w}^c}) - \left(K+\sigma^2 J_{\mathbf{w}}\right)^{-1}(\mathbf{y}-m_{\mathbf{w}})\|_1^2 \\ &\leq \widetilde{C}_4\|K\|_F\|2((\widetilde{C}_6+\widetilde{C}_8)^2 + (\widetilde{C}_7+\widetilde{C}_9)^2(w(x_n,y_n^c)^2y_n^c)^2) \\ &\leq \widetilde{C}_{10} + \widetilde{C}_{11}(w(x_n,y_n^c)^2y_n^c)^2,\end{aligned}$$

where $\widetilde{C}_{10} = \widetilde{C}_4\|K\|_F\|2((\widetilde{C}_6+\widetilde{C}_8)^2$, $\widetilde{C}_{11} = \widetilde{C}_4\|K\|_F\|2(\widetilde{C}_7+\widetilde{C}_9)^2$. Lastly, the third and final term can be rewritten using properties of determinants as

$$\begin{aligned}(3) = \ln\left(\frac{\det\Sigma_c^R}{\det\Sigma^R}\right) &= \ln\left(\frac{\det\left(\left(K+\sigma^2 J_{\mathbf{w}^c}\right)^{-1}\sigma^2 J_{\mathbf{w}^c}\right)}{\det\left(\left(K+\sigma^2 J_{\mathbf{w}}\right)^{-1}\sigma^2 J_{\mathbf{w}}\right)}\right) \\ &= \ln\left(\det\left(\left(K+\sigma^2 J_{\mathbf{w}}\right)\sigma^{-2}J_{\mathbf{w}}^{-1}\right)\det\left(\left(K+\sigma^2 J_{\mathbf{w}^c}\right)^{-1}\right)\det\left(\sigma^2 J_{\mathbf{w}^c}\right)\right).\end{aligned}$$

Here, we defined $\widetilde{C}_{12} = \det\left(\left(K+\sigma^2 J_{\mathbf{w}}\right)\sigma^{-2}J_{\mathbf{w}}^{-1}\right)$ since it does not depend on the contamination, and write

$$\begin{aligned}(3) &= \ln\left(\widetilde{C}_{12}\det\left(\left(K+\sigma^2 J_{\mathbf{w}^c}\right)^{-1}\right)\det\left(\sigma^2 J_{\mathbf{w}^c}\right)\right) \\ &= \ln\left(\widetilde{C}_{12}\frac{\det\left(\sigma^2 J_{\mathbf{w}^c}\right)}{\det\left(K+\sigma^2 J_{\mathbf{w}^c}\right)}\right) \\ &\leq \ln\left(\widetilde{C}_{12}\frac{\det\left(\sigma^2 J_{\mathbf{w}^c}\right)}{\det\left(K\right)+\det\left(\sigma^2 J_{\mathbf{w}^c}\right)}\right),\end{aligned}$$

where in the last inequality, we use the fact that for two positive semidefinite matrices A, B, it also holds that $\det(A+B) \geq \det(A)+\det(B)$. Finally, $\det(K) > 0$ and $\det\left(\sigma^2 J_{\mathbf{w}^c}\right) > 0$, since both are positive definite matrices. Therefore,

$$\frac{\det\left(\sigma^2 J_{\mathbf{w}^c}\right)}{\det\left(K\right)+\det\left(\sigma^2 J_{\mathbf{w}^c}\right)} \leq 1,$$

which leads to

$$(3) \leq \ln\left(\widetilde{C}_{12}\right) = \widetilde{C}_{13}.$$

Finally, putting the three terms together we obtain the desire bound:

$$\mathrm{PIF}_{\mathrm{RCGP}}(y_m^c, D) \leq \widetilde{C}_2 + \widetilde{C}_{10} + \widetilde{C}_{11}(w(x_n,y_n^c)^2y_n^c)^2 + \widetilde{C}_{13} = C_1(w(x_n,y_n^c)^2y_n^c)^2 + C_2$$

where $C_2 = \widetilde{C}_2 + \widetilde{C}_{10} + \widetilde{C}_{11} + \widetilde{C}_{13}$, and $C_1 = \widetilde{C}_{11}$.

□

A.5. Proof of Proposition 4.1

For $f \sim \mathcal{GP}(m,k)$, $\varepsilon \sim \mathcal{N}(0,\sigma^2)$, the RCSVGP posterior is $f \sim \mathcal{GP}(\widetilde{\mu},\widetilde{\Sigma})$, where

$$\begin{aligned}\widetilde{\mu}(x) &= \phi_{\mathbf{u}}(x)^\top\mu_{\mathbf{u}}, \\ \widetilde{\Sigma}(x,x') &= k(x,x') - \phi_{\mathbf{u}}(x)^\top\left(K_{\mathbf{uu}}-\Sigma_{\mathbf{u}}\right)\phi_u(x'), \\ \mu_{\mathbf{u}} &= \mathbf{m} + K_{\mathbf{uu}}P_{\mathbf{u}}^{-1}K_{\mathbf{u}}\sigma^{-2}J_{\mathbf{w}}^{-1}(\mathbf{y}-m_{\mathbf{w}}), \\ \Sigma_{\mathbf{u}} &= K_{\mathbf{uu}}P_{\mathbf{u}}^{-1}K_{\mathbf{uu}},\end{aligned}$$

for $P_{\mathbf{u}} = \left(K_{\mathbf{uu}} + K_{\mathbf{u}}^\top\sigma^{-2}J_{\mathbf{w}}^{-1}K_{\mathbf{u}}\right)$, $[K_{\mathbf{uu}}]_{ij} = k(u_i,u_j)$, $[K_{\mathbf{u}}]_{ij} = k(u_i,x_j)$, $[k_{\mathbf{u}}(x)]_i = k(u_i,x)$, and $\phi_{\mathbf{u}}(x) = K_{\mathbf{uu}}^{-1}k_{\mathbf{u}}(x)$. As a function of $\mathbf{u}$, θ, and σ^2, the corresponding variational objective is

$$\begin{aligned}J(\mathbf{u},\theta,\sigma^2) = \frac{1}{2}\nu^\top K_{\mathbf{u}}^\top Q_{\mathbf{u}}^{-1}K_{\mathbf{u}}\nu + C(\sigma^2) + \frac{1}{2}\log\left(\frac{\det\left(K_{\mathbf{uu}}\right)^2}{\det\left(Q_{\mathbf{u}}\right)}\right) \\ - \mathrm{Tr}\left(\sigma^{-2}J_{\mathbf{w}}^{-\frac{1}{2}}(K - K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1}K_{\mathbf{u}})J_{\mathbf{w}}^{-\frac{1}{2}}\right),\end{aligned}$$

where $Q_{\mathbf{u}} = K_{\mathbf{uu}} + K_{\mathbf{u}}^\top \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}$, $\nu = \sigma^{-2} J_{\mathbf{w}}^{-1}(\mathbf{y} - \mathbf{m}_{\mathbf{w}})$, and $C(\sigma^2) = C(\mathbf{x}, \mathbf{y}, \sigma^2)$ is a function that depends on the data and σ^2.

Proof. Recall that for the standard case, we consider $\mathbf{f}_{\mathbf{u}} = (f(u_1), \ldots, f(u_m))^\top$ as the evaluations corresponding to the inducing input locations $\mathbf{u} = (u_1, ..., u_m)^\top$. Given this, we define the variational distribution as

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

where $q(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \mu, \Sigma)$. We now seek to approximate the exact posterior $p(\mathbf{f}, \mathbf{f}_{\mathbf{u}}|\mathbf{y})$ by the variatonal distribution. We do so by minimising the Kullback-Leibler (KL) divergence between $q(\mathbf{f}, \mathbf{f}_{\mathbf{u}})$ and $p(\mathbf{f}, \mathbf{f}_{\mathbf{u}}|\mathbf{y})$. This is equivalent to maximising the ELBO, which is defined as:

$$\mathrm{ELBO}(\mathbf{u}) = \int \log \frac{\Psi(\mathbf{y}, \mathbf{f}_{\mathbf{u}}) p(\mathbf{f}_{\mathbf{u}})}{q(\mathbf{f}_{\mathbf{u}})} q(\mathbf{f}_{\mathbf{u}}) d\mathbf{f}_{\mathbf{u}},$$

where

$$\Psi(\mathbf{y}, \mathbf{f}_{\mathbf{u}}) = \exp\left(\int_{\mathbb{R}^n} \log(p(\mathbf{y}|\mathbf{f})) p(\mathbf{f}|\mathbf{f}_{\mathbf{u}}) d\mathbf{f} \right).$$

It is straightforward to verify that the optimal variational distribution is

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

Plugging this back into the ELBO, we then get

$$\mathrm{ELBO}(\mathbf{u}, \mu, \Sigma) = \log \left(\int \Psi(\mathbf{y}, \mathbf{f}_{\mathbf{u}}) p(\mathbf{f}_{\mathbf{u}}) d\mathbf{f}_{\mathbf{u}} \right).$$

For the RCSVGP, we replace the standard likelihood $p(\mathbf{y}|\mathbf{f})$, with the pseudo likelihood $\exp\{-nL_n^w(\mathbf{f}, \mathbf{y}, \mathbf{x})\}$, leading to the RCSVGP's optimal variational distribution, which is given by

$$q^w(\mathbf{f}_{\mathbf{u}}) = \frac{\Psi^w(\mathbf{y}, \mathbf{f}_{\mathbf{u}}) p(\mathbf{f}_{\mathbf{u}})}{\int \Psi^w(\mathbf{y}, \mathbf{f}_{\mathbf{u}}) p(\mathbf{f}_{\mathbf{u}}) d\mathbf{f}_u},$$

where

$$\begin{aligned} \log \Psi^w(\mathbf{y}, \mathbf{f}_{\mathbf{u}}) &= \int_{\mathbb{R}^n} \log(\exp\{-nL_n^w(\mathbf{f}, \mathbf{y}, \mathbf{x})\}) p(\mathbf{f}|\mathbf{f}_{\mathbf{u}}) d\mathbf{f} \\ &= \int_{\mathbb{R}^n} -\frac{1}{2} \left(\mathbf{f}^\top \sigma^{-2} J_{\mathbf{w}}^{-1} \mathbf{f} - 2\mathbf{f}^\top \nu + C(\mathbf{x}, \mathbf{y}, \sigma^2) \right) p(\mathbf{f}|\mathbf{f}_{\mathbf{u}}) d\mathbf{f}, \end{aligned}$$

where we used the expression of $L_n^w(\mathbf{f}, \mathbf{y}, \mathbf{x})$ from Appendix A.1 and defined $\nu = \sigma^{-2} J_{\mathbf{w}}^{-1}(\mathbf{y} - \mathbf{m}_{\mathbf{w}})$. Recallling that $C(\mathbf{x}, \mathbf{y}, \sigma^2) = \mathbf{y}^\top \sigma^{-2} \operatorname{diag}(\mathbf{w}^2) \mathbf{y} - 2\nabla_y \mathbf{y}^\top \mathbf{w}^2$.

Recalling that the density of the conditional distribution of $\mathbf{f}$ given $\mathbf{f}_{\mathbf{u}}$ is

$$\begin{aligned} p(\mathbf{f}|\mathbf{f}_{\mathbf{u}}) &= \mathcal{N}(\mathbf{f}; \mu_{\mathbf{f}|\mathbf{f}_{\mathbf{u}}}, \Sigma_{\mathbf{f}|\mathbf{f}_{\mathbf{u}}}), \\ \mu_{\mathbf{f}|\mathbf{f}_{\mathbf{u}}} &= K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1} \mathbf{f}_{\mathbf{u}}, \\ \Sigma_{\mathbf{f}|\mathbf{f}_{\mathbf{u}}} &= K - K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1} K_{\mathbf{u}}, \end{aligned}$$

$[K_{\mathbf{uu}}]_{ij} = k(u_i, u_j)$, $[K_{\mathbf{u}}]_{ij} = k(u_i, x_j)$, $[k_{\mathbf{u}}(x)]_i = k(u_i, x)$. Then, we can write the above expression as:

$$\begin{aligned} \log \Psi^w(\mathbf{y}, \mathbf{f}_{\mathbf{u}}) &= \int_{\mathbb{R}^n} -\frac{1}{2} \mathbf{f}^\top \sigma^{-2} J_{\mathbf{w}}^{-1} \mathbf{f} p(\mathbf{f}|\mathbf{f}_{\mathbf{u}}) d\mathbf{f} + \int_{\mathbb{R}^n} \mathbf{f}^\top \nu p(\mathbf{f}|\mathbf{f}_{\mathbf{u}}) d\mathbf{f} - \int_{\mathbb{R}^n} \frac{1}{2} C(\mathbf{x}, \mathbf{y}) p(\mathbf{f}|\mathbf{f}_{\mathbf{u}}) d\mathbf{f} \\ &= -\frac{1}{2} \int_{\mathbb{R}^n} \mathbf{f}^\top \sigma^{-2} J_{\mathbf{w}}^{-1} \mathbf{f} p(\mathbf{f}|\mathbf{f}_{\mathbf{u}}) d\mathbf{f} + \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f}_{\mathbf{u}})}[\mathbf{f}^\top \nu] - \frac{1}{2} C(\mathbf{x}, \mathbf{y}, \sigma^2) \end{aligned}$$

Recalling that we can write the inner product between two vectors as the trace of their outer product, in this case

$$\mathbf{f}^\top \sigma^{-2} J_{\mathbf{w}}^{-1} \mathbf{f} = (\sigma^{-1} J_{\mathbf{w}}^{-1/2} \mathbf{f})^\top (\sigma^{-1} J_{\mathbf{w}}^{-1/2} \mathbf{f}) = \mathrm{Tr}(\sigma^{-1} J_{\mathbf{w}}^{-1/2} \mathbf{f}\mathbf{f}^\top \sigma^{-1} J_{\mathbf{w}}^{-1/2}).$$

Therefore,

$$\begin{aligned}
\log \Psi^w(\mathbf{y}, \mathbf{f_u}) &= -\frac{1}{2} \int_{\mathbb{R}^n} \mathrm{Tr}(\sigma^{-1} J_{\mathbf{w}}^{-1/2} \mathbf{f}\mathbf{f}^\top \sigma^{-1} J_{\mathbf{w}}^{-1/2}) p(\mathbf{f}|\mathbf{f_u}) d\mathbf{f} + \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}^\top \nu] - \frac{1}{2} C(\mathbf{x}, \mathbf{y}, \sigma^2) \\
&= -\frac{1}{2} \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathrm{Tr}(\sigma^{-1} J_{\mathbf{w}}^{-1/2} \mathbf{f}\mathbf{f}^\top \sigma^{-1} J_{\mathbf{w}}^{-1/2}) | \mathbf{f_u}] + \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}^\top \nu] - \frac{1}{2} C(\mathbf{x}, \mathbf{y}, \sigma^2).
\end{aligned}$$

It is well known that the expectation of the trace is equal to the trace of the expectation so that

$$\begin{aligned}
\log \Psi^w(\mathbf{y}, \mathbf{f_u}) &= -\frac{1}{2} \mathrm{Tr}(\mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\sigma^{-1} J_{\mathbf{w}}^{-1/2} \mathbf{f}\mathbf{f}^\top \sigma^{-1} J_{\mathbf{w}}^{-1/2}]) + \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}^\top \nu] - \frac{1}{2} C(\mathbf{x}, \mathbf{y}, \sigma^2) \\
&= -\frac{1}{2} \mathrm{Tr}(\sigma^{-1} J_{\mathbf{w}}^{-1/2} \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}\mathbf{f}^\top] \sigma^{-1} J_{\mathbf{w}}^{-1/2}) + \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}]^\top \nu - \frac{1}{2} C(\mathbf{x}, \mathbf{y}, \sigma^2).
\end{aligned}$$

Recalling that the variance of a random variable is defined for a random variable X distributed by p as

$$\mathbb{V}_{X \sim p}[X] = \mathbb{E}_{X \sim p}[XX^\top] - \mathbb{E}_{X \sim p}[X] \mathbb{E}_X[X]^\top,$$

which implies that $\mathbb{E}_{X \sim p}[XX^\top] = \mathbb{V}_{X \sim p}[X] + \mathbb{E}_{X \sim p}[X] \mathbb{E}_{X \sim p}[X]^\top$. Therefore, we can use this relationship to write

$$\begin{aligned}
\log \Psi^w(\mathbf{y}, \mathbf{f_u}) &= -\frac{1}{2} \mathrm{Tr}(\sigma^{-1} J_{\mathbf{w}}^{-1/2} \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}\mathbf{f}^\top] \sigma^{-1} J_{\mathbf{w}}^{-1/2}) + \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}(\mathbf{f})^\top \nu - \frac{1}{2} C(\mathbf{x}, \mathbf{y}, \sigma^2) \\
&= -\frac{1}{2} \mathrm{Tr}(\sigma^{-1} J_{\mathbf{w}}^{-1/2} (\mathbb{V}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}] + \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}] \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}]^\top) \sigma^{-1} J_{\mathbf{w}}^{-1/2}) + \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}]^\top \nu - \frac{1}{2} C(\mathbf{x}, \mathbf{y}, \sigma^2).
\end{aligned}$$

Since $p(\mathbf{f}|\mathbf{f_u}) = \mathcal{N}(\mathbf{f}; \mu_{\mathbf{f}|\mathbf{f_u}}, \Sigma_{\mathbf{f}|\mathbf{f_u}})$, we know explicitly $\mathbb{V}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}]$ and $\mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}]$. Plug these expressions in and rearranging terms using traces properties, we obtain the final expression

$$\begin{aligned}
\log \Psi^w(\mathbf{y}, \mathbf{f_u}) &= -\frac{1}{2} \mathrm{Tr}(\sigma^{-1} J_{\mathbf{w}}^{-1/2} (\mathbb{V}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}] + \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}] \mathbb{E}_{\mathbf{f} \sim p(\mathbf{f}|\mathbf{f_u})}[\mathbf{f}]^\top) \sigma^{-1} J_{\mathbf{w}}^{-1/2}) + \mathbb{E}(\mathbf{f}|\mathbf{f_u})^\top \nu - \frac{1}{2} C(\mathbf{x}, \mathbf{y}, \sigma^2) \\
&= -\frac{1}{2} \mathrm{Tr}(\sigma^{-1} J_{\mathbf{w}}^{-1/2} (\Sigma_{\mathbf{f}|\mathbf{f_u}} + \mu_{\mathbf{f}|\mathbf{f_u}} \mu_{\mathbf{f}|\mathbf{f_u}}^\top) \sigma^{-1} J_{\mathbf{w}}^{-1/2}) + \mu_{\mathbf{f}|\mathbf{f_u}}^\top \nu - \frac{1}{2} C(\mathbf{x}, \mathbf{y}, \sigma^2) \\
&= -\frac{1}{2} \mathrm{Tr}(\sigma^{-2} J_{\mathbf{w}}^{-1/2} \Sigma_{\mathbf{f}|\mathbf{f_u}} J_{\mathbf{w}}^{-1/2}) - \frac{1}{2} \mathrm{Tr}(\sigma^{-2} J_{\mathbf{w}}^{-1/2} \mu_{\mathbf{f}|\mathbf{f_u}} \mu_{\mathbf{f}|\mathbf{f_u}}^\top J_{\mathbf{w}}^{-1/2}) + \mu_{\mathbf{f}|\mathbf{f_u}}^\top \nu - \frac{1}{2} C(\mathbf{x}, \mathbf{y}, \sigma^2) \\
&= -\frac{1}{2} \mathrm{Tr}(\sigma^{-2} J_{\mathbf{w}}^{-1/2} \Sigma_{\mathbf{f}|\mathbf{f_u}} J_{\mathbf{w}}^{-1/2}) - \frac{1}{2} \mu_{\mathbf{f}|\mathbf{f_u}}^\top \sigma^{-2} J_{\mathbf{w}}^{-1} \mu_{\mathbf{f}|\mathbf{f_u}} + \mu_{\mathbf{f}|\mathbf{f_u}}^\top \nu - \frac{1}{2} C(\mathbf{x}, \mathbf{y}, \sigma^2).
\end{aligned}$$

We now will plug Ψ in optimal variational distribution to finish the proof. Replacing and rearranging terms, we obtain

$$\begin{aligned}
q^w(\mathbf{f_u}) &\propto \Psi^w(\mathbf{y}, \mathbf{f_u}) p(\mathbf{f_u}) \\
&= \exp\left(-\frac{1}{2} \mu_{\mathbf{f}|\mathbf{f_u}}^\top \sigma^{-2} J_{\mathbf{w}}^{-1} \mu_{\mathbf{f}|\mathbf{f_u}} + \mu_{\mathbf{f}|\mathbf{f_u}}^\top \nu - \frac{1}{2} \mathbf{f_u}^\top K_{\mathbf{uu}}^{-1} \mathbf{f_u} \right) \\
&= \exp\left(-\frac{1}{2} \mathbf{f_u}^\top K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1} \mathbf{f_u} + \mathbf{f_u}^\top K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \nu - \frac{1}{2} \mathbf{f_u}^\top K_{\mathbf{uu}}^{-1} \mathbf{f_u} \right) \\
&= \exp\left(-\frac{1}{2} \mathbf{f_u}^\top (K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1} + K_{\mathbf{uu}}^{-1}) \mathbf{f_u} + \mathbf{f_u}^\top K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \nu \right)
\end{aligned}$$

completing squares, we obtain

$$\begin{aligned}
q^w(\mathbf{f_u}) &\propto \exp\left(-\frac{1}{2} (\mathbf{f_u} - \mu_{\mathbf{u}})^\top \Sigma_S^{-1} (\mathbf{f_u} - \mu_{\mathbf{u}})^\top \right) \\
\mu_{\mathbf{u}} &= (K_{\mathbf{uu}}^{-1} + K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1})^{-1} K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \nu \\
&= K_{\mathbf{uu}} (K_{\mathbf{uu}} + K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top)^{-1} K_{\mathbf{u}} \nu \\
&= K_{\mathbf{uu}} P_{\mathbf{u}}^{-1} K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} (\mathbf{y} - \mathbf{m_w}), \\
\Sigma_{\mathbf{u}} &= (K_{\mathbf{uu}}^{-1} + K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1})^{-1} \\
&= K_{\mathbf{uu}} (K_{\mathbf{uu}} + K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top)^{-1} K_{\mathbf{uu}} \\
&= K_{\mathbf{uu}} P_{\mathbf{u}}^{-1} K_{\mathbf{uu}},
\end{aligned}$$

for $P_{\mathbf{u}} = \left(K_{\mathbf{uu}} + K_{\mathbf{u}}^\top \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}\right)$. Now, for the predictive posterior over $f_\star = f(x_\star)$ at new point $x^\star \in \mathcal{X}$, we have

$$\begin{aligned} p^w(f_\star|, \mathbf{y}, \mathbf{x}) &= \int_{\mathbb{R}} p^w(f_\star|x_\star, \mathbf{f_u}) q^w(\mathbf{f_u}) d\mathbf{f_u} \\ &= \mathcal{N}(f_\star; \mu_{f_\star|\mathbf{u}}, \Sigma_{f_\star|\mathbf{u}}) \mathcal{N}(\mathbf{f_u}; \mu_{\mathbf{u}}, \Sigma_{\mathbf{u}}) d\mathbf{f_u}. \end{aligned}$$

Here, we can use the integral obtain for the predictive in Appendix A.1 to get

$$\begin{aligned} p^w(f_\star|x, \mathbf{y}, \mathbf{x}) &= \mathcal{N}(f_\star; \widetilde{\mu}(x_\star), \widetilde{\Sigma}(x_\star, x_\star)) \\ \widetilde{\mu}(x_\star) &= \phi_{\mathbf{u}}(x_\star)^\top \mu_{\mathbf{u}}, \\ \widetilde{\Sigma}(x_\star, x_\star) &= k(x_\star, x_\star) - \phi_{\mathbf{u}}(x_\star)^\top \left(K_{\mathbf{uu}} - \Sigma_{\mathbf{u}}\right) \phi_u(x_\star), \end{aligned}$$

for $[k_{\mathbf{u}}(x)]_i = k(u_i, x)$, and $\phi_{\mathbf{u}}(x) = K_{\mathbf{uu}}^{-1} k_{\mathbf{u}}(x)$. Finally, plugging Ψ^w in the ELBO and using log and exponential properties, we get

$$\begin{aligned} &\mathrm{ELBO}(\mathbf{u}) = \log\left(\int \Psi^w(\mathbf{y}, \mathbf{f_u}) p(\mathbf{f_u}) d\mathbf{y}\right) \\ = \;& \log\left(\int_{\mathbb{R}^n} \exp\left(-\frac{1}{2}\operatorname{Tr}(\sigma^{-2} J_{\mathbf{w}}^{-1/2} \Sigma_{\mathbf{f}|\mathbf{f_u}} J_{\mathbf{w}}^{-1/2}) - \frac{1}{2}\mu_{\mathbf{f}|\mathbf{f_u}}^\top \sigma^{-2} J_{\mathbf{w}}^{-1} \mu_{\mathbf{f}|\mathbf{f_u}} + \mu_{\mathbf{f}|\mathbf{f_u}}^\top \nu - \frac{1}{2} C(\mathbf{x}, \mathbf{y}, \sigma^2) - \frac{1}{2}\mathbf{f_u}^\top K_{\mathbf{uu}}^{-1} \mathbf{f_u}\right) d\mathbf{f_u}\right) \\ = \;& -\frac{1}{2}(\operatorname{Tr}(\sigma^{-2} J_{\mathbf{w}}^{-1/2}(K - K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1} K_{\mathbf{u}}) J_{\mathbf{w}}^{-1/2}) - C(\mathbf{x}, \mathbf{y}, \sigma^2)) \\ &+ \log\left(\int_{\mathbb{R}^n} \exp\left(-\frac{1}{2}\mathbf{f_u}^\top (K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1} + K_{\mathbf{uu}}^{-1}) \mathbf{f_u} + \mathbf{f_u}^\top K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \nu\right) d\mathbf{f_u}\right) \end{aligned}$$

integrating over $\mathbf{f_u}$ and just arithmetic rules for matrix-vector multiplication, we obtain

$$\begin{aligned} \mathrm{ELBO}(\mathbf{u}) \;=\; & -\frac{1}{2}(\operatorname{Tr}(\sigma^{-2} J_{\mathbf{w}}^{-1/2}(K - K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1} K_{\mathbf{u}}) J_{\mathbf{w}}^{-1/2}) - C(\mathbf{x}, \mathbf{y}, \sigma^2)) \\ & + \log\left(\frac{\exp\left(\frac{1}{2}\nu^\top K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1}(K_{\mathbf{uu}}^{-1} + K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1})^{-1} K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \nu\right)}{\sqrt{\det\left(K_{\mathbf{uu}}^{-1} + K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1}\right)}}\right) \\ =\; & -\frac{1}{2}(\operatorname{Tr}(\sigma^{-2} J_{\mathbf{w}}^{-1/2}(K - K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1} K_{\mathbf{u}}) J_{\mathbf{w}}^{-1/2}) - C(\mathbf{x}, \mathbf{y}, \sigma^2)) \\ & + \log\left(\frac{\exp\left(\frac{1}{2}\nu^\top K_{\mathbf{u}}^\top (K_{\mathbf{uu}} + K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top)^{-1} K_{\mathbf{u}} \nu\right)}{\sqrt{\det\left(K_{\mathbf{uu}}^{-1} + K_{\mathbf{uu}}^{-1} K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1}\right)}}\right) \\ =\; & -\frac{1}{2}(\operatorname{Tr}(\sigma^{-2} J_{\mathbf{w}}^{-1/2}(K - K_{\mathbf{u}}^\top K_{\mathbf{uu}}^{-1} K_{\mathbf{u}}) J_{\mathbf{w}}^{-1/2}) - C(\mathbf{x}, \mathbf{y}, \sigma^2)) \\ & + \frac{1}{2}\nu^\top K_{\mathbf{u}}^\top (K_{\mathbf{uu}} + K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top)^{-1} K_{\mathbf{u}} \nu + \frac{1}{2}\log\left(\frac{\det(K_{\mathbf{uu}})^2}{\det\left(K_{\mathbf{uu}} + K_{\mathbf{u}} \sigma^{-2} J_{\mathbf{w}}^{-1} K_{\mathbf{u}}^\top\right)}\right) \end{aligned}$$

□

A.6. Proof of Proposition 4.2

Let $f \sim \mathcal{GP}(m, k)$, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. The UCB and PI acquisition functions for RCGPs are

$$\begin{aligned} a_{\text{UCB-RCGP}}(x_\star) &= \mu_\star^R + \lambda(\Sigma_\star^R)^{1/2} \\ a_{\text{PI-RCGP}}(x_\star) &= \Phi\Big(\big(\mu_\star^R - f(x_{\text{max}})\big)(\Sigma_\star^R)^{-1/2}\Big) \end{aligned}$$

where Φ is the cdf of a standard normal, and x_{max} is the best solution we have so far.

Proof. Since the RCGP posterior is Gaussian, we can straightforwardly define the upper confidence bound acquisition function as

$$a_{\text{UCB-RCGP}}(x_\star) = \mu_\star^R + \lambda(\Sigma_\star^R)^{1/2}$$

For probability of improvement, suppose that the best solution we have so far is x_{max}. We define the improvement function as:

$$\mathrm{I}(x_\star) = \max(f(x_\star) - f(x_{\text{max}}), 0)$$

we know that $f(x_\star) \sim \mathcal{N}(\mu_\star^R, \Sigma_\star^R)$, therefore, we can rewrite it as $f(x_\star) = \mu_\star^R + \Sigma_\star^R z$, with $z \sim \mathcal{N}(0,1)$, leading to:

$$\mathrm{I}(x_\star) = \max(\mu_\star^R + \Sigma_\star^R z - f(x_{\text{max}}), 0) \quad z \sim \mathcal{N}(0,1)$$

Finally, we define the probabilty of improvement acquisition function as:

$$\begin{aligned} a_{\text{PI-RCGP}}(x_\star) &= \mathbb{P}(\mathrm{I}(x_\star) > 0) \\ &= \mathbb{P}(f(x_\star) > f(x_{\text{max}})) \\ &= \mathbb{P}(z > \big(f(x_{\text{max}}) - \mu_\star^R\big)(\Sigma_\star^R)^{-1/2}) \\ &= \Phi\Big(\big(\mu_\star^R - f(x_{\text{max}})\big)(\Sigma_\star^R)^{-1/2}\Big) \end{aligned}$$

□

B. Additional Experimental Results

All the experiments were running on an Apple M2 Pro CPU with 16 GB of memory.

B.1. Prior mean

This section illustrates how the choice of the prior mean affects our method. In this particular example, we generated data from a Gaussian Process (GP) with a zero mean and a periodic squared exponential kernel, where the length scale and variance are set to 1, and the period is set to 3. We then added 1% of focused contamination near zero. Figure 9 demonstrates the performance of two RCGPs on this dataset: the blue one with a zero prior mean and the brown one with a prior mean selected through fitting a polynomial regression. Since the outliers in this dataset are close to zero, the method with a zero prior mean does not identify them as outliers. In fact, as the outliers align exactly with the prior mean, the corresponding weight assigned to them is the largest. This leads to a non-optimal performance. In contrast, for the brown RCGP, since the prior mean is carefully chosen, the outliers are far from it and, therefore, are down-weighted, resulting in a posterior that matches the true generative process.

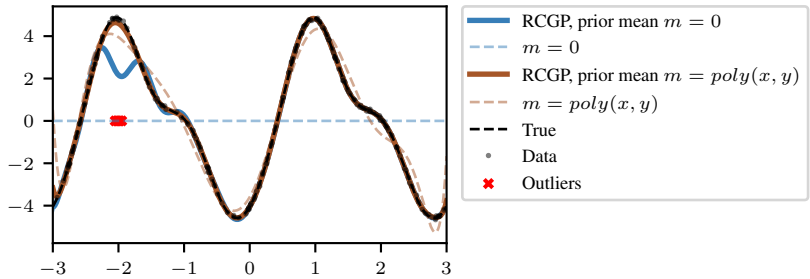

Figure 9. The posterior predictive mean of the RCGP with zero prior mean (**blue**) and the RCGP with a prior mean selected through fitting a polynomial regression (**brown**) on a synthetic dataset where 1% of the data are focused generated outliers.

B.2. Implementation of Competing Methods

GP For the standard GP, we choose hyperparameters via maximum likelihood and utilise the implementation in `GPflow` (Matthews et al., 2017)

t-GP This model replaces the usual choice of Gaussian observation noise with a Student-t distributed observation noise; i.e.

$$p(y|f(x),\nu,\sigma^2) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi\sigma^2}\,\Gamma(\frac{\nu}{2})}\left(1+\frac{(y-f(x))^2}{\nu\sigma^2}\right)^{-(\nu+1)/2}$$

where $\nu > 0$ is the degrees of freedom and σ^2 is the scale parameter. The challenge in this particular scenario lies in the fact that both its posterior and posterior predictive distributions no longer lend analytically tractable, necessitating the use of approximate inference methods. There are several ways to approximate this posterior: MCMC, Laplace's method or variational inference (Jylänki et al., 2011). Our experiments use the variational inference technique implemented in `GPflow` (Matthews et al., 2017) to estimate the posterior, posterior predictive, and hyperparameter selection. [1] In particular, the student-t `GPflow` implementation uses the variational approximation proposed by Opper & Archambeau (2009). The main result of their work is that for a Gaussian process with a non-Gaussian likelihood, the optimal Gaussian approximation — in terms of the Kullback-Leibler divergence — is given by the expression: $q(\mathbf{f}) = \mathcal{N}(\mathbf{K}\boldsymbol{\alpha}, [\mathbf{K}^{-1} + \text{diag}(\boldsymbol{\lambda})]^{-1})$, where $\mathbf{K}$ is the kernel of the GP and $\boldsymbol{\alpha}$ and $\boldsymbol{\lambda}$ are the variational parameters

m-GP This model explicitly considers the generation process of outliers as a uniform distribution over a bounded region– that covers the output y. Then, each observation is associated with a latent variable $z \in \{0,1\}$, where $z = 0$ indicates that the observation is generated by outlier distribution and $z = 1$ inlier. Therefore, the observation model is

$$p(y,z|f(x),\gamma,\sigma^2) = \left((1-\gamma)\frac{1}{a}\right)^{1-z}\left(\gamma\mathcal{N}(y;f(x),\sigma^2)\right)^{z},$$

where $\gamma = p(z=1)$, and $a > 0$ denotes the volume of the outlier region. The variable γ controls the probability of occurrence of two models, and a Beta prior is assumed. Like t-GP, its posterior and posterior predictive distributions no longer lend analytically tractable; thus, approximation is needed. We follow the official implementation provided in the paper. [2]

B.3. Benchmarking

B.3.1. Description of the Datasets

Synthetic The dataset consists of $n = 300$ samples from a GP with zero mean and squared exponential kernel (length scale and variance equal to 1). Then, we added Gaussian noise $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, 0.3)$ to the observations. For the experiments, we selected a GP prior with mean function $m(x) = \frac{1}{n}\sum_{i=1}^{n} y_i$, and squared exponential kernel as covariance function.

Boston The dataset consists of $n = 506$ observations, each representing a suburban or town area in Boston. It encompasses $d = 13$ features containing data like the average number of rooms in dwellings, pupil-teacher ratios, and per capita crime rates. We try to predict the median price of homes residents own (excluding rented properties). The dataset can be found at `https://www.cs.toronto.edu/~delve/data/boston/bostonDetail.html`. We selected a GP prior with mean function $m(x) = \frac{1}{n}\sum_{i=1}^{n} y_i$, and squared exponential kernel as covariance function.

Energy The dataset describes the energy efficiency of buildings by correlating their heating and cooling load requirements with various building parameters. It consists of $n = 768$ data samples, each characterised by $d = 8$ distinct features, with the ultimate goal of predicting a single continuous response variable found in the last column. The dataset can be found at `https://archive.ics.uci.edu/dataset/242/energy+efficiency`. We selected a GP prior with mean function $m(x) = \frac{1}{n}\sum_{i=1}^{n} y_i$, and squared exponential kernel as covariance function.

Yacht The dataset's main focus is on predicting the residuary resistance of sailing yachts during their initial design phase, a critical aspect in evaluating a vessel's performance and estimating the essential propulsive power required. This prediction relies on $d = 6$ primary input parameters, which include the fundamental hull dimensions and boat velocity. The dataset contains $n = 308$ observations. The dataset can be found at `https://archive.ics.uci.edu/dataset/243/`

[1] An official usage example of this implementation can be found here: `https://gpflow.github.io/GPflow/develop/notebooks/getting_started/classification_and_other_data_distributions.html#Non-gaussian-regression`

[2] `https://github.com/YifanLu2000/Robust-Scalable-GPR`

`yacht+hydrodynamics`. We selected a GP prior with mean function $m(x) = \frac{1}{n}\sum_{i=1}^{n} y_i$, and squared exponential kernel as covariance function.

B.3.2. Description of the Outlier Generation Process

While contamination can occur in both the covariate x and the observation y, our work is focused on cases where contamination exclusively affects the observations y. We now describe the three outlier generation processes presented in this paper.

Uniform In this setting, our initial step involves uniformly selecting a specified proportion of the dataset that will be contaminated. To uniformly contaminate the selection, we did a random 50-50 split of this subset: half of the selected subset is contaminated by adding $z \sim U(3\sigma, 9\sigma)$, while the other half is contaminated by subtracting $z \sim U(3\sigma, 9\sigma)$ where σ is the standard deviation of the original observations, and U denotes the uniform distribution.

Asymmetric Much like the uniform outliers, we randomly select a subset of data points that we will contaminate. The key distinction lies in the fact that we do not split the selected subset; instead, we contaminate the entire subset by subtracting $z \sim U(3\sigma, 9\sigma)$ where σ is the standard deviations of the original observations.

Focused In this outlier generation process, we randomly select and remove a subset of data points, which will be replaced by outliers. For these outliers, we deterministically choose their values in $\mathcal{X}$. To do so, we calculate the median value for each input data dimension j. However, we do not place the outliers at this median position directly. Instead, we replace the removed input values by $(m_1 + \delta_1, m_2 + \delta_2 \ldots, m_d + \delta_d)^\top$, where m_j is the median in the j-th input data dimension, and $\delta_j = \alpha_j u$, where α_j is the median absolute deviation of the j-th data dimension times 0.1, and $u \sim U(0, 1)$. Simultaneously, the outlier values on $\mathcal{Y}$ are obtained by subtracting three times the standard deviation of the median of the observations M_y. To not have the same value for every outlier position, we also add a small perturbation $\delta_y = \alpha_y u$, where α_y is the median absolute deviation of $\mathbf{y}$ times 0.1, and $u \sim U(0, 1)$.

B.3.3. Additional Results

We ran all benchmark experiments, choosing c via leave-one-out ('c-LOO'), and compared it with the proposed way to choose c ('c-Q_n'). Overall, the performance is slightly worse for c-LOO—likely because maximising the predictive posterior for extreme observations tends to match/fit these outliers by *increasing* c, leading to a less robust method. Illustrative results can be seen in Table 4.

Table 4. Average test set mean absolute error and standard deviation (in brackets) for 50 train–test splits.

	GP	c-LOO	c-Q_n
	No Outliers		
Synthetic	**0.09 (0.00)**	**0.09 (0.00)**	**0.09 (0.00)**
Boston	**0.19 (0.01)**	**0.19 (0.01)**	**0.19 (0.01)**
Energy	0.03 (0.00)	**0.02 (0.00)**	**0.02 (0.00)**
Yacht	0.02 (0.01)	0.02 (0.01)	0.02 (0.01)
	Focused Outliers		
Synthetic	0.19 (0.00)	0.17 (0.00)	**0.15 (0.00)**
Boston	0.23 (0.06)	**0.22 (0.03)**	**0.22 (0.01)**
Energy	0.03 (0.04)	**0.02 (0.00)**	**0.02 (0.00)**
Yacht	0.26 (0.15)	0.11 (0.13)	**0.10 (0.14)**
	Asymmetric Outliers		
Synthetic	1.14 (0.00)	1.08 (0.00)	**0.63 (0.00)**
Boston	0.63 (0.02)	0.61 (0.01)	**0.49 (0.00)**
Energy	0.54 (0.02)	**0.37 (0.10)**	0.44 (0.04)
Yacht	0.54 (0.06)	0.47 (0.03)	**0.35 (0.02)**

B.4. Sparse Variational Gaussian Processes

This section presents a numerical comparison between SVGP and RCSVGP on the four benchmark datasets presented in Appendix B.3, with three outliers regimes: no outliers, focused outliers, asymmetric outliers. For both methods, we selected a GP prior with mean function $m(x) = \frac{1}{n}\sum_{i=1}^{n} y_i$, and squared exponential kernel as covariance function. We consider a fixed noise variance $\sigma^2 = 0.01$. Table 5 shows that RCSVGP outperforms SVGP even in cases without outliers.

Table 5. Average test set mean absolute error and standard deviation (in brackets) for 10 train–test splits.

	SVGP	SVRCGP	SVGP	SVRCGP	SVGP	SVRCGP
	No Outliers		Focused Outliers		Asymmetric Outliers	
Synthetic	**0.08 (0.00)**	**0.08 (0.00)**	0.19 (0.00)	**0.13 (0.00)**	1.32 (0.00)	**1.02 (0.00)**
Boston	0.65 (0.01)	**0.56 (0.18)**	0.64 (0.02)	**0.55 (0.16)**	0.65 (0.04)	**0.64 (0.01)**
Energy	**0.02 (0.00)**	0.03 (0.00)	0.05 (0.05)	**0.05 (0.00**	0.64 (0.05)	**0.09 (0.16)**
Yacht	**0.01 (0.00)**	0.03 (0.00)	0.29 (0.00)	**0.13 (0.02)**	0.65 (0.04)	**0.60 (0.02)**

B.5. Bayesian Optimisation

In the Bayesian optimisation experiment section, we compared RCGPs to GPs and t-GPs on two classical functions: the Six-Hump Camel function and the Branin function. Here, we state the functions explicitly:

$$\textbf{Six-Hump Camel:} \quad g_1(x, x') = \left(4 - 2.1x^2 + \frac{x^4}{3}\right)x^2 + xx' + (4x'^2 - 4)x'^2 \qquad x \in (-2, 2), x' \in (-1, 1)$$

$$\textbf{Branin:} \quad g_2(x, x') = \left(x' - \frac{5.1}{4\pi^2}x^2 + \frac{5}{\pi}x - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x) + 10 \qquad x \in (-5, 10), x' \in (1, 15)$$

$$\textbf{McCormick:} \quad g_3(x, x') = \sin(x + x') + (x - x') + (-1.5x + 2.5x' + 1) \qquad x \in (-1.5, 4), x' \in (-3, 4)$$

$$\textbf{Rosenbrock:} \quad g_4(x, x') = 100(x' - x^2)^2 + (x^2 - 1)^2 \qquad x \in (-5, 10), x' \in (-5, 1)$$

with global minima $g_1(x_\star, x'_\star) = -1.0316$, $g_2(x_\star, x'_\star) = 0.3979$, $g_3(x_\star, x'_\star) = -1.9133$ and $g_4(x_\star, x'_\star) = 0$.

In order to contaminate with outliers, we consider the scenario where, for each function evaluation, there is a 20% chance of being contaminated by an asymmetric outlier generated as Appendix B.3.2. In order to not change the global minimum, we consider the case where the outlier is bigger than the actual observation, i.e. asymmetric with adding a value.

We selected a GP prior with zero mean function and squared exponential kernel for the experiments for the three models. Figure 10 shows the results using PI as the acquisition function. In terms of cumulative regret, RCGPs outperform GPs. While t-GPs can match this, they take orders of magnitude longer to run.

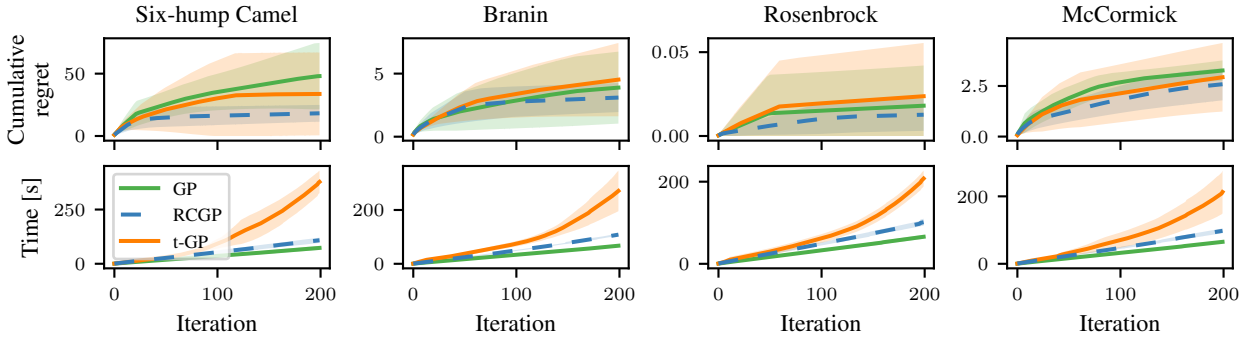

Figure 10. Mean cumulative regret (top) and clock time (bottom) for BO with GP (green), RCGP (blue) and t-GP (orange) with 20% asymmetric outliers and PI acquisition function over 10 realisations.

Figure 11 shows the results without outliers. In terms of cumulative regret, RCGPs match or sometimes outperform GPs. While t-GPs can match this, they take orders of magnitude longer to run. It is notable that even without outliers.

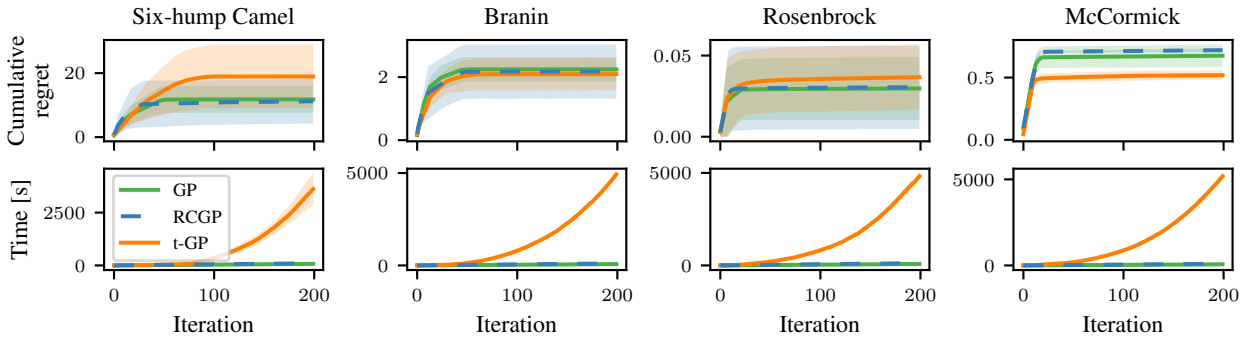

Figure 11. Mean cumulative regret (top) and clock time (bottom) for BO with GP (green), RCGP (blue) and t-GP (orange) without outliers over 10 realisations.