Revisiting Penalized Likelihood Estimation for Gaussian Processes

Ayumi Mutoh* Annie S. Booth† Jonathan W. Stallrich‡

November 25, 2025

Abstract

Gaussian processes (GPs) are popular as nonlinear regression models for expensive computer simulations, yet GP performance relies heavily on estimation of unknown covariance parameters. Maximum likelihood estimation (MLE) is common, but it can be plagued by numerical issues in small data settings. The addition of a nugget helps but is not a cure-all. Penalized likelihood methods may improve upon traditional MLE, but their success depends on tuning parameter selection. We introduce a new cross-validation (CV) metric called "decorrelated prediction error" (DPE), within the penalized likelihood framework for GPs. Inspired by the Mahalanobis distance, DPE provides more consistent and reliable tuning parameter selection than traditional metrics like prediction error, particularly for K-fold CV. Our proposed metric performs comparably to standard MLE when penalization is unnecessary and outperforms traditional tuning parameter selection metrics in scenarios where regularization is beneficial, especially under the one-standard error rule.

Keywords: computer experiments; cross validation; emulator; maximum likelihood estimation; Mahalanobis distance; surrogate

1 Introduction

Computer experiments (or "simulators") are valuable tools in the study of complex physical processes, offering an alternative to expensive or time-consuming physical experiments. Applications span engineering (Chen et al., 2006) including automotive crash analysis (Berthelson et al., 2021), building design (Westermann and Evins, 2019), and the study of natural hazards (Bayarri et al., 2009). The computational complexity of computer experiments often limits the number of times the simulator can be evaluated in practice. Limited simulation data necessitates a surrogate model (or "emulator") to stand-in-place of true simulator evaluations at unobserved input settings. A good surrogate should be flexible, interpretable, and quick to evaluate. Most importantly, it should provide accurate predictions with appropriate uncertainty quantification (UQ), but this is a tall order when training data is scarce.

While a variety of surrogate modeling approaches exist (e.g., Chen et al., 2006; Levy and Steinberg, 2010; Kudela and Matousek, 2022), Gaussian processes (GPs) have risen to the forefront (Sacks et al., 1989; Santner et al., 2003; Gramacy, 2020). The covariance function is the workhorse of any GP surrogate. It specifies the relationship between response values as a function of their input locations. Typical covariance functions are inverse functions of Euclidean distance, further parameterized by a scale (or "variance") and lengthscale. Proper estimation of these hyperparameters can make-or-break a GP surrogate. If the scale is too small/large, the surrogate may underestimate/overestimate uncertainty, respectively. If lengthscales are too "smooth", they may break interpolation of training data which is essential when the computer experiment is known to be deterministic. If lengthscales are too "wiggly," the surrogate will be uninformative, reverting to prior beliefs at unobserved inputs.

While cross validation (CV) methods and Bayesian posterior sampling are possible (e.g., Geisser and Eddy, 1979; MacKay, 1992), maximum likelihood estimation (MLE) of covariance hyperparameters is usually favored for its simplicity and computational efficiency. As evidence of MLE's popularity, most popular

*Corresponding author: Department of Statistics, NC State University, amutoh@ncsu.edu
†Department of Statistics, Virginia Tech
‡Department of Statistics, NC State University

GP software packages adopt MLE as their default estimation method. These include: `GPML` (Rasmussen and Nickisch, 2010) and `DACE` (Nielsen et al., 2002) in MATLAB; `Scikit-learn` (Pedregosa et al., 2011), `GPy` (GPy, 2012), and `GPflow` (Matthews et al., 2016) in Python; and `GPfit` (MacDoanld et al., 2015), `DiceKriging` (Roustant et al., 2012b), and `mlegp` (Dancik, 2013) in R.

Unfortunately, convergence issues can arise when numerically maximizing the GP likelihood, especially with small data sizes. Tricks to circumvent these issues in the GP likelihood include reparameterizing the covariance function or bounding the possible values of the parameters (e.g., MacDoanld et al., 2015; Butler et al., 2014; Basak et al., 2021; Binois and Gramacy, 2021). If convergence issues in the GP likelihood stem from a nearly-singular covariance matrix, they may be allayed by the addition of a nugget (or "jitter") along the diagonal (Gramacy and Lee, 2012). Alternatively, some works employ the Moore-Penrose inverse (Moore, 1920) for singular covariance matrices (Li and Sudjianto, 2005; Lowe et al., 2025). Avoiding nearly-singular covariance matrices is important to ensure the GP likelihood can be evaluated for all possible hyperparameters, but it does not address other convergence issues caused by multi-modal or flat likelihood surfaces.

Penalized likelihood estimation is a common tool to reign in the likelihood of linear models (Tibshirani, 1996; Hoerl and Kennard, 1970; Fu, 1998), but it has not received the same attention for GPs. Li and Sudjianto (2005) proposed a penalized GP likelihood estimator to discourage extreme lengthscale estimates with a tuning parameter selection strategy that minimized prediction error via CV. Even though this approach has merit, it has not been adopted in the literature or by practitioners. For example, the R package `DiceKriging` (Roustant et al., 2012b) offers penalized GP estimation, but the authors of the package did not find performance sufficiently convincing (Roustant et al., 2012a).

In this paper, we revisit penalized likelihood estimation for GPs with limited data, outlining potential pitfalls in MLE and penalized MLE approaches. We show that the nugget is a crucial ingredient for stable computations for penalized MLE, but that it can also complicate tuning parameter selection done via CV with conventional metrics like prediction error and Mahalanobis distance. This complication is resolved through a new CV metric that we call decorrelated prediction error (DPE). We show that CV based on DPE offers more consistent and reliable tuning parameter selection than existing approaches, often producing a surrogate that either matches or outperforms that under the MLE.

1.1 Motivating Examples

To motivate our contribution, we present two synthetic exercises. From the first, we will justify the addition of a nugget in place of a pseudoinverse and will demonstrate the benefit of penalization. From the second, we will highlight a flaw in traditional tuning parameter selection methods, which our proposed metric rectifies.

The following examples warrant some explanation of GP fundamentals. Let $y_i = f(\mathbf{x}_i)$ denote the i^{th} observation of a deterministic, real-valued, black-box function f at some d-dimensional input $\mathbf{x}_i$. Likewise, let $\mathbf{y}_n = f(\mathbf{X}_n)$ denote the corresponding vector of n observations at row-stacked input locations $\mathbf{X}_n$ of dimension $n \times d$. A GP prior over f presumes $\mathbf{y}_n \sim \mathcal{N}_n(\boldsymbol{\mu}, \boldsymbol{\Sigma}_n)$. Prior mean $\boldsymbol{\mu}$ may be constant or a function of $\mathbf{X}_n$; we will use $\boldsymbol{\mu} = \mathbf{0}$ after centering responses. Covariance $\boldsymbol{\Sigma}_n = \sigma^2\mathbf{R}_n$ is formed from the "kernel" function $R(\mathbf{x}_i, \mathbf{x}_j)$ which provides the correlation between y_i and y_j. Kernel functions typically involve one or more hyperparameters, denoted by $\boldsymbol{\theta}$, leading to the full set of parameters $\Omega = \{\boldsymbol{\theta}, \sigma^2\}$. Given $\mathcal{D} = \{\mathbf{X}_n, \mathbf{y}_n\}$, predictions for f at m input locations $\mathcal{X}$ follow

$$
\begin{aligned}
f(\mathcal{X} \mid \mathcal{D}) &\sim \mathcal{N}_m\left(\, \boldsymbol{\mu}_\theta(\mathcal{X} \mid \mathcal{D}), \boldsymbol{\Sigma}_\Omega(\mathcal{X} \mid \mathcal{D}) \,\right) \\
\boldsymbol{\mu}_\theta(\mathcal{X} \mid \mathcal{D}) &= \mathbf{R}(\mathcal{X}, \mathbf{X}_n)\mathbf{R}_n^{-1}\mathbf{y}_n \\
\boldsymbol{\Sigma}_\Omega(\mathcal{X} \mid \mathcal{D}) &= \sigma^2\left(\mathbf{R}(\mathcal{X}, \mathcal{X}) - \mathbf{R}(\mathcal{X}, \mathbf{X}_n)\mathbf{R}_n^{-1}\mathbf{R}(\mathbf{X}_n, \mathcal{X})\right) \\
&:= \sigma^2\mathbf{R}_\theta(\mathcal{X} \mid \mathcal{D})\,,
\end{aligned}
\tag{1}
$$

with $\mathbf{R}(\mathcal{X}, \mathbf{X}_n)$ denoting the $m \times n$ matrix of correlations for all pairs of rows between $\mathcal{X}$ and $\mathbf{X}_n$. The subscripts "θ" and "Ω" emphasize dependencies on the corresponding parameters. We introduce the new notation $\mathbf{R}_\theta(\mathcal{X} \mid \mathcal{D})$ to denote the correlations among the predicted responses conditioned on the training data, which will be integral to our proposed DPE metric. In practice, Ω is estimated, and its values are plugged into $\boldsymbol{\mu}_\theta(\mathcal{X} \mid \mathcal{D})$ and $\boldsymbol{\Sigma}_\Omega(\mathcal{X} \mid \mathcal{D})$.

Estimation of Ω is driven by the GP log likelihood:

$$\log \mathcal{L}(\Omega \mid \mathbf{y}_n) \propto -\frac{n}{2}\log \sigma^2 - \frac{1}{2}\log|\mathbf{R}_n| - \frac{1}{2\sigma^2}\mathbf{y}_n^\top \mathbf{R}_n^{-1}\mathbf{y}_n \ ,$$

where "$\propto$" indicates an additive constant has been dropped. A closed-form MLE exists for the scale, $\hat{\sigma}^2_\theta = \frac{1}{n}\mathbf{y}_n^\top \mathbf{R}_n^{-1}\mathbf{y}_n$, providing the profile log likelihood for $\boldsymbol{\theta}$:

$$\log \mathcal{L}(\boldsymbol{\theta} \mid \mathbf{y}_n, \sigma^2 = \hat{\sigma}^2_\theta) \propto -\frac{n}{2}\log\left(\mathbf{y}_n^\top \mathbf{R}_n^{-1}\mathbf{y}_n\right) - \frac{1}{2}\log|\mathbf{R}_n| \ . \tag{2}$$

The maximum likelihood estimator is $\hat{\boldsymbol{\theta}} = \text{argmax} \log \mathcal{L}(\boldsymbol{\theta} \mid \mathbf{y}_n, \sigma^2 = \hat{\sigma}^2_\theta)$, which must be found numerically. Henceforth we will adopt this profile log likelihood and replace σ^2 in Eq. (1) with $\hat{\sigma}^2_\theta$.

In this section, we consider two functions with $d = 1$ and work with the squared exponential kernel parameterized as $\mathbf{R}_n^{ij} = R(x_i, x_j) = \exp\left(-\theta(x_i - x_j)^2\right)$.[1] The first example duplicates the premier example in Li and Sudjianto (2005): a one dimensional sine curve, $f(x) = \sin(x)$, observed at 6 equally spaced inputs between 0 and 10 as shown in Figure 1 (with inputs scaled to the unit interval, which we will do throughout). The profile log likelihood (Eq. 2) as a function of $\theta \in [0.001, 100]$ is shown by the light blue dashed line in Figure 1a (with θ also on the log scale).

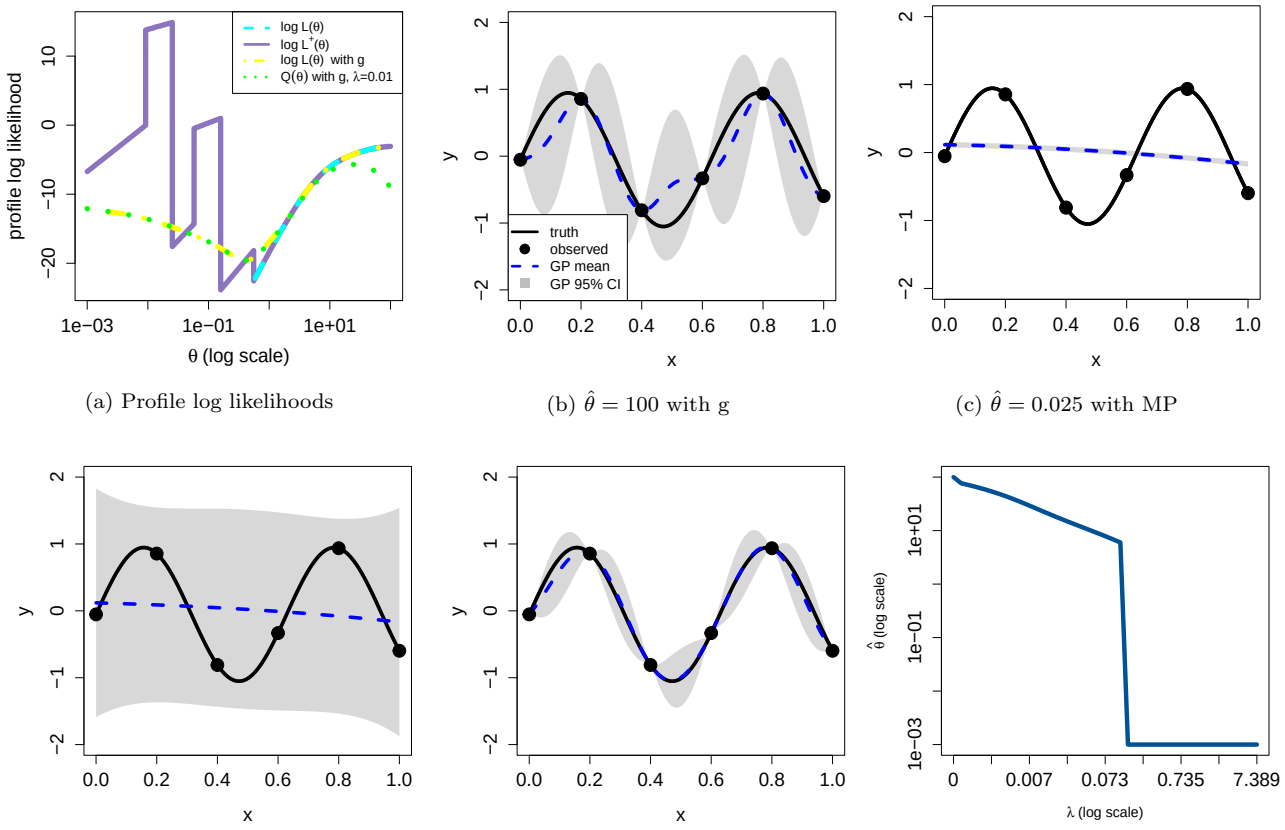
Notice, as θ approaches zero, the likelihood curve $\log \mathcal{L}(\boldsymbol{\theta})$ (light blue dashed line) disappears due to numerical issues encountered during the inversion of $\mathbf{R}_n$. Since our contribution is focused on shrinking θ with strategic penalization, we must pay particular attention to the ramifications of smaller θ values. There are two strategies to avoid this instability: using a pseudoinverse or adding a nugget. Li and Sudjianto (2005) chose the former, using the Moore-Penrose (MP) inverse $\mathbf{R}_n^+$ in place of $\mathbf{R}_n^{-1}$. The profile log likelihood with the MP inverse, $\log \mathcal{L}^+(\boldsymbol{\theta})$, overlayed as the purple solid line in Figure 1a, has a definitive global maximum at $\theta = 0.025$, but the GP resulting from this lengthscale (Figure 1c) is abysmal. It fails to interpolate the training observations or provide effective UQ. This behavior stems from computational issues in evaluating the determinant of $\mathbf{R}_n$ because there are very small but nonzero eigenvalues present. Small eigenvalues should be treated as 0, but it is difficult to choose an appropriate tolerance value. Unsatisfied with the MP inverse, we embrace the addition of a nugget term by upgrading the kernel function to $R(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\theta||\mathbf{x}_i - \mathbf{x}_j||^2\right) + g\mathbb{I}_{(i=j)}$, where g is fixed at a small positive value and $\mathbb{I}_{(i=j)}$ is an indicator function. The nugget ensures $\text{rank}(\mathbf{R}_n) = n$ with eigenvalues bounded below by g. The profile log likelihood with the addition of $g = 1 \times 10^{-5}$ is shown by the yellow dashed-dotted line. The nugget enables computations for small θ. It is also beneficial in preserving stability in $\Sigma_\Omega(\mathcal{X} \mid \mathcal{D})$, so we will use it from here on out. While the addition of a nugget improves numerical stability, it is not a catch-all solution. As evidence, we present the GP fit with the same $\hat{\theta} = 0.025$ but with the nugget in-place of the MP inverse in Figure 1d. The addition of the nugget is not enough to overcome the computational issues of a poorly chosen θ. In fact, it causes the scale estimate $\hat{\sigma}^2$ to blow up, hyperinflating the posterior variance. We will discuss this and further intricacies in Section 3.

The more recognizable issue with the likelihoods of Figure 1a is the flatness of the surface for larger θ. There is no conclusive global maximum to select. We could arbitrarily pick $\hat{\theta} = 100$ as a potential upper bound, but the GP fit resulting from this (shown in Figure 1b) is suboptimal. It is overly "wiggly" with inflated uncertainty. To address the flat likelihood, we employ a penalized log likelihood, defined as

$$Q\left(\boldsymbol{\theta} \mid \mathbf{y}_n, \sigma^2 = \hat{\sigma}^2_\theta\right) = \log \mathcal{L}(\boldsymbol{\theta} \mid \mathbf{y}_n, \sigma^2 = \hat{\sigma}^2_\theta) - np_\lambda(\boldsymbol{\theta}) \ , \tag{3}$$

for some penalty function p_λ that is nondecreasing for $\boldsymbol{\theta} > 0$. The penalty function involves a tuning parameter $\lambda \geq 0$ that controls the relative importance of the penalization; larger λ intend to produce a $\hat{\boldsymbol{\theta}}$ that is shrunk towards 0. The penalized log likelihood $Q(\boldsymbol{\theta})$, for $\lambda = 0.01$ with a LASSO penalty is shown as the green dotted line in Figure 1a, successfully resolving the flat likelihood problem. For a specified λ, the penalized MLE (pMLE) may be obtained as $\hat{\boldsymbol{\theta}} = \text{argmax}\, Q(\boldsymbol{\theta} \mid \mathbf{y}_n, \sigma^2 = \hat{\sigma}^2_\theta)$. The GP resulting from the pMLE for $\lambda = 0.01$ is shown in Figure 1f. This GP provides interpolation of training data, accurate out-of-sample predictions, and effective UQ. Yet in practice, the choice of λ is a difficult problem. Figure 1f

[1] It is common to use the reciprocal of θ here, but that parameterization is not compatible with penalization. In our parameterization, penalties encourage smaller θ and discourage abrupt correlation decay.



(a) Profile log likelihoods

(b) $\hat{\theta} = 100$ with g

(c) $\hat{\theta} = 0.025$ with MP

(d) $\hat{\theta} = 0.025$ with g

(e) $\hat{\theta} = 24.207$ with g

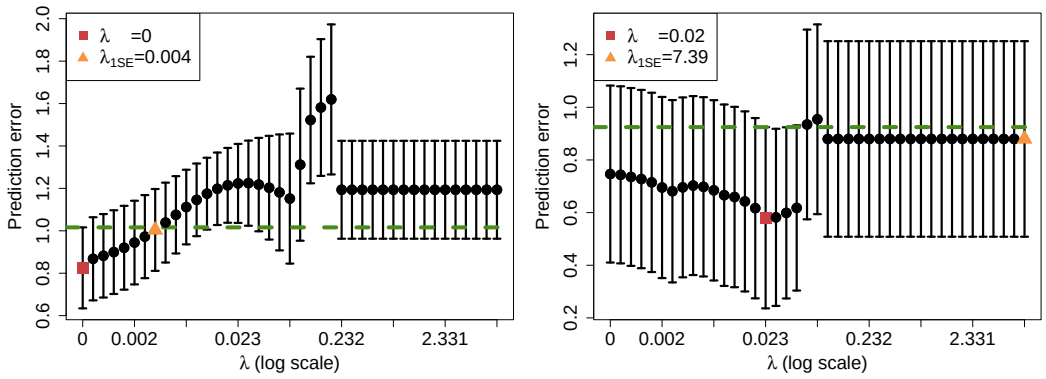
(f) λ vs $\hat{\theta}$

Figure 1: (a) Various profile log likelihoods for the sine example. All curves are plotted over $\theta \in [0.001, 100]$ on the log scale. (b) GP using the MLE from the flat $\log \mathcal{L}(\theta)$ with nugget added for stability. (c) GP using the MP inverse and the resulting MLE. (d) GP with the $\hat{\theta}$ from (c), but with nugget instead of MP. (e) GP using $\hat{\theta} = 24.207$, the maximizer of the penalized likelihood $Q(\theta)$ shown in panel (a). (f) The penalized $\hat{\theta}$ estimates that result from $\lambda \in [0, 7.39]$.

shows the pMLE's that result from $\lambda \in [0, 7.39]$. A penalty that is too large will open the door to the numerical issues discussed previously, even with the addition of a nugget (e.g., Figure 1d).

The example from Figure 1 demonstrates the potential of improving a GP surrogate via penalization. We now turn to tuning parameter selection. An ideal selection strategy will identify a λ value that produces as good or better estimates than its unpenalized counterpart. Li and Sudjianto (2005) recommended leave-one-out CV (LOOCV) or K-fold CV where the validation point or fold is evaluated in terms of prediction error (see Section 2). We employed LOOCV with prediction error on the sine example; Figure 2a shows the mean and standard error across the "leave-one-out" folds as a function of $\lambda \in [0, 7.39]$. The red square marks the degree of penalization that resulted in the lowest average prediction error, $\lambda = 0$ (i.e., no penalization). We have already seen that a degree of penalization is beneficial in this setting, so the selection of $\lambda = 0$ is disappointing. Another common approach for selecting λ is the "one-standard-error" (1SE) rule (Breiman et al., 1984; Chen and Yang, 2021). This involves selecting the smallest λ value whose average performance metric is within one standard error of the actual lowest performance metric. To demonstrate, the green dashed line in Figure 2a marks one standard error above the lowest average prediction error. The yellow triangle marks $\lambda = 0.004$, the value whose average prediction error is below that threshold, corresponding to a modest degree of penalization.

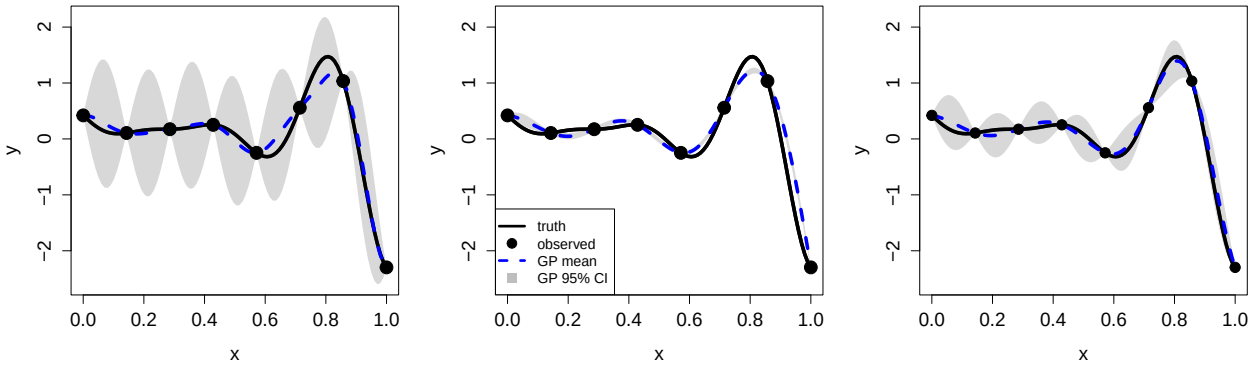
Consider now the Forrester function (Forrester et al., 2008), defined as $y = (6x - 2)^2 \sin(12x - 4)$ with 8 equally spaced data points over the input range $x \in [0, 1.25]$, again scaled to $[0, 1]$ for analysis. Although not



(a) Sine function (b) Forrester function

Figure 2: LOOCV prediction error for the Sine function (a) and the Forrester function (b) across $\lambda \in [0, 7.39]$. Black vertical lines indicate plus/minus one standard error. Red squares mark the λ values that minimize average prediction error ($\lambda = 0$ and $\lambda = 0.02$, respectively). Orange triangles mark the largest λ within one standard error of the "best" λ ($\lambda_{1\text{SE}} = 0.004$ and $\lambda_{1\text{SE}} = 7.39$, respectively).

shown, the profile log likelihood flattens for large values of θ. Figure 3a shows the GP surrogate resulting from the MLE $\hat{\theta} = 100$. There is room for improvement as the GP has inflated variance over the entire space and inaccurate mean predictions for $x \in [0.6, 0.9]$. We again performed LOOCV with the LASSO penalty which selected $\lambda = 0.02$ (see Figure 2b). The resulting GP, shown in Figure 3b severely underestimates uncertainty. There is significant variability across the leave-one-out folds, as shown by the standard error bars in Figure 2b. Leveraging the one-standard-error rule here is inadvisable, as it would recommend even more aggressive penalization ($\lambda_{1\text{SE}} = 7.39$). Despite the failures of traditional tuning parameter selection methods in this setting, there is still potential for a degree of penalization to improve performance. As evidence, we present the GP surrogate with $\lambda = 0.004$, which results in the pMLE $\hat{\theta} = 33.919$, in Figure 3c. This GP offers more accurate predictions and more effective UQ than those previously shown. The punchline: penalization can improve upon MLE, but only if λ is chosen effectively. The erratic behavior of the prediction error metrics in Figure 2 is also of interest. The abrupt jumps in performance raise concerns regarding the consistency and reliability of this tuning parameter selection strategy – another source of motivation for our proposed DPE metric.



(a) $\hat{\theta} = 100$ (b) $\hat{\theta} = 10.565$ (c) $\hat{\theta} = 33.919$

Figure 3: GP surrogates for the Forrester (Forrester et al., 2008) function with various lengthscales. Panel (a) shows the MLE, (b) shows the pMLE with LOOCV using prediction error, and (c) shows a happy medium.

1.2 Contributions and Overview

These motivating examples demonstrate the potential for penalized estimation to improve GP surrogate performance, but they also highlight the challenges and potential pitfalls. Small lengthscale values can "break" the surrogate, causing GP means to revert to the prior and GP variances to inflate, even with the addition of a nugget (e.g., Figure 1d). Proper tuning parameter selection is essential, as too much penalization can be detrimental (e.g., Figure 3b).

The main contribution of this paper is a new tuning parameter selection metric for penalized likelihood estimation with GP surrogates, called decorrelated prediction error (DPE). DPE may be interpreted as an external estimator of $\hat{\sigma}^2_\theta$ based on the validation set, using the $\hat{\theta}$ from the training set. We show that DPE is more consistent and reliable than traditional metrics, particularly for K-fold CV involving small datasets, which is common for expensive computer experiments. We compare DPE to prediction error, Mahalanobis distance, and Score, finding that prediction error tends to overpenalize in these types of datasets, while Mahalanobis distance and Score can be highly variable. Unlike the other three metrics, DPE is highly sensitive to non-interpolating solutions, making it preferable for deterministic functions.

Another contribution of this paper is a simulation study across multiple test functions to examine the performance of penalized estimation against traditional MLE for GP surrogates of deterministic black-box functions. In particular, the conditions that warrant penalization for GP surrogates are not well known. This is in stark contrast to penalized least squares, which has been thoroughly explored (Bunea et al., 2011; James et al., 2013). Our simulation study provides partial answers to this question, although this is still an open research problem.

The remainder of this article is organized as follows. In Section 2 we review GPs and penalized likelihood estimation. Section 3 motivates and describes the DPE metric for K-fold CV, including the incorporation of a one-standard-error rule. Section 4 benchmarks DPE against competing tuning parameter selection strategies on a variety of test functions. Section 5 revisits the Piston slap noise data example from Li and Sudjianto (2005), comparing the performance of their LOOCV with prediction error to our K-fold CV with DPE. We conclude the paper with a discussion and opportunities for future work in Section 6. All methods, including our proposed DPE metric, are supported by the new `GPpenalty` package on CRAN (Mutoh, 2025).

2 Review of Penalized MLE

For higher dimensional functions, we upgrade our kernel function to incorporate separable lengthscales $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_d)^\top$, where θ_p determines how quickly the correlation between function values decays in dimension $p \in \{1, \ldots, d\}$. Specifically, we use

$$R(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(- \sum_{p=1}^{d} \theta_p (x_{ip} - x_{jp})^2 \right) + g\mathbb{I}_{(i=j)} \ .$$

As we saw in Section 1.1, smaller $\boldsymbol{\theta}$ values result in smoother and flatter predictions, but their effect on UQ may be inconsistent (sometimes shrinking and other times overinflating variances).

Penalized MLE for GPs proceeds through numerical optimization of $Q(\boldsymbol{\theta} \mid y_n, \sigma^2 = \hat{\sigma}^2_\theta)$ from Eq. (3). Throughout, we use the LASSO penalty, $p_\lambda(\boldsymbol{\theta}) = \lambda \sum_{p=1}^{d} |\theta_p|$, although our software is not restricted to this choice (more on this in Section 4). Selection of the tuning parameter λ is key (Fan and Tang, 2013; Arlot and Celisse, 2010). CV is a popular selection strategy that first partitions the data into two sets: training data $\mathcal{D}^t$ and validation data $\mathcal{D}^v$. Then penalized estimation of $\boldsymbol{\theta}$ is performed using $\mathcal{D}^t$ across a range of λ values, each producing a prediction mean, $\boldsymbol{\mu}_\lambda(\mathbf{X}^v \mid \mathcal{D}^t)$, and prediction covariance, $\boldsymbol{\Sigma}_\lambda(\mathbf{X}^v \mid \mathcal{D}^t)$, at the validation set's input locations, $\mathbf{X}^v$. Replacement of the subscript "θ" with "λ" is meant to emphasize how the tuning parameter is the driver of the resulting pMLE values (we find "θ_λ" too cumbersome). The quality of these penalized estimators is evaluated on $\mathcal{D}^v$ using some metric, say $C(\mathcal{D}^v \,|\, \mathcal{D}^t, \lambda)$, involving $\boldsymbol{\mu}_\lambda(\mathbf{X}^v \mid \mathcal{D}^t)$ and/or $\boldsymbol{\Sigma}_\lambda(\mathbf{X}^v \mid \mathcal{D}^t)$. The λ that minimizes the metric, denoted λ^*, is selected and used to estimate $\boldsymbol{\theta}$ with the full data, $\mathcal{D}$, to produce the final surrogate.

A related strategy is K-fold CV, in which $\mathcal{D}$ is partitioned into K sets, or folds, denoted by $\mathcal{D}_1, \ldots, \mathcal{D}_K$, each of size $n_v = n/K$. LOOCV corresponds to $K = n$, making each $\mathcal{D}_k$ a single data point. The above CV procedure is performed for $k = 1, \ldots, K$, setting $\mathcal{D}^v = \mathcal{D}_k$ and $\mathcal{D}^t = \mathcal{D} \setminus \mathcal{D}_k \equiv \mathcal{D}_{-k}$, producing K

metrics $C(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda)$. These metrics are typically averaged at each λ value to produce the function $C(\lambda) = \frac{1}{K}\sum_{k=1}^{K} C(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda)$. The optimal λ^* is the one that minimizes $C(\lambda)$.

It may be desired to introduce even more penalization than that under λ^*. A popular approach is the one-standard-error rule that chooses the largest λ whose $C(\lambda)$ is within one standard error of $C(\lambda^*)$. Applying the 1SE rule is fairly straightforward: identify λ^*, calculate $\text{SD}(\lambda^*)$, the estimated standard deviation of the $C(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda^*)$, and then calculate the estimated standard error, $\text{SE}(\lambda^*) = \text{SD}(\lambda^*)/\sqrt{K}$. The new λ value is then

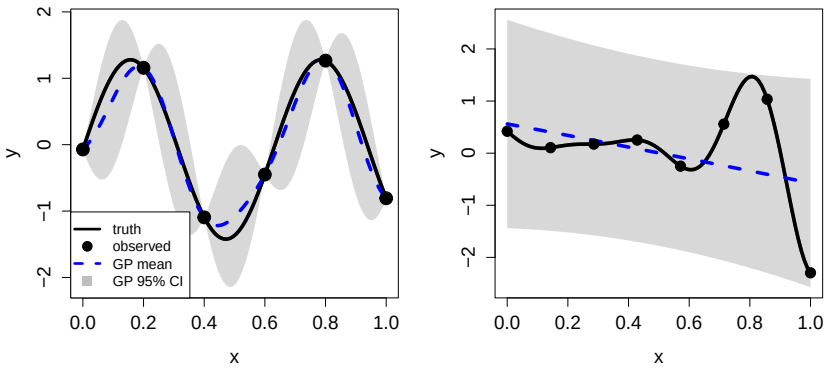
$$\lambda_{\text{1SE}} = \max\left\{\lambda \mid C(\lambda) \leq C(\lambda^*) + \text{SE}(\lambda^*)\right\}.$$

Clearly $\lambda_{\text{1SE}} \geq \lambda^*$ so the corresponding penalized estimator will experience at least as much, if not more, shrinkage compared to that under λ^*. However, as shown in Figure 2b, sometimes $\text{SE}(\lambda^*)$ can be large, which will induce excessive penalization.

The choice of the CV metric is nontrivial, especially on small data sets. In fact, an additional contribution of this paper is a discussion of why popular CV metrics can have erratic behavior for penalized MLEs of GPs. The chosen CV metric should target the statistical properties of interest and become inflated if said properties are not met. An intuitive metric is prediction error (PE):

$$\text{PE}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) = (\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k})^\top (\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k}) \ ,$$

where we simplify notation with $\boldsymbol{\mu}_{\lambda,k} \equiv \boldsymbol{\mu}_\lambda(\mathbf{X}_k \mid \mathcal{D}_{-k})$. This metric was recommended by Li and Sudjianto (2005), Yi et al. (2011), and Zhang et al. (2020), but the motivating examples in Section 1.1 demonstrated its potential issues. Furthermore, the effect of the 1SE rule when using the PE metric is unreliable. To demonstrate, Figure 4 shows the GP surrogates resulting from λ_{1SE} (the orange triangles in Figure 2) for the two examples introduced in Section 1.1. In the Sine example, the 1SE rule offers similar performance to the MLE. In the Forrester example, the 1SE rule severely overpenalizes, leading to numerical issues and providing a useless surrogate.



(a) $\hat{\theta} = 43.330$ (b) $\hat{\theta} = 0.001$

Figure 4: Predictive performance of the penalized GP using λ_{1SE} for the Sine and Forrester functions.

PE is limited in its assessment of the predictive surrogate because it ignores UQ. It can encourage $\hat{\boldsymbol{\theta}}$ that produce narrower prediction intervals which might fail to cover the truth. One CV metric that incorporates UQ is Mahalanobis distance (MD; Mahalanobis, 1936), a popular metric for Normally distributed data that accounts for uncertainty:

$$\text{MD}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) = (\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k})^\top \boldsymbol{\Sigma}_{\lambda,k}^{-1} (\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k})$$

where $\boldsymbol{\Sigma}_{\lambda,k} \equiv \boldsymbol{\Sigma}_\lambda(\mathbf{X}_k \mid \mathcal{D}_{-k}) = \hat{\sigma}^2_{\lambda,-k}\mathbf{R}_\lambda(\mathbf{X}_k \mid \mathcal{D}_{-k})$ with $\hat{\sigma}^2_{\lambda,-k} = \left(\mathbf{y}_{-k}^\top \mathbf{R}_{\lambda,-k}^{-1} \mathbf{y}_{-k}\right)/(n - n_v)$. If the estimates for $\boldsymbol{\theta}$ and σ^2 equal their true values, this quantity follows a χ^2 distribution with n_v degrees of freedom.

Another metric that incorporates UQ is Score (Gneiting and Raftery, 2007):

$$\text{Score}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) = \text{MD}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) + \log|\boldsymbol{\Sigma}_{\lambda,k}|$$

which is proportional to the negative log likelihood of the penalized estimates evaluated on $\mathcal{D}_k$. While closely related to Mahalanobis distance, adding $\log|\boldsymbol{\Sigma}_{\lambda,k}|$ can sometimes help to avoid surrogates with large uncertainty (Gramacy, 2020). Choosing $n_v \geq 2$ is recommended for MD and Score to incorporate both the predicted variances and covariances in the assessment of the surrogate. In practice, smaller values of these metrics are considered to be indicative of a better fitting surrogate.

3 Decorrelated Prediction Error

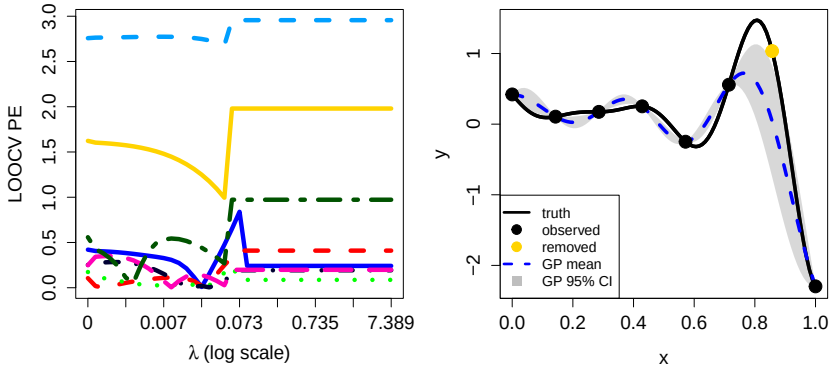
Section 1.1 presented two examples where the profile log likelihood is both flat and maximized for large values of θ. Large θ values can be detrimental to surrogate performance as they shrink the correlation among responses. Specifically, for large enough θ, $\mathbf{R}(\mathcal{X}, \mathbf{X}_n) = \mathbf{0}$ and $\mathbf{R}_n^{-1} \approx \mathbf{I}$, leading to $\boldsymbol{\mu}_\theta(\mathcal{X} \mid \mathcal{D}) = \mathbf{0}$ and $\boldsymbol{\Sigma}_\theta(\mathcal{X} \mid \mathcal{D}) \approx \hat{\sigma}^2_\theta \mathbf{I}$, where $\hat{\sigma}^2_\theta = \mathbf{y}_n^\top \mathbf{y}_n$. The exception is when a row of $\mathbf{X}_n$ is included in $\mathcal{X}$, for which $\mu_\theta(\mathbf{x}_i \mid \mathcal{D}) \approx y_i$ and $\boldsymbol{\Sigma}_\theta(\mathcal{X} \mid \mathcal{D}) \approx 0$. Details of these derivations and other results in this section may be found in Section A.

Penalization encourages smaller values of θ, but if it goes too far it can be detrimental too. Increasing λ shrinks θ, producing smoother $\boldsymbol{\mu}_\theta(\mathcal{X} \mid \mathcal{D})$. As $\lambda \to \infty$, $\theta \to 0$, resulting in $\mathbf{R}(\mathcal{X}, \mathbf{X}_n) \approx \mathbf{J}$, a matrix of all ones, and $\mathbf{R}_n^{-1} \approx (g\mathbf{I} + \mathbf{J})^{-1} \approx g^{-1}\left(\mathbf{I} - \frac{1}{n}\mathbf{J}\right)$. As $\mathbf{J}\left(\mathbf{I} - \frac{1}{n}\mathbf{J}\right) = \mathbf{0}$, the resulting surrogate has $\boldsymbol{\mu}_\theta(\mathcal{X} \mid \mathcal{D}) = \mathbf{0}$, just like when $\theta \to \infty$, except now this holds whether or not the training data is included in $\mathcal{X}$. The impact of increasing λ on $\boldsymbol{\Sigma}_\theta(\mathcal{X} \mid \mathcal{D})$ is not as straightforward. As λ initially moves away from 0, prediction variances will start to decrease, but as $\lambda \to \infty$, $\boldsymbol{\Sigma}_\theta(\mathcal{X} \mid \mathcal{D}) \approx \hat{\sigma}^2_\theta \mathbf{I}$, where $\hat{\sigma}^2_\theta \approx g^{-1}\mathbf{y}_n^\top(\mathbf{I} - \frac{1}{n}\mathbf{J})\mathbf{y}_n = g^{-1}\mathbf{y}_n^\top \mathbf{y}_n$ becomes inflated due to our choice of g. These derivations explain the behavior shown in Figure 1d and Figure 4b. We will refer to surrogates with this inflated variance and $\boldsymbol{\mu}_\theta(\mathcal{X} \mid \mathcal{D}) \approx \mathbf{0}$ as "nugget-dominated surrogates."

Nugget-dominated surrogates and surrogates with excessively large θ are practically useless. Ideally, $C(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda)$ will be inflated under both surrogates and will decrease for λ values whose penalized estimates actually improve the surrogate. Unfortunately, we have found this ideal behavior does not necessarily hold for the three metrics given in Section 2, particularly for small data sets where penalization has the most potential for improvement. Without a consistent and reliable tuning parameter selection method, this potential may not be realized in practice.

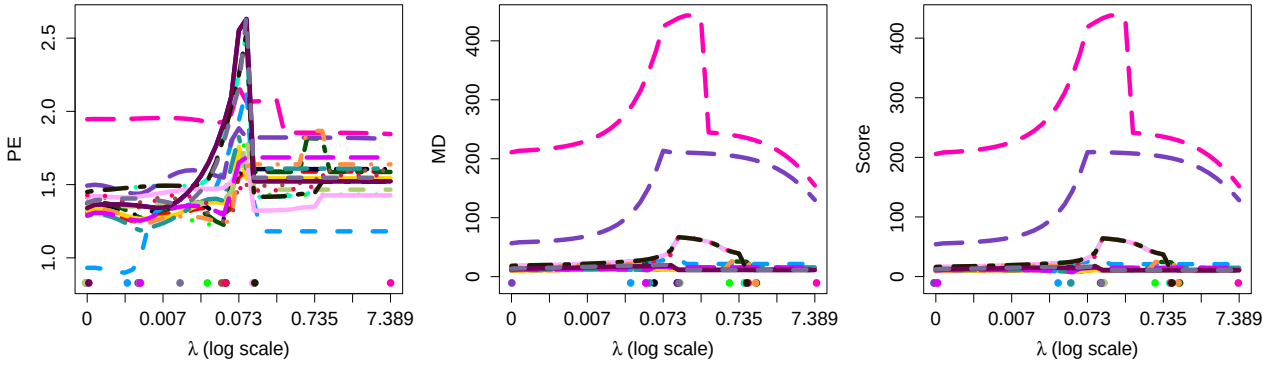
To demonstrate these issues, we revisit the Forrester example from Section 1.1. First, we performed LOOCV with PE, as recommended by Li and Sudjianto (2005). Figure 5a shows the individual PE curves across λ – one line for each leave-one-out fold (the average, PE(λ), and standard errors across these lines were shown in Figure 2b). There is large between-fold variability across λ. The curve with the largest values corresponds to $\mathcal{D}_8$, where the outlier data point ($x_8 = 1$) has been left out. However, because the curve is fairly constant, it has little influence on PE(λ). The second largest curve has the greatest influence on PE(λ), exhibiting the most change across λ. It corresponds to $\mathcal{D}_7$, where the observation at $x_7 = 0.857$ has been left out. This observation, highlighted by the yellow point in Figure 5b, has the largest observed y value. The surrogate based on $\mathcal{D}_{-7}$ under $\lambda = 0.046$, which minimizes PE($\mathcal{D}_7 \mid \mathcal{D}_{-7}, \lambda$), is shown in Figure 5b. This "best" surrogate was still unable to capture the observed $y_7 = 1.034$ and other values of the true function in a neighborhood of this observation. This serves as a reminder that there are limitations to $\mathcal{D}_{-k}$'s ability to predict at $\mathcal{D}_k$. This in turn could produce one or more folds with observations that lead to inflated $C(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda)$ values due to extrapolation. Such folds can dominate the behavior of $C(\lambda)$, and hence the choice of λ^*.

Next, we repeatedly performed 4-fold CV for the Forrester example twenty times, each with a random partitioning of the data. Each line in Figure 6a - 6c is an averaged CV metric for one of these repetitions; we considered PE, MD, and Score. The points just above the x-axis are the corresponding λ^* values. Although not shown, we also observed large between-fold variability for each of the 4-fold CVs, similar to Figure 5a. All three panels show large variability among the twenty $C(\lambda)$ curves and inconsistent λ^* values across the entire range of possibilities, sometimes producing nugget-dominated surrogates. Across the twenty 4-fold CVs, PE(λ) sometimes chose $0 < \lambda^* \leq 0.007$, which produced surrogates comparable to Figure 3c. However,



(a) LOOCV PE (b) LOOCV Prediction at $\mathcal{D}_7$

Figure 5: (a) LOOCV PE curves across λ values for the Forrester function. Each colored line represents the PE curve obtained by leaving out a different data point. (b) Surrogate predictive performance conditioned on $\mathcal{D}_{-7}$ under $\lambda = 0.046$. The yellow circle marks $\mathcal{D}_7 = (0.857, 1.034)$, an influential point with the highest observed y value.



(a) PE (b) MD (c) Score

Figure 6: Behavior of 4-fold CV metrics across λ values for the Forrester function. Each line represents one of 20 repetitions with re-randomized folds. Colored circles along the x-axis indicate the selected λ value for each repetition.

more often it chose λ^* that produced a surrogate that was either overly smoothed, as in Figure 3b, or nugget-dominated, as in Figure 4b.

Since PE uses only $\mu_\lambda(\mathbf{X}_k \mid \mathcal{D}_{-k})$, we conjectured that MD and Score, which incorporate the surrogate's UQ, could perform better. Figure 6b and 6c show the twenty 4-fold CV curves for MD and Score, respectively. They are nearly indistinguishable. While the distribution of λ^* values for PE had a noticeable gap between 0.073 and 7.389, the distribution of λ^* values for MD and Score flips the script, exhibiting a noticeable gap between 0 and approximately 0.073. The large λ^* values chosen result in nugget-dominated surrogates resembling Figure 4b. The tendency for MD(λ) and Score(λ) to be minimized by such large λ^* for the Forrester data can be explained by looking at the limits of MD($\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda$) and Score($\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda$) as $\lambda \to \infty$.

Let $\text{SS}_k = \mathbf{y}_k^\top (\mathbf{I} - \frac{1}{n_v}\mathbf{J})\mathbf{y}_k$ denote the corrected sum-of-squares for $\mathbf{y}_k$; similarly define SS_{-k}. For brevity,

let $\mathbf{R}_{\lambda,k} \equiv \mathbf{R}_\lambda(\mathbf{X}_k \mid \mathcal{D}_{-k})$. Using previous limiting arguments, we find

$$\lim_{\lambda\to\infty} \mathrm{MD}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) = \lim_{\lambda\to\infty} \frac{(\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k})^\top \mathbf{R}_{\lambda,k}^{-1}(\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k})}{\hat{\sigma}^2_{\lambda,-k}}$$
$$\approx (n - n_v)\frac{\mathrm{SS}_k}{\mathrm{SS}_{-k}} \ ,$$

which no longer involves the nugget term g. For the Forrester data with $n_v = 2$, there are multiple situations where SS_k is significantly less than SS_{-k}, hence encouraging $\mathrm{MD}(\lambda)$ to be minimized by large λ values. The limit of Score also shares similar issues:

$$\lim_{\lambda\to\infty} \mathrm{Score}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) = \lim_{\lambda\to\infty} \mathrm{MD}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) + n_v \log(\hat{\sigma}^2_{\lambda,-k}) + \log|\mathbf{R}_{\lambda,k}|$$
$$\approx (n - n_v)\frac{\mathrm{SS}_k}{\mathrm{SS}_{-k}} + n_v \log\left(\frac{\mathrm{SS}_{-k}}{n - n_v}\right) + \log\left(\frac{n}{n - n_v}\right) \ .$$

Again, g has no effect on Score when $\lambda \to \infty$, and the additional terms provide no assurances that $\mathrm{Score}(\lambda)$ will prevent selecting λ^* that produces a nugget-dominated model.

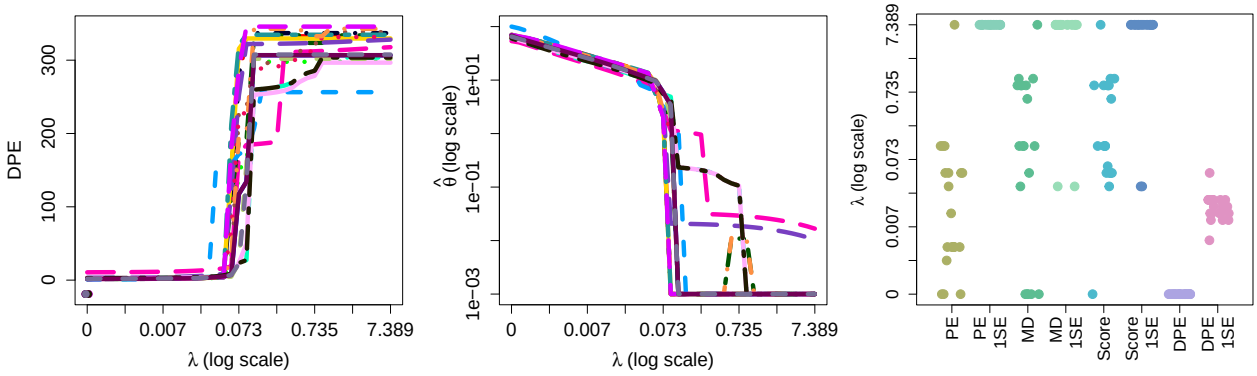
When λ becomes large enough to drive $\hat{\theta} \to 0$, the variance estimate $\hat{\sigma}^2_\lambda$ inflates dramatically due to the small fixed nugget g. This inflation is a useful indicator for when we are approaching a nugget-dominated surrogate. Both MD and Score, while initially appealing for their incorporation of UQ, can fail to capitalize on this information and, as a result, are prone to selecting an impractical, nugget-dominated surrogate.

To take advantage of the inflation property of $\hat{\sigma}^2_\lambda$, we propose the decorrelated prediction error metric that incorporates both prediction accuracy and the correlation structure:

$$\mathrm{DPE}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) = (\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k})^\top \mathbf{R}_{\lambda,k}^{-1}(\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k}) \ .$$

DPE gets its name from its relationship to PE, which looks at correlated responses $\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k}$. Assuming λ produces an estimate of $\boldsymbol{\theta}$ that is close to its true value, the responses are decorrelated by pre-multiplying by $\mathbf{R}_{\lambda,k}^{-1/2}$, a square-root matrix of $\mathbf{R}_{\lambda,k}^{-1}$. DPE is also closely related to MD with the scaling factor $1/\hat{\sigma}^2_{\lambda,-k}$ removed. If $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}$, $\mathrm{DPE}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda)$ follows a scaled χ^2-squared distribution, $\sigma^2\chi^2_{n_v}$, with n_v degrees of freedom. Otherwise, we expect $\mathrm{DPE}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda)$ to become inflated. Therefore, smaller $\mathrm{DPE}(\lambda)$ values indicate a better surrogate. Finally, DPE has the desired inflation property to avoid nugget-dominated surrogates:

$$\lim_{\lambda\to\infty} \mathrm{DPE}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) = \frac{1}{g}\mathrm{SS}_k \ . \tag{4}$$



(a) DPE (b) λ vs $\log\hat{\theta}$ (c) Selected λ

Figure 7: (a) DPE metric for the Forrester example across 20 4-fold CV repetitions. (b) Resulting $\hat{\theta}$, averaged across folds. (c) $\lambda^\star$ and $\lambda_{1\mathrm{SE}}$ for each of the four metrics.

Returning to the Forrester example, Figure 7a shows the DPE metric across the same 20 repeated 4-fold CVs from Figure 6. Compared to PE, MD, and Score, the DPE curves have consistently higher values for larger λ and consistently lower values when $\lambda < 0.073$. This indicates that the DPE metric may be less sensitive to fold-to-fold variability, reducing concerns about instability in tuning parameter selection. Figure 7b shows the resulting $\hat{\theta}$ values, estimated for each fold and averaged across folds. When $\lambda > 0.073$, the average $\hat{\theta}$'s drop sharply toward zero, and the DPE curves in Figure 7a become inflated, matching the expectations set by Eq. (4). DPE chose $\lambda^* = 0$ for all of the 4-fold CVs. Moreover, the reduced fold-to-fold variability and the rapid inflation of DPE for large λ suggests potential value in applying the 1SE rule. The λ^* and $\lambda_{1\text{SE}}$ values selected by all four metrics are shown in Figure 7c. Due to the high between-fold variability of PE, MD, and Score, applying the 1SE rule almost always resulted in $\lambda_{1\text{SE}} = 7.389$. The $\lambda_{1\text{SE}}$ values under DPE show remarkable consistency while avoiding overly large λ. In the following simulation studies, we will show that this behavior is not unique to the Forrester function.

4 Numerical Studies

In this section, we examine the performance of tuning parameter selection with K-fold CV under PE, MD, Score, and DPE on a variety of synthetic benchmarks. Given our observations in Section 3, the 1SE method was only applied to the DPE metric.

Implementation details. As the lengthscale parameter $\boldsymbol{\theta}$ lacks a closed-form solution, numerical optimization of Eq. (3) is required. We use the `optim` function in R with the L-BFGS-B algorithm (Byrd et al., 1995), which supports gradient-based updates and bound constraints. We restrict the search space of $\boldsymbol{\theta}$ to the interval $[0.001, 1000]^d$. Since gradient-based optimization methods are sensitive to initial values, we use 10 random multi-starts to reduce the risk of convergence to local optima (Martí, 2003). Our software (`GPpenalty`; Mutoh, 2025) offers both the LASSO penalty introduced in Section 2 and the smoothly clipped absolute deviation (SCAD) penalty (Fan and Li, 2001). After examining both LASSO and SCAD, we found no significant difference in performance and therefore adopted the simpler LASSO penalty for the following exercises.

Test functions. We consider four different test functions that are commonly found in the GP literature: Lim Nonpolynomial (Lim et al., 2002), Franke (Franke, 1979), Piston Simulation (Ben-Ari and Steinberg, 2007), and Borehole (Worley, 1987). Training and testing data were generated with Latin hypercube sampling using the `lhs` R package (Carnell, 2022). We varied data sizes based on input dimension (see Table 1), aiming to replicate real-world small data scenarios. We performed 5-fold CV for all data sets. The final surrogate for each method was estimated using the full dataset $\mathcal{D}$ with either λ^* or $\lambda_{1\text{SE}}$, then evaluated on the test data set. We repeated this process 100 times with new Latin hypercube samples.

Function	d	n	n_{test}
Lim Nonpolynomial	2	10	200
Franke	2	10	200
Piston Simulation	7	15	700
Borehole	8	15	800

Table 1: Simulation settings for each test function.

Validation metrics. To assess the surrogate's prediction accuracy and UQ, we evaluated root mean squared error (RMSE) and continuous ranked probability score (CRPS; Gneiting and Raftery, 2007) on the test data. RMSE is defined as

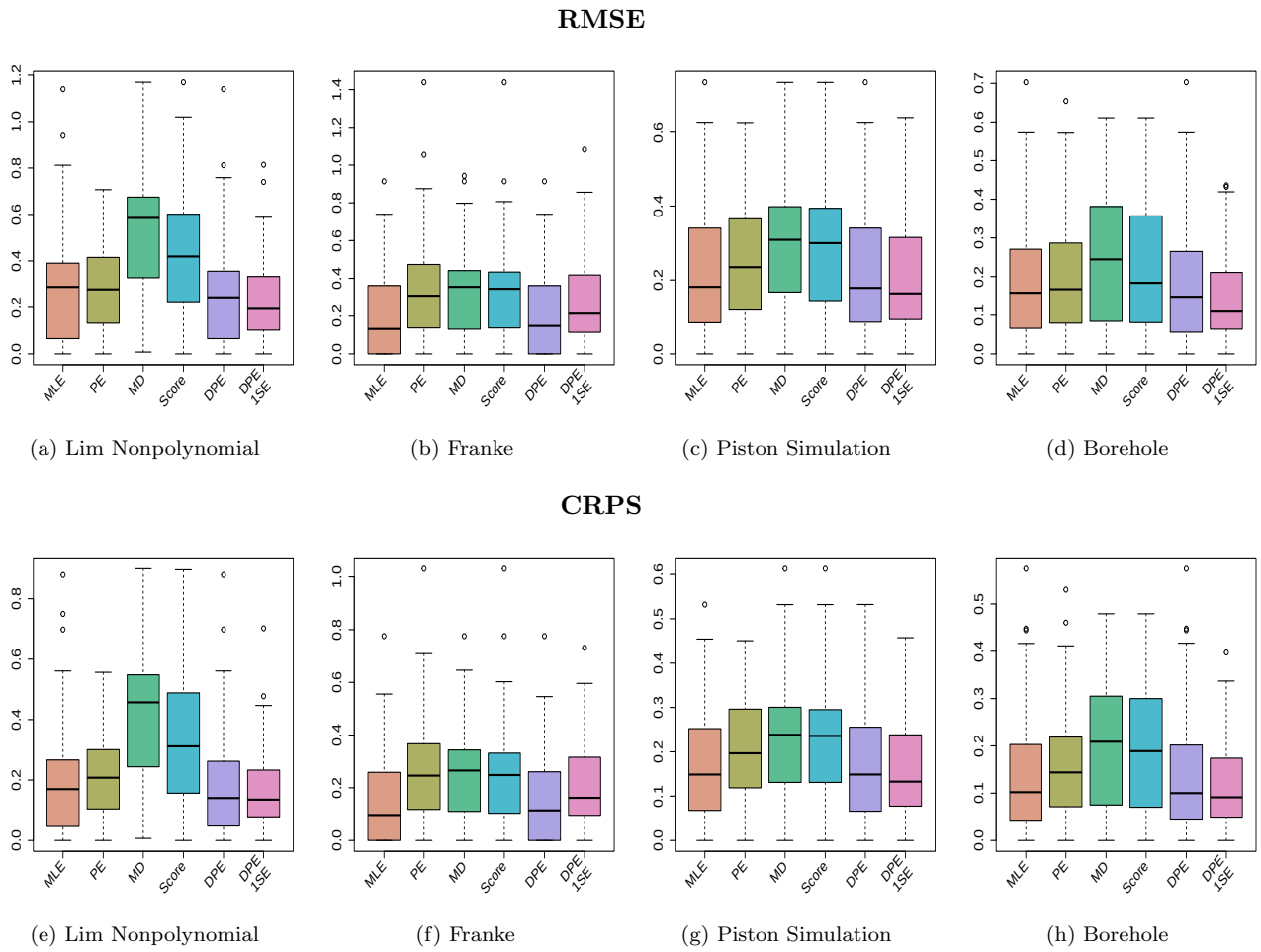
$$\text{RMSE} = \sqrt{\frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} (y_{\text{test},i} - \mu_{\lambda^*,i})^2} \ ,$$

where $\mu_{\lambda^*,i} \equiv \mu_{\lambda^*}(\mathbf{x}_{\text{test},i} \mid \mathcal{D})$. When the predictive distribution is Gaussian with mean $\mu_{\lambda^*,i}$ and variance $\tau^2_{\lambda^*,i} = \mathbf{\Sigma}_{\lambda^*}(\mathbf{x}_{\text{test},i} \mid \mathcal{D})$, CRPS is defined as

$$\text{CRPS} = -\frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \tau_{\lambda^*,i} \left(\frac{1}{\sqrt{\pi}} - 2\phi(z_{\lambda^*,i}) - z_{\lambda^*,i}(2\Phi(z_{\lambda^*,i}) - 1) \right) \quad \text{for} \quad z_{\lambda^*,i} = \frac{y_{\text{test},i} - \mu_{\lambda^*,i}}{\tau_{\lambda^*,i}} ,$$

where ϕ and Φ represent the $\mathcal{N}(0,1)$ probability density function and cumulative distribution function, respectively. Here, we have negated the standard form from Gneiting and Raftery (2007), so a lower CRPS indicates a better distributional prediction. To demonstrate the potential for penalized MLE to improve the surrogate, we included two additional competitors: the GP with unpenalized MLE and the GP with optimal penalization, denoted pMLE*, which uses the λ that minimized RMSE on the test data.

Results. Figure 8 shows the RMSE and CRPS for MLE and the different tuning parameter selection strategies relative to those under the optimal pMLE*, which always had the best performance. Across the board, the performances of DPE and MLE were closely aligned. For all functions other than Franke, DPE 1SE provided modest improvements over MLE. The additional penalization chosen by DPE 1SE appears undesirable for the Franke function, slightly elevating both RMSE and CRPS. Notably, both DPE and DPE 1SE outperformed PE, MD, and Score metrics for all functions, especially in terms of CRPS. This performance improvement is due to DPE's tendency to select smaller values of λ. Although none of the metrics were able to match the performance that pMLE* could achieve, the DPE and DPE 1SE metrics proved to be the most reliable.



(a) Lim Nonpolynomial (b) Franke (c) Piston Simulation (d) Borehole

(e) Lim Nonpolynomial (f) Franke (g) Piston Simulation (h) Borehole

Figure 8: RMSE (panels (a)-(d), upper) and CRPS (panels (e)-(h), lower) evaluated across 100 repetitions. The y-axis shows the relative difference from pMLE* on a square root scale.

5 Piston Slap Noise Data

In this section, we analyze the piston slap noise data introduced by Hoffman et al. (2003) and considered in Li and Sudjianto (2005). The response variable is piston slap noise, and the 6 input settings are: cylinder liner (x_1), location of peak pressure (x_2), skirt length (x_3), skirt profile (x_4), skirt ovality (x_5), and pin offset (x_6). Each run of the computer experiment required 24 hours of computation. We copy the training/testing split from Li and Sudjianto (2005) with $n = 12$ and $n_{\text{test}} = 100$ (training data details are provided in Section C). We first evaluated the performance of MLE, resulting in an RMSE of 0.767 and a CRPS of 0.500, and LOOCV with PE, resulting in an RMSE of 0.768 and a CRPS of 0.504 (a slightly inferior result).

We then implemented 4-fold CV with PE, DPE, and DPE 1SE. Since the performance of 4-fold CV can depend on the construction of the folds, we repeated the procedure for 100 randomly generated folds. We evaluated performance by comparing RMSE/CRPS against the "benchmark" MLE values. Table 2 shows the breakdown of the 100 randomized repetitions based on whether their RMSE/CRPS was less than, equal to, or greater than that of the MLE (the frequencies were the same for RMSE and CRPS). It also reports the median RMSE and CRPS value for each scenario. Of the CV metrics considered, DPE matched MLE most often (65/100 times). Notably, even in the times when each metric performed worse than the MLE, the median RMSE/CRPS values were only slightly elevated. Both PE and DPE 1SE significantly improved over MLE in 8 out of the 100 repetitions, with significantly lower median RMSE/CRPS values.

Freq. Median RMSE Median CRPS	PE	DPE	DPE 1SE
Less than MLE	8 0.431 0.263	2 0.431 0.263	8 0.431 0.263
Equal to MLE	31 0.767 0.500	65 0.767 0.500	4 0.767 0.500
Greater than MLE	61 0.769 0.505	33 0.768 0.503	88 0.770 0.507

Table 2: Summary of each 4-fold CV metric's performance compared to MLE. The rows correspond to scenarios where the chosen λ's RMSE/CRPS was less than, equal to, or greater than that under the MLE. Each cell reports the frequency of each occurrence, the median RMSE value, and the median CRPS value.

	MLE	LOOCV PE	Best 4-fold CV
λ	0	0.006	0.058
$\hat{\sigma}^2_\theta$	1.151	1.241	5.382
$\hat{\theta}_1$	4.067	3.728	0.387
$\hat{\theta}_2$	0.001	0.001	0.001
$\hat{\theta}_3$	0.588	0.532	0.001
$\hat{\theta}_4$	0.001	0.001	0.906
$\hat{\theta}_5$	0.001	0.001	0.019
$\hat{\theta}_6$	2.751	2.550	0.428

Table 3: Parameter estimates from the piston slap noise data for MLE, LOOCV with PE, and a 4-fold CV providing improved RMSE/CRPS (top row of Table 2).

To provide more context for this improved performance, Table 3 shows $\hat{\sigma}^2_\theta$ and $\hat{\theta}_p$ for the MLE and LOOCV with PE surrogates alongside the estimates for $\lambda = 0.058$, which produced the median RMSE value of 0.431 for the 4-fold CV metrics. Even slight penalization with this λ led to distincly different estimates.

Notably, $\hat{\theta}_1$ and $\hat{\theta}_6$ were shrunk significantly, yet $\hat{\theta}_2$ was unaffected by the penalization. Perhaps the most impactful changes occurred for $\hat{\theta}_4$, which went from the smallest lengthscale under MLE and LOOCV to the largest lengthscale under the penalization. These findings illustrate that even slight penalization can reshape hyperparameter estimates and influence model behavior and performance.

6 Discussion

This article examined the penalized likelihood framework for GPs, originally proposed by Li and Sudjianto (2005), discussing common pitfalls and proposing potential remedies. We demonstrated the importance of including a nugget for numerical stability, but then showed how it may have unintended consequences on other parameter estimates, leading to "nugget-dominated surrogates." Popular metrics for evaluating GPs, including PE, MD, and Score, often struggle to consistently identify reliable tuning parameters under K-fold CV with small data sets. In particular, PE's ignorance of UQ tended to produce surrogates whose UQ failed to capture the true function. MD and Score, which incorporate UQ, were shown to have issues caused by inclusion of the nugget effect, leading to their tendency to choose nugget-dominated surrogates. We ultimately proposed a new metric, DPE, that accounts for UQ and avoids λ that produce nugget-dominated surrogates. In our numerical studies across multiple test functions, we found DPE to be conservative in the amount of recommended penalization; combining it with the 1SE rule often produced smoother predictive surfaces with improved UQ compared to MLE.

The potential for penalized estimation of GPs to improve upon MLE is greatest with small training data, such as with resource-heavy computer experiments. However, this potential can only be realized with a consistent and reliable tuning parameter selection strategy. CV remains the most widely used approach for tuning parameter selection, but it relies heavily on the choice of metric and construction of the folds. We have given analytical and numerical arguments for why DPE and DPE 1SE are preferable CV metrics, but selection of the best λ from limited observations remains a difficult problem.

Based on our observations on the Forrester function (see Figure 5) and piston slap noise data, we find that fold construction may have a greater impact on CV performance than the choice of metric, particularly for small data settings. The quality of the folds also depends on the structure of the training data. For example, the 12 training observations of the piston slap noise data (Appendix C) have poor one-dimensional projection properties. Variables 4 and 5 only have three unique values, and the remaining variables only have six unique observed values. Any partitioning of this data is likely to create CV folds with even weaker projection properties. The most important direction for future work in this area is developing more effective design construction strategies for both the training data and its CV folds.

References

Arlot, S. and Celisse, A. (2010). A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4:40 – 79.

Basak, S., Petit, S., Bect, J., and Vazquez, E. (2021). Numerical issues in maximum likelihood parameter estimation for Gaussian process interpolation. *In International Conference on Machine Learning, Optimization, and Data Science*, pages 116–131.

Bayarri, M. J., Berger, J. O., Calder, E. S., Dalbey, K., Lunagomez, S., Patra, A. K., Pitman, E. B., Spiller, E. T., and Wolpert, R. L. (2009). Using Statistical and Computer Models to Quantify Volcanic Hazards. *Technometrics*, 51(4):402–413.

Ben-Ari, E. and Steinberg, D. (2007). Modeling Data from Computer Experiments: An Empirical Comparison of Kriging with mars and Projection Pursuit Regression. *Quality Engineering*, 19:327–338.

Berthelson, P., Ghassemi, P., Wood, J., Stubblefield, G., Al-Graitti, A., Jones, M., M.F., H., Chowdhury, S., and Prabhu, R. (2021). A finite element–guided mathematical surrogate modeling approach for assessing occupant injury trends across variations in simplified vehicular impact conditions. *Medical & Biological Engineering & Computing*, 59:1065–1079.

Binois, M. and Gramacy, R. B. (2021). hetGP: Heteroskedastic Gaussian Process Modeling and Sequential Design in R. *Journal of Statistical Software*, 98(13):1–44.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. J. (1984). *Classification and Regression Trees*. Chapman and Hall/CRC, 1st edition.

Bunea, F., She, Y., Ombao, H., Gongvatana, A., Devlin, K., and Cohen, R. (2011). Penalized Least Squares Regression Methods and Applications to Neuroimaging. *NeuroImage*, 55:1519–1527.

Butler, A., Haynes, R., Humphries, T., and Ranjan, P. (2014). Efficient optimization of the likelihood function in Gaussian process modelling. *Computational Statistics & Data Analysis*, 73:40–52.

Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C. (1995). A Limited Memory Algorithm for Bound Constrained Optimization. *SIAM Journal on Scientific Computing*, 16:1190–1208.

Carnell, R. (2022). *lhs: Latin Hypercube Samples*. R package version 1.1.6.

Chen, V. C., Tsui, K.-L., Barton, R. R., and Meckesheimer, M. (2006). A review on design, modeling and applications of computer experiments. *IIE Transactions*, 38(4):273–291.

Chen, Y. and Yang, Y. (2021). The One Standard Error Rule for Model Selection: Does It Work? *Stats*, 4(4):868–892.

Dancik, G. M. (2013). *mlegp: Maximum Likelihood Estimates of Gaussian Processes*. R package version 3.1.9.

Fan, J. and Li, R. (2001). Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties. *Journal of the American Statistical Association*, 96.

Fan, Y. and Tang, C. Y. (2013). Tuning Parameter Selection in High Dimensional Penalized Likelihood. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 75(3):531–552.

Forrester, A., Sobester, A., and Keane, A. (2008). *Engineering Design Via Surrogate Modelling: A Practical Guide*. Wiley.

Franke, R. (1979). *A Critical Comparison of Some Methods for Interpolation of Scattered Data*. Defense Technical Information Center.

Fu, W. J. (1998). Penalized Regressions: The Bridge versus the Lasso. *Journal of Computational and Graphical Statistics*, 7(3):397–416.

Geisser, S. and Eddy, W. F. (1979). A Predictive Approach to Model Selection. *Journal of the American Statistical Association*, 74(365):153–160.

Gneiting, T. and Raftery, A. E. (2007). Strictly Proper Scoring Rules, Prediction, and Estimation. *Journal of the American Statistical Association*, 102:359–378.

GPy (since 2012). GPy: A gaussian process framework in python. `http://github.com/SheffieldML/GPy`.

Gramacy, R. B. (2020). *Surrogates: Gaussian Process Modeling, Design and Optimization for the Applied Sciences*. Chapman Hall/CRC, Boca Raton, Florida. `http://bobby.gramacy.com/surrogates/`.

Gramacy, R. B. and Lee, H. K. H. (2012). Cases for the nugget in modeling computer experiments. *Statistics and Computing*, 22:713–722.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 12(1):55–67.

Hoffman, R. M., Sudjianto, A., Du, X., and Stout, J. (2003). Robust Piston Design and Optimization Using Piston Secondary Motion Analysis. *SAE Technical Paper*.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An Introduction to Statistical Learning: With Applications in R*. Springer Texts in Statistics. Springer.

Kudela, J. and Matousek, R. (2022). Recent advances and applications of surrogate models for finite elementmethod computations: a review. *Soft Computing*, 26:13709–13733.

Levy, S. and Steinberg, D. (2010). Computer experiments: a review. *AStA Advances in Statistical Analysis*, 94(4):311–324.

Li, R. and Sudjianto, A. (2005). Analysis of Computer Experiments Using Penalized Likelihood in Gaussian Kriging Models. *Technometrics*, 47:111–120.

Lim, Y. B., Sacks, J., Studden, W. J., and Welch, W. J. (2002). Design and analysis of computer experiments when the output is highly correlated over the input space. *Canadian Journal of Statistics*, 30(1):109–126.

Lowe, D., Kim, M. S., and Bondesan, R. (2025). Assessing Quantum Advantage for Gaussian Process Regression.

MacDoanld, B., Ranjan, P., and Chipman, H. (2015). Gpfit: An R Package for Fitting a Gaussian Process Model to Deterministic Simulator Outputs. *Journal of Statistical Software*, 64.

MacKay, D. J. C. (1992). Bayesian interpolation. *Neural Computation*, 4(3):415–447.

Mahalanobis, P. C. (1936). On the Generalised Distance in Statistics. *Proceedings of the National Institute of Sciences of India*, 2:49–55.

Martí, R. (2003). Multi-start methods. In Glover, F. and Kochenberger, G. A., editors, *Handbook of Metaheuristics*, pages 355–368. Springer US.

Matthews, A. G. d. G., Wilk, M. v. d., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrá, P., Ghahramani, Z., and Hensman, J. (2016). Gpflow: A Gaussian process library using TensorFlow. *Journal of Machine Learning Research*, 18.

Moore, E. H. (1920). On the reciprocal of the general algebraic matrix. *Bulletin of the American Mathematical Society*, 26(9):394–395.

Mutoh, A. (2025). *GPpenalty: Penalized Likelihood in Gaussian Processes*. R package version 1.0.0.

Nielsen, H. B., Lophaven, S. N., and Søndergaard, J. (2002). Dace - A Matlab Kriging Toolbox.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Rasmussen, C. E. and Nickisch, H. (2010). Gaussian Processes for Machine learning (gpml) Toolbox. *Journal of Machine Learning Research*, 11(100):3011–3015.

Roustant, O., Ginsbourger, D., and Deville, Y. (2012a). DiceKriging, DiceOptim: Two R Packages for the Analysis of Computer Experiments by Kriging-Based Metamodeling and Optimization. *Journal of Statistical Software*, 51(1):1–55.

Roustant, O., Ginsbourger, D., and Deville, Y. (2012b). *DiceKriging: Kriging Methods for Computer Experimens*. R package version 1.6.0.

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989). Design and Analysis of Computer Experiments. *Statistical Science*, 4(4):409 – 423.

Santner, T. J., Williams, B. J., and Notz, W. (2003). *The Design and Analysis of Computer Experiments*. Springer, New York, NY.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, 58:267–288.

Westermann, P. and Evins, R. (2019). Surrogate modelling for sustainable building design – A review. *Energy and Buildings*, 198:170–186.

Worley, B. A. (1987). Deterministic uncertainty analysis. Technical report, Oak Ridge National Lab. (ORNL), Oak Ridge, TN (United States).

Yi, G., Shi, J. Q., and Choi, T. (2011). Penalized Gaussian Process Regression and Classification for High-Dimensional Nonlinear Data. *Biometrics*, 67:1285–1294.

Zhang, Y., Yao, W., Chen, X., and Ye, S. (2020). A penalized blind likelihood Kriging method for surrogate modeling. *Structural and Multidisciplinary Optimization*, 61:457 – 474.

Appendix

A Detailed Derivations for Section 3

A.1 Inverse covariance matrix under extreme parameter values

We consider an isotropic case, where the kernel function is defined as $\mathbf{R}_n^{ij} = R(x_i, x_j) = \exp\left(-\theta(x_i - x_j)^2\right)$. For large values of θ, if $x_i \neq x_j$, then $|x_i - x_j| > 0$, and thus $R(x_i, x_j) = \exp(-\theta(x_i - x_j)^2) \to 0$ as $\theta \to \infty$. When $x_i = x_j$, we have $R(x_i, x_i) = 1$ for all θ. Consequently, as $\theta \to \infty$, $\mathbf{R}(\mathcal{X}, \mathbf{X}_n) \approx \mathbf{0}$, and $\mathbf{R}_n \approx \mathbf{I}$. It follows that

$$\begin{aligned}
\boldsymbol{\mu}_\theta(\mathcal{X} \mid \mathcal{D}) &= \mathbf{R}(\mathcal{X}, \mathbf{X}_n)\mathbf{R}_n^{-1}\mathbf{y}_n \\
&= \mathbf{0} \\
\boldsymbol{\Sigma}_\Omega(\mathcal{X} \mid \mathcal{D}) &= \hat{\sigma}_\theta^2 \left(\mathbf{R}(\mathcal{X}, \mathcal{X}) - \mathbf{R}(\mathcal{X}, \mathbf{X}_n)\mathbf{R}_n^{-1}\mathbf{R}(\mathbf{X}_n, \mathcal{X})\right) \\
&= \hat{\sigma}_\theta^2 \mathbf{I} \, .
\end{aligned}$$

In contrast, when the penalized estimate for θ approaches 0, $R(x_i, x_j) \approx 1$, which implies $\mathbf{R}(\mathcal{X}, \mathbf{X}_n) \approx \mathbf{J}$. Consider now the modified covariance function, $R(x_i, x_j) = \exp\left(-\theta(x_i - x_j)^2\right) + g\mathbb{I}_{(i=j)}$. With the inclusion of the g term, as $\theta \to 0$, the covariance function becomes

$$\mathbf{R}_n \approx g\mathbf{I} + \mathbf{J} = g\mathbf{I} + \mathbf{1}\mathbf{1}^\top.$$

Applying the Sherman-Morrison formula, we obtain

$$\begin{aligned}
\mathbf{R}_n^{-1} &= (g\mathbf{I} + \mathbf{1}\mathbf{1}^\top)^{-1} \\
&= \frac{1}{g}\mathbf{I} - \frac{\frac{1}{g^2}}{1 + \mathbf{1}^\top \frac{1}{g}\mathbf{1}}\mathbf{J} \\
&= \frac{1}{g}\mathbf{I} - \frac{\frac{1}{g^2}}{1 + \frac{n}{g}}\mathbf{J} \\
&= \frac{1}{g}\left(\mathbf{I} - \frac{1}{g+n}\mathbf{J}\right) \\
&\approx \frac{1}{g}\left(\mathbf{I} - \frac{1}{n}\mathbf{J}\right) \, .
\end{aligned}$$

Here, the approximation $g + n \approx n$ holds since g is set to be small.

A.2 Limiting form of the predictive covariance matrix

The predictive covariance matrix under K-fold CV is given by

$$\mathbf{R}_\lambda(\mathbf{X}_k \mid \mathcal{D}_{-k}) = \mathbf{R}(\mathbf{X}^v, \mathbf{X}^v) - \mathbf{R}(\mathbf{X}^v, \mathbf{X}^t)\mathbf{R}_{n_t}^{-1}\mathbf{R}(\mathbf{X}^t, \mathbf{X}^v)$$

Let $\mathbf{R}_{\lambda,k} \equiv \mathbf{R}_\lambda(\mathbf{X}_k \mid \mathcal{D}_{-k})$. As $\lambda \to \infty$,

$$\begin{aligned}
\lim_{\lambda\to\infty} \mathbf{R}_{\lambda,k} &= \mathbf{J}_{n_v} + g\mathbf{I}_{n_v} - \mathbf{J}_{n_t \times n_v}\left(\frac{1}{g}\left(1 - \frac{1}{g+n_t}\mathbf{J}_{n_t}\right)\right)\mathbf{J}_{n_v \times n_t} \\
&= \mathbf{J}_{n_v} + g\mathbf{I}_{n_v} - \frac{n_t}{g}\mathbf{J}_{n_v} + \frac{n_t^2}{g(g+n_t)}\mathbf{J}_{n_v} \\
&= \mathbf{J}_{n_v} + g\mathbf{I}_{n_v} - \frac{n_t}{g+n_t}\mathbf{J}_{n_v} \\
&= \frac{g}{g+n_t}\mathbf{J}_{n_v} + g\mathbf{I}_{n_v} \\
&= g\left(\frac{1}{g+n_t}\mathbf{J}_{n_v} + \mathbf{I}_{n_v}\right) \, .
\end{aligned}$$

A.3 Limit of Mahalanobis distance

Mahalobis distance (MD) is defined as

$$\begin{aligned}\mathrm{MD}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) &= (\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k})^\top \boldsymbol{\Sigma}_{\lambda,k}^{-1} (\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k}) \\ &= \frac{1}{\hat{\sigma}^2_{\lambda,-k}} (\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k})^\top \mathbf{R}_\lambda^{-1} (\mathbf{X}_k \mid \mathcal{D}_{-k}) (\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k}) \ ,\end{aligned}$$

where $\hat{\sigma}^2_{\lambda,-k} = y_{-k}^\top \mathbf{R}_{-k}^{-1} \mathbf{y}_{-k}/(n - n_v)$. As $\lambda \to \infty$, $\mathbf{R}_{\lambda,k} = g\left(\frac{1}{g+n_t}\mathbf{J}_{n_v} + \mathbf{I}_{n_v}\right)$. Applying the Sherman-Morrison formula, we obtain

$$\begin{aligned}\mathbf{R}_{\lambda,k}^{-1} &= \frac{1}{g}\left(\mathbf{I}_{n_v} - \frac{\alpha}{1+\alpha n_v}\mathbf{J}_{n_v}\right) \text{ where } \alpha = \frac{1}{g+n_t} \\ &= \frac{1}{g}\left(\mathbf{I}_{n_v} - \frac{1}{g+n_t+n_v}\mathbf{J}_{n_v}\right) \\ &= \frac{1}{g}\left(\mathbf{I}_{n_v} - \frac{1}{g+n}\mathbf{J}_{n_v}\right) \\ &\approx \frac{1}{g}\left(\mathbf{I}_{n_v} - \frac{1}{n}\mathbf{J}_{n_v}\right) \ .\end{aligned}$$

Here, the approximation $g + n \approx n$ holds since g is set to be small. Therefore, the numerator of $\mathrm{MD}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda)$ simplifies to $g^{-1}\mathbf{y}_k^\top \left(\mathbf{I}_{n_v} - \frac{1}{n}\mathbf{J}_{n_v}\right)\mathbf{y}_k$.
As $\lambda \to \infty$,

$$\begin{aligned}\lim_{\lambda\to\infty} \mathrm{MD}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) &= \lim_{\lambda\to\infty} \frac{(\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k})^\top \mathbf{R}_{\lambda.k}^{-1} (\mathbf{y}_k - \boldsymbol{\mu}_{\lambda,k})}{\hat{\sigma}^2_{\lambda,-k}} \\ &\approx (n - n_v) \frac{\mathbf{y}_k^\top (\mathbf{I}_{n_v} - \frac{1}{n}\mathbf{J}_{n_v})\mathbf{y}_k}{\mathbf{y}_{-k}^\top (\mathbf{I}_{n_t} - \frac{1}{n}\mathbf{J}_{n_t})\mathbf{y}_{-k}} \\ &\equiv (n - n_v)\frac{\mathrm{SS}_k}{\mathrm{SS}_{-k}}\end{aligned}$$

where $\mathrm{SS}_{-k} = \mathbf{y}_{-k}^\top (\mathbf{I}_{n_t} - \frac{1}{n}\mathbf{J}_{n_t})\mathbf{y}_{-k}$.

A.4 Limit of Score

Score is defined as

$$\begin{aligned}\mathrm{Score}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) &= \mathrm{MD}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) + \log|\boldsymbol{\Sigma}_{\lambda,k}| \\ &= \mathrm{MD}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) + n_v \log(\hat{\sigma}^2_{\lambda,-k}) + \log|\mathbf{R}_{\lambda,k}| \ .\end{aligned}$$

As $\lambda \to \infty$, the variance term satisfies

$$\lim_{\lambda\to\infty} n_v \log(\hat{\sigma}^2_{\lambda,-k}) = n_v(-\log(g) - \log(n - n_v) + \log(\mathrm{SS}_{-k})) \ .$$

For the determinant term, we consider eigenvalues of $\mathbf{R}_{\lambda,k}$. As $\theta \to 0$, $\mathbf{R}_{\lambda,k} = g\left(\frac{1}{g+n_t}\mathbf{J}_{n_v} + \mathbf{I}_{n_v}\right)$. Since $\mathbf{J}_{n_v}$ has eigenvalues n_v (once) and 0 (multiplicity $n_v - 1$), the eigenvalues of $\mathbf{R}_{\lambda,k}$ are $g\left(1 + \frac{n_v}{g+n_t}\right)$ (once) and g (multiplicity $n_v - 1$). Therefore,

$$\log|\mathbf{R}_{\lambda,k}| = n_v \log(g) + \log\left(1 + \frac{n_v}{g+n_t}\right)$$

Combining terms, as $\lambda \to \infty$,

$$\begin{aligned}\lim_{\lambda\to\infty} \mathrm{Score}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) &= \lim_{\lambda\to\infty} \mathrm{MD}(\mathcal{D}_k \mid \mathcal{D}_{-k}, \lambda) - n_v \log(n - n_v) + n_v \log(\mathrm{SS}_{-k}) + \log\left(1 + \frac{n_v}{g+n_t}\right) \\ &\approx (n - n_v)\frac{\mathrm{SS}_k}{\mathrm{SS}_{-k}} + n_v \log\left(\frac{\mathrm{SS}_{-k}}{n - n_v}\right) + \log\left(\frac{n}{n_t}\right)\end{aligned}$$

Here, the approximation $g + n_t \approx n_t$ holds since g is set to be small.

B Test Functions

Lim Nonpolynomial. The two-dimensional Lim Nonpolynomial function (Lim et al., 2002) is defined as

$$f(\mathbf{x}) = \frac{1}{6}[(30 + 5x_1 \sin(5x_1))(4 + \exp(-5x_2)) - 100] \quad x_i \in [0, 1]$$

Franke. The two-dimensional Franke function (Franke, 1979) is defined as

$$\begin{aligned} f(\mathbf{x}) = {} & 0.75 \exp\left(-\frac{(9x_1 - 2)^2}{4} - \frac{(9x_2 - 2)^2}{4}\right) \\ & + 0.75 \exp\left(-\frac{(9x_1 + 1)^2}{49} - \frac{9x_2 + 1}{10}\right) \\ & + 0.5 \exp\left(-\frac{(9x_1 - 7)^2}{4} - \frac{(9x_2 - 3)^2}{4}\right) \\ & - 0.2 \exp\left(-(9x_1 - 4)^2 - (9x_2 - 7)^2\right), \quad x_i \in [0, 1]. \end{aligned}$$

Piston Simulation. The seven-dimensional Piston Simulation function (Ben-Ari and Steinberg, 2007) is defined as

$$f(\mathbf{x}) = 2\pi \sqrt{\frac{M}{k + S^2 \frac{P_0 V_0}{T_0} \frac{T_a}{V^2}}},$$

where

$$V = \frac{S}{2k}\left(\sqrt{A^2 + 4k\frac{P_0 V_0}{T_0} T_a} - A\right), \quad A = P_0 S + 19.62M - \frac{kV_0}{S}.$$

The input ranges are given by $M \in [30, 60]$, $S \in [0.005, 0.020]$, $V_0 \in [0.002, 0.010]$, $k \in [1000, 5000]$, $P_0 \in [90000, 110000]$, $T_a \in [290, 296]$, $T_0 \in [340, 360]$

Borehole. The eight-dimensional Borehole function (Worley, 1987) is defined as

$$f(\mathbf{x}) = \frac{2\pi T_u (H_u - H_l)}{\log(r/r_w)\left(1 + \frac{2LT_u}{\log(r/r_w) r_w^2 K_w}\right) + \frac{T_u}{T_l}}.$$

The input ranges are given by $r_2 \in [0.05, 0.15]$, $r \in [100, 50000]$, $T_u \in [63070, 115600]$, $H_u \in [990, 1110]$, $T_l \in [63.1, 116]$, $H_l \in [700, 820]$, $L \in [1120, 1680]$, $K_w \in [9855, 12045]$.

C Piston Slap Noise Data

x_1	x_2	x_3	x_4	x_5	x_6	Noise(dB)
71	16.8	21.0	2	1	0.98	56.75
15	15.6	21.8	1	2	1.30	57.65
29	14.4	25.0	2	1	1.14	53.97
85	14.4	21.8	2	3	0.66	58.77
29	12.0	21.0	3	2	0.82	56.34
57	12.0	23.4	1	3	0.98	56.85
85	13.2	24.2	3	2	1.30	56.68
71	18.0	25.0	1	2	0.82	58.45
43	18.0	22.6	3	3	1.14	55.50
15	16.8	24.2	2	3	0.50	52.77
43	13.2	22.6	1	1	0.50	57.36
57	15.6	23.4	3	1	0.66	59.64

Table 4: Training data for piston noise slap example in Section 5.