# Improved Most Likely Heteroscedastic Gaussian Process Regression via Bayesian Residual Moment Estimator

Qiu-Hu Zhang and Yi-Qing Ni

***Abstract*—This paper proposes an improved most likely heteroscedastic Gaussian process (MLHGP) algorithm to handle a kind of nonlinear regression problems involving input-dependent noise. The improved MLHGP follows the same learning scheme as the current algorithm by use of two Gaussian processes (GPs), with the first GP for recovering the unknown function and the second GP for modeling the input-dependent noise. Unlike the current MLHGP pursuing an empirical estimate of the noise level which is provably biased in most of local noise cases, the improved algorithm gives rise to an approximately unbiased estimate of the input-dependent noise. The approximately unbiased noise estimate is elicited from Bayesian residuals by the method of moments. As a by-product of this improvement, the expectation maximization (EM)-like procedure in the current MLHGP is avoided such that the improved algorithm requires only standard GP learnings to be performed twice. Four benchmark experiments, consisting of two synthetic cases and two real-world datasets, demonstrate that the improved MLHGP algorithm outperforms the current version not only in accuracy and stability, but also in computational efficiency.**

***Index Terms*—Gaussian process regression, most likely heteroscedastic Gaussian process, input-dependent noise, Bayesian residual, method of moments.**

## I. Introduction

Gaussian process (GP) has been proven to be a powerful Bayesian nonparametric method for solving nonlinear regression or multi-class classification problems [1]. It enables the realization of a probabilistic prediction within an elegant inference framework while holding excellent resilience to overfitting that often occurs in machine learning. In a standard GP regression model, the noise level is typically presumed to be constant throughout the input space. In many real-world problems [2]–[6], however, the observation variability heavily depends on the input. The misuse of such a strong assumption may give rise to a GP model with poor capability in the interpretation of heteroscedastic data. Moreover, it is likely to invalidate statistical hypothesis tests where the observed data is postulated to be independent and identically distributed.

Over the past two decades, various heteroscedastic Gaussian process (HGP) models [7]–[16] have been proposed to release the constant-noise assumption and allow the noise level to be variant across the input space. The HGP configuration typically makes use of two GPs, with one for modeling the latent function and the other for learning the input-dependent noise. A combination of the two GPs will generate a joint posterior distribution over the latent function and the input-dependent noise, that is non-Gaussian and no longer analytically intractable.

To obtain the numerical solution, we usually resort to Markov chain Monte Carlo (MCMC) samplings [7], [8] or analytical approximations [9]–[17]. The MCMC samplings are often viewed as a principled “gold standard” for inference in that the solutions of the MCMC samplings can converge to the exact non-Gaussian posterior when the sample size tends to infinity. However, the MCMC methods can be prohibitively expensive in large datasets. Rather, analytical approximations are recently more preferred as they achieve a trade-off between computational accuracy and efficiency. The expectation propagation (EP) approximations [9]–[11] are much faster than MCMC samplings, but they remain very costly for large-scale regression problems. The Laplace approximation [12] is more straightforward, utilizing a Gaussian distribution to approximate the joint posterior via the second-order Taylor expansion. This method, however, may produce a poor posterior approximation when it is highly skewed. A better analytical approximation with computational cost comparable to the Laplace method, is the variational heteroscedastic Gaussian process (VHGP) [13] in which the joint posterior is approximated by a two-factor variational distribution. The most likely noise approaches [14], [15] are deemed the most computationally attractive approximation, in which the noise posterior is simply replaced by a point estimate at its most likely level such that the predictive posterior distribution can be obtained analytically. Nevertheless, the most likely noise approaches may suffer from numerical inaccuracy and instability. For example, the most likely heteroscedastic Gaussian process (MLHGP) [14] as a typical representative of the most likely noise approaches, is not guaranteed to converge but rather might oscillate due to empirical estimation of the input-dependent noise. This flaw was

Manuscript received January 20, 2020; revised April 28, 2020; accepted May 24, 2020. Date of publication May 29, 2020; date of current version June 17, 2020. The associate editor coordinating the review of this manuscript and approving it for publication was Sotirios Chatzis. This work was supported in part by the Research Grants Council of the Hong Kong Special Administrative Region (SAR), China under Grant PolyU 152014/18E and in part by the Innovation and Technology Commission of the Hong Kong SAR Government, China under Grant K-BBY1. *(Corresponding author: Yi-Qing Ni.)*

Qiu-Hu Zhang is with the Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hung Hom, Hong Kong (e-mail: q.h.zhang@connect.polyu.hk).

Yi-Qing Ni is with the Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hung Hom, Hong Kong, and also with the National Engineering Research Center on Rail Transit Electrification and Automation (Hong Kong Branch), Hong Kong (e-mail: ceyqni@polyu.edu.hk).

Digital Object Identifier 10.1109/TSP.2020.2997940

later dealt with in the *maximum a posteriori* heteroscedastic Gaussian process (MAPHGP) [15], by introducing marginal likelihood of the data to penalize improper noise level. However, the MAPHGP tends to overfit severely when there exist many latent noise variables to learn. Other approximative approaches are also available in the literature and interested readers may refer to references [16]–[19].

While the most likely noise approaches have deficiency in numerical inaccuracy and instability, their computational efficiency is highly attractive as they require only standard GP inference. Therefore, there has been much interest in the use of the most likely noise approaches in practical applications [20]–[25]. With the intent to overcome numerical inaccuracy and instability of the most likely noise approaches, this study develops an improved MLHGP algorithm in terms of the moment estimation of Bayesian residuals. After attesting to the fact that the empirical estimate of the noise level in the current MLHGP is biased for most input-dependent noises, an approximately unbiased noise estimate is proposed based on the method of moments for Bayesian residuals. This refinement in the noise estimate can significantly benefit the most likely noise approaches in algorithmic accuracy and stability when dealing with regression problems with input-dependent noise. Moreover, the expectation maximization (EM)-like learning procedure in the current MLHGP is exempted such that the computational cost of the improved algorithm is only twice that of a standard GP. To validate the superiority and effectiveness of the proposed MLHGP, benchmark examples using synthetic datasets and real-world datasets are provided.

The rest of this paper is organized as follows. In Section II, the GP regression model is briefed. In Section III, the current MLHGP is introduced, followed by the improved algorithm proposed in this study. In Section IV, the improved MLHGP is validated by using four benchmark experiments in conjunction with detailed comparisons to the standard GP, VHGP and MLHGP. Finally, conclusions and further lines of research are presented in Section V.

## II. Gaussian Process

The nonlinear regression is aimed at recovering an unknown function $f : R^d \to R$ from a dataset $D = \{\mathbf{x}_i, y_i\}_{i=1}^{n}$, where $\mathbf{x}_i \in R^d$ denotes the input vector of dimension $d$ and $y_i \in R$ denotes a scalar of the observed output such that

$$y_i = f(\mathbf{x}_i) + \varepsilon_i \text{ with } \varepsilon_i \sim \mathcal{N}\left(0, g^2(\mathbf{x}_i)\right) \tag{1}$$

where the observation error $\varepsilon_i$ is typically assumed to be independently and normally distributed with mean zero and variance $g^2(\mathbf{x}_i)$. The noise variance $g^2(\mathbf{x}_i)$ can be constant or varying across the input space. For the sake of brevity, we denote here the true function value $f_i = f(\mathbf{x}_i)$ and the noise standard deviation $g_i = g(\mathbf{x}_i)$. The inputs, outputs, function values and noise standard deviations are then aggregated into $X = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^{\mathrm{T}}$, $\mathbf{y} = (y_1, \ldots, y_n)^{\mathrm{T}}$, $\mathbf{f} = (f_1, \ldots, f_n)^{\mathrm{T}}$ and $\mathbf{g} = (g_1, \ldots, g_n)^{\mathrm{T}}$, respectively. In this study, we consider the general regression problems, mapping from an input $\mathbf{x}_i$ to an output $f(\mathbf{x}_i)$, which do not involve specific application backgrounds such as robotic control with initial conditions or output constraints.

The GP is a nonparametric Bayesian modeling for the unknown function, that can be fully specified by a mean function $m(\mathbf{x})$ and a covariance function $k(\mathbf{x}, \mathbf{x}')$ [1]. A simplifying assumption is to place a zero-mean GP prior over the function value, given as

$$p(\mathbf{f}|X) = \mathcal{N}(\mathbf{0}, \mathbf{K}) \tag{2}$$

where $\mathbf{K}$ is the covariance matrix with entries $[\mathbf{K}]_{ii}$ calculated from the covariance function $k(\mathbf{x}_i, \mathbf{x}_j)$ at input points $\mathbf{x}_i$ and $\mathbf{x}_j$. Many covariance functions are available to define a GP prior, such as squared exponential (SE) or Matérn kernels [1]. The present study is mainly focused on the SE kernel that is infinitely differentiable, expressed by

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \eta^2 \exp\left[\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 / \left(2l^2\right)\right] \tag{3}$$

where $\|\cdot\|$ denotes the Euclidean distance between input locations $\mathbf{x}_i$ and $\mathbf{x}_j$, $\eta$ is the signal amplitude, and $l$ is the characteristic length-scale. The SE kernel parameterized by $\boldsymbol{\theta}_f = \{\eta, l\}$ is a measure of similarity between two observations. The observed output $\mathbf{y}$ and the function value $f_*$ at test input $\mathbf{x}_*$ are jointly Gaussian distributed as

$$\begin{bmatrix} \boldsymbol{y} \\ f_* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ 0 \end{bmatrix}, \begin{bmatrix} \mathbf{K} + \mathbf{S} & \mathbf{k}_* \\ \mathbf{k}_*^{\mathrm{T}} & \mathrm{k}_{**} \end{bmatrix}\right) \tag{4}$$

where $\mathbf{S}$ is the diagonal matrix of noise variances with entries $[\mathbf{S}]_{ii} = g_i^2$, $\mathbf{k}_*$ is the covariance vector calculated by $k(\mathbf{x}_*, \mathbf{x}_i)$ between test input $\mathbf{x}_*$ and training input $\mathbf{x}_i$, and $k_{**}$ is the prior variance calculated from $k(\mathbf{x}_*, \mathbf{x}_*)$ at $\mathbf{x}_*$. The use of the conditional identity of a multivariate Gaussian distribution results in the posterior distribution of the function value $f_*$ at test input $\mathbf{x}_*$ as

$$p(f_*|\mathbf{x}_*, \boldsymbol{\theta}_f, \mathbf{g}, D) = \mathcal{N}\left(\mu_{f_*}, \sigma_{f_*}^2\right), \text{ where} \tag{5}$$

$$\mu_{f_*} = \mathbf{k}_*^{\mathrm{T}}(\mathbf{K} + \mathbf{S})^{-1}\mathbf{y} \tag{6}$$

$$\sigma_{f_*}^2 = k_{**} - \mathbf{k}_*^{\mathrm{T}}(\mathbf{K} + \mathbf{S})^{-1}\mathbf{k}_* \tag{7}$$

The posterior distribution over the test output $y_*$ can simply be obtained by adding the noise variance $g_*^2$ at test location $\mathbf{x}_*$ to the posterior variance of the function value $f_*$ as

$$p(y_*|\mathbf{x}_*, \boldsymbol{\theta}_f, \mathbf{g}, g_*, D) = \mathcal{N}\left(\mu_{y_*}, \sigma_{y_*}^2\right), \text{ where} \tag{8}$$

$$\mu_{y_*} = \mu_{f_*} = \mathbf{k}_*^{\mathrm{T}}(\mathbf{K} + \mathbf{S})^{-1}\mathbf{y} \tag{9}$$

$$\sigma_{y_*}^2 = \sigma_{f_*}^2 + g_*^2 = k_{**} - \mathbf{k}_*^{\mathrm{T}}(\mathbf{K} + \mathbf{S})^{-1}\mathbf{k}_* + g_*^2 \tag{10}$$

In realistic modeling situations, there is no access to either the kernel parameters $\boldsymbol{\theta}_f$ or the noise level $g(\mathbf{x})$, and they must be learned from the data. In the standard GP, the noise level is assumed to be constant throughout the input space, and thus we have the noise power $g^2(\mathbf{x}) \equiv \sigma_n^2$ and the noise matrix $\mathbf{S} \equiv \sigma_n^2\mathbf{I}$. The unknown parameters, including kernel parameters $\boldsymbol{\theta}_f$ and noise variance $\sigma_n^2$ are then collectively referred to as hyperparameters of the GP model, denoted as $\boldsymbol{\theta}_y = \{\boldsymbol{\theta}_f, \sigma_n^2\}$, that can be learned by maximizing the log marginal likelihood of the data

$$\begin{aligned}\log p(\mathbf{y}|X, \boldsymbol{\theta}_y) = & -\frac{1}{2}\mathbf{y}^{\mathrm{T}}\left(\mathbf{K} + \sigma_n^2\mathbf{I}\right)^{-1} - \frac{1}{2}\log\left|\mathbf{K} + \sigma_n^2\mathbf{I}\right| \\ & - \frac{n}{2}\log(2\pi)\end{aligned} \tag{11}$$

This is known as the type II maximum likelihood (ML-II) estimate of the hyperparameters $\boldsymbol{\theta}_y$, which can be obtained by an optimization algorithm [26] in pursuit of an acceptable local maximum or a global optimum if possible.

In the HGP, another GP needs to be built for modeling the log noise level $z_i = \log g_i^2$, with a separate covariance function $k_z(\mathbf{x}, \mathbf{x}')$ parameterized by $\boldsymbol{\theta}_z$. As a result, two GPs are involved in the HGP, with the first for recovering the unknown function (the y-process) and the second for learning the input-dependent noise level (the z-process). The resulting predictive posterior distribution over test output $y_*$ is given by the following integral

$$\begin{aligned} &p\left(y_*|\mathbf{x}_*, \boldsymbol{\theta}_f, \boldsymbol{\theta}_z, D\right) \\ &\quad = \iint p\left(y_*|\mathbf{x}_*, \boldsymbol{\theta}_f, \mathbf{z}, z_*, D\right) p\left(\mathbf{z}, z_*|\mathbf{x}_*, \boldsymbol{\theta}_z, D\right) d\mathbf{z} dz_* \end{aligned} \tag{12}$$

where $\mathbf{z} = (z_1, \ldots, z_n)^{\mathrm{T}}$ are the log noise variances at training inputs $X$, and $z_*$ is the log noise variance at test input $\mathbf{x}_*$. Given the noise levels $(\mathbf{z}, z_*)$, the integral in Eq. (12) is analytically tractable and the posterior distribution of the test output $y_*$ remains Gaussian with the posterior mean and variance given by Eqs. (9) and (10), respectively. Nevertheless, in regard to the full posterior of the noise levels $p(\mathbf{z}, z_*|\mathbf{x}_*, \boldsymbol{\theta}_z, D)$ the integral is no longer solvable analytically and thus one has to employ the MCMC samplings or analytical approximations as afore mentioned. In the next section, after introducing the current MLHGP, we will present an improved algorithm based on the method of moments for Bayesian residuals.

## III. Heteroscedastic Gaussian Process

The MLHGP [14] is very simple and computationally attractive in dealing with regression problems with input-dependent noise, in that the full posterior distribution of the varying noise is simply replaced by a point estimate at the most likely value such that the predictive posterior over the test output can be treated analytically. In the MLHGP, the noise posterior $p(\mathbf{z}, z_*|\mathbf{x}_*, \boldsymbol{\theta}_z, D)$ is approximated as

$$p\left(\mathbf{z}, z_*|\mathbf{x}_*, \boldsymbol{\theta}_z, D\right) \approx \delta\left(\tilde{\mathbf{z}}, \tilde{z}_*\right) \tag{13}$$

where $(\tilde{\mathbf{z}}, \tilde{z}_*)$ is the most likely log noise level, and $\delta$ is the Dirac delta function with $\delta(\tilde{\mathbf{z}}, \tilde{z}_*) = 1$ when $\tilde{\mathbf{z}} = \tilde{z}_*$ and zero otherwise. The integral in Eq. (12) is thus approximated as

$$\begin{aligned} &p\left(y_*|\mathbf{x}_*, \boldsymbol{\theta}_f, \boldsymbol{\theta}_z, D\right) \\ &\quad \approx \iint p\left(y_*|\mathbf{x}_*, \boldsymbol{\theta}_f, \mathbf{z}, z_*, D\right) \delta\left(\tilde{\mathbf{z}}, \tilde{z}_*\right) d\mathbf{z} dz_* \\ &\quad \approx p\left(y_*|\mathbf{x}_*, \boldsymbol{\theta}_f, \tilde{\mathbf{z}}, \tilde{z}_*, D\right) \end{aligned} \tag{14}$$

The most likely noise level is typically at the mode of its noise posterior, given by

$$\left(\tilde{\mathbf{z}}, \tilde{z}_*\right) = \underset{(\mathbf{z}, z_*)}{\mathrm{argmax}} \log p\left(\mathbf{z}, z_*|\mathbf{x}_*, \boldsymbol{\theta}_z, D\right) \tag{15}$$

As the input-dependent noise is modeled by a GP as well, its posterior is also normally distributed and thus the most likely noise level is simply given as

```
Algorithm 1: Most Likely Heteroscedastic Gaussian Process.
  1. Train a standard GP $G_1$ on the training dataset
     $D = \{\mathbf{x}_i, y_i\}_{i=1}^n$ and estimate the posterior
     distribution over training outputs
     $y_i|\mathbf{x}_i, \boldsymbol{\theta}_y, D \sim \mathcal{N}(\mu_{y_i}, \sigma_{y_i}^2)$;
  2. Estimate empirically noise variances
     $g_i^2 = \frac{1}{2}[(y_i - \mu_{y_i})^2 + \sigma_{y_i}^2]$ and build a new training
     dataset $D' = \{\mathbf{x}_i, z_i\}_{i=1}^n$ with $z_i = \log g_i^2$;
  3. Train another standard GP $G_2$ on the new dataset $D'$
     and estimate log noise variances
     $z_i|\mathbf{x}_i, \boldsymbol{\theta}_z, D' \sim \mathcal{N}(\mu_{z_i}, \sigma_{z_i}^2)$;
  4. Train a heteroscedastic GP $G_3$ on the dataset $D$ with
     the most likely noise variances $\tilde{g}_i^2 = e^{\mu_{z_i}}$ to update the
     posterior distribution over training outputs
     $y_i|\mathbf{x}_i, \boldsymbol{\theta'}_f, \tilde{\mathbf{g}}^2, D \sim \mathcal{N}({\mu'}_{y_i}, {\sigma'}^2_{y_i})$;
  5. If not converged, set $G_1 = G_3$ and go back to step 2.
     Otherwise, make prediction on future observations
     $y_*|\mathbf{x}_*, \boldsymbol{\theta'}_f, \tilde{\mathbf{g}}^2, \tilde{g}_*^2, D \sim \mathcal{N}(\mu_{y_*}, \sigma_{y_*}^2)$.
```

$$\left(\tilde{\mathbf{z}}, \tilde{z}_*\right) = \left(\boldsymbol{\mu}_{\mathbf{z}}, \mu_{\mathrm{z}_*}\right) \tag{16}$$

where $\boldsymbol{\mu}_{\mathbf{z}}$ and $\mu_{\mathrm{z}*}$ are respectively the posterior means of log noise levels at training points $X$ and test point $\mathbf{x}_*$. The integral in Eq. (12) is thus

$$\begin{aligned} p\left(y_*|\mathbf{x}_*, \boldsymbol{\theta}_f, \boldsymbol{\theta}_z, D\right) &\approx p\left(y_*|\mathbf{x}_*, \boldsymbol{\theta}_f, \tilde{\mathbf{z}}, \tilde{z}_*, D\right) \\ &= p\left(y_*|\mathbf{x}_*, \boldsymbol{\theta}_f, \boldsymbol{\mu}_{\mathbf{z}}, \mu_{\mathrm{z}_*}, D\right) \end{aligned} \tag{17}$$

Hence, the predictive posterior distribution of the test output is Gaussian, with its mean and variance given by Eqs. (9) and (10), respectively.

### A. Noise Estimation in Current MLHGP

The estimation of the most likely noise level is at the core of the MLHGP approach. An empirical estimate of the noise level is employed in the current MLHGP [14] as

$$g_i^2 = \frac{1}{s} \sum_{j=1}^{s} 0.5(y_i - y_i^j)^2 \tag{18}$$

where $s$ is the sample size, and $y_i^j$ are samples from the posterior predictive distribution of the training output $y_i$ that is Gaussian with mean $\mu_{y_i}$ and variance $\sigma_{y_i}^2$ given in Eqs. (9) and (10) respectively. In fact, the above empirical estimate can be simplified by the Gaussian identity as

$$g_i^2 = \frac{1}{2}\left[(y_i - \mu_{y_i})^2 + \sigma_{y_i}^2\right] \tag{19}$$

This simplification can not only reduce the computational cost of the current MLHGP, but also can significantly enhance numerical stability of it. As such, a new training dataset $D' = \{\mathbf{x}_i, z_i\}_{i=1}^n$ with $z_i = \log\{\frac{1}{2}[(y_i - \mu_{y_i})^2 + \sigma_{y_i}^2]\}$ can be built to train another GP for estimating the most likely noise variance, given by $\tilde{g}_i^2 = e^{\mu_{z_i}}$. The current MLHGP is delineated in Algorithm 1.

The MLHGP is much simpler and computationally more efficient compared to MCMC sampling methods and other analytical approximations, but the algorithm is not guaranteed to converge as the empirical estimate of the noise level given in Eq. (19) is biased. The expectation of the empirical noise estimate is

$$\begin{aligned}
&\mathrm{E}\left\{\frac{1}{2}\left[\left(y_i-\mu_{y_i}\right)^2+\sigma_{y_i}^2\right]\right\}\\
&\quad=\mathrm{E}\left\{\frac{1}{2}\left[\left(y_i-\mu_{y_i}\right)^2+\sigma_{f_i}^2+\sigma_n^2\right]\right\}\\
&\quad=\mathrm{E}\left\{\frac{1}{2}\left[\left(y_i-f_i+f_i-\mu_{f_i}\right)^2+\sigma_{f_i}^2+\sigma_n^2\right]\right\}\\
&\quad=\mathrm{E}\left\{\frac{1}{2}\left[\left(\varepsilon_i+f_i-\mu_{f_i}\right)^2+\sigma_{f_i}^2+\sigma_n^2\right]\right\}\\
&\quad=\mathrm{E}\left\{\frac{1}{2}\left[\varepsilon_i^2+2\varepsilon_i\left(f_i-\mu_{f_i}\right)+\left(f_i-\mu_{f_i}\right)^2+\sigma_{f_i}^2+\sigma_n^2\right]\right\}\\
&\quad=\frac{1}{2}\left[g_i^2+\left(f_i-\mu_{f_i}\right)^2+\sigma_{f_i}^2+\sigma_n^2\right] \qquad (20)
\end{aligned}$$

where $\mu_{f_i}$ and $\sigma_{f_i}^2$ are respectively the posterior mean and variance of the function value $f_i$ at training input $\mathbf{x}_i$, given by Eqs. (6) and (7); $\sigma_n^2$ is the global noise variance estimated in the y-process; and $g_i^2$ is the true noise variance at $\mathbf{x}_i$. The item $(f_i - \mu_{f_i})$ being the difference between the true function value and its expected value is termed herein the modeling error. When the training data are enough and the first GP for learning the function value is well defined, the modeling error $(f_i - \mu_{f_i})$ and the modeling variability $\sigma_{f_i}^2$ can be neglected. The expectation of the empirical noise estimate is approximated as

$$\mathrm{E}\left\{\frac{1}{2}\left[\left(y_i-\mu_{y_i}\right)^2+\sigma_{y_i}^2\right]\right\}\approx\frac{1}{2}\left(g_i^2+\sigma_n^2\right) \qquad (21)$$

When the noise level is fixed, the global noise variance $\sigma_n^2$ estimated in the y-process can be a good approximation for each local noise level $g_i^2$ and thus

$$\mathrm{E}\left\{\frac{1}{2}\left[\left(y_i-\mu_{y_i}\right)^2+\sigma_{y_i}^2\right]\right\}\approx g_i^2 \qquad (22)$$

In such case, the empirical noise estimate in Eq. (19) can be approximately unbiased. However, when the noise level is input-dependent, the majority of local noise levels $g_i^2$ will not be equal to the estimated global noise variance $\sigma_n^2$ and thus we have

$$\mathrm{E}\left\{\frac{1}{2}\left[\left(y_i-\mu_{y_i}\right)^2+\sigma_{y_i}^2\right]\right\}\neq g_i^2 \qquad (23)$$

Clearly, the empirical noise estimate in the current MLHGP is biased for most of local noise cases if the noise level is varying in the input domain.

### B. Noise Estimation in Improved MLHGP

In this section, an approximately unbiased noise estimate is proposed based on the moment estimation of regression residuals. In Gaussian process regression, residuals $r_i$ are the difference between the observed outputs $y_i$ and the corresponding posterior means $\mu_{y_i}$ at $\mathbf{x}_i$ [27], [28],

$$r_i = y_i - \mu_{y_i} \qquad (24)$$

These residuals are referred to as Bayesian residuals by contrast with classical residuals in ordinary least square regression [29]; the latter are the difference between the observed outputs $y_i$ and the corresponding point estimates $\hat{y}_i$. The Bayesian residuals $r_i$ can be rewritten as

$$r_i = y_i - \mu_{y_i} = y_i - f_i + f_i - \mu_{y_i} = \varepsilon_i + (f_i - \mu_{f_i}) \qquad (25)$$

Apparently, each Bayesian residual $r_i$ comprises two items: $\varepsilon_i$ which is the observation error and $(f_i - \mu_{f_i})$ which is the modeling error. The observation error $\varepsilon_i$ is a random variable, while the modeling error $(f_i - \mu_{f_i})$ is a deterministic variable. Thus, each Bayesian residual $r_i$ is also a random variable, which is normally distributed with mean $\mu_{r_i} = f_i - \mu_{f_i}$ and variance $\sigma_{r_i}^2 = g_i^2$,

$$r_i \sim \mathcal{N}\left(\mu_{r_i}, \sigma_{r_i}^2\right) \qquad (26)$$

The residual $r_i$ and the expected function value $\mu_{f_i}$ are available in the first GP, while the true function value $f_i$ and the local noise variance $g_i^2$ have to be estimated from the data.

Regression residuals can be utilized to estimate the input-dependent noise in that the dispersion of the residual series is controlled by the varying noise level. Assuming the input-dependent noise can be depicted by a smooth function, one can extend regression techniques originally for recovering the underlying function $f(\mathbf{x}_i)$ to estimate the noise function $g(\mathbf{x}_i)$. Typically, regression techniques are performed on the transformed residuals $z_i = \mathrm{T}(r_i)$ such as the absolute residuals $z_i = |r_i|$ or the squared residuals $z_i = |r_i|^2$, rather than the raw residuals $r_i$, to facilitate recognition of the dispersion pattern of the residuals (z-function). Yet, the obtained z-function by fitting a curve for the transformed residuals $z_i$ may not provide an unbiased estimate for the noise function $g(\mathbf{x}_i)$, depending upon the adopted transformation function $\mathrm{T}(r_i)$. Therefore, it is necessary to calibrate the obtained z-function and make it unbiased for the input-dependent noise. Otherwise, the input-dependent noise level $g(\mathbf{x}_i)$ could be globally underestimated or overestimated.

The method of moments is a common practice for parameter estimation in statistics [30] and it enables to provide an unbiased estimate for the parameters of interest. For the input-dependent noise, its local levels can be derived from statistical moments of Bayesian residuals. Various moments, such as raw or central moments, and raw or central absolute moments, of Bayesian residuals are available to estimate the local noise levels. In this study, the raw absolute moments of the residuals are preferred because each order of the raw absolute moments of the residuals contains information about the noise power. The $v$th raw absolute moment of the residual $r_i$ at training point $\mathbf{x}_i$ is

TABLE I
TYPICAL VALUES OF APPROXIMATE CORRECTION FACTOR

| $v$ | $s(v)$ |
|---|---|
| 1 | $\sqrt{\pi/2}$ |
| 2 | 1 |
| 3 | $\sqrt{\pi/8}$ |
| 4 | 1/3 |

given by

$$\mathrm{E}\left\{|r_i|^v\right\} = \sigma_{r_i}^v / s\left(v\right) = g_i^v / s\left(v\right) \tag{27}$$

where the correction factor $s(v)$ depends on the moment order $v$, given by [31]

$$s\left(v\right) = \sqrt{\pi}\psi\left(-v/2, 1/2; -\mu_{r_i}^2 / \left(2\sigma_{r_i}^2\right)\right) \Big/ \left[2^{v/2}\Gamma\left((v+1)/2\right)\right] \tag{28}$$

where $\psi(\cdot)$ is the Kummer's confluent hypergeometric function and $\Gamma(\cdot)$ is the gamma function. The local noise level $g_i^v$ is thus obtained as

$$g_i^v = \mathrm{E}\left\{|r_i|^v\right\} s\left(v\right) \tag{29}$$

Clearly, $\mathrm{E}\{|r_i|^v\}s(v)$ is an unbiased estimate of the local noise level $g_i^v$ at $\mathbf{x}_i$.

When the first GP in the y-process for learning the function value is well defined, the modeling error $\mu_{r_i} = f_i - \mu_{f_i}$ can be neglected and we have $\psi(-v/2, v/2; 0) = 1$. The raw absolute moment of the residual degenerates to the central absolute moment of it. The correction factor can be approximated as

$$s\left(v\right) \approx \sqrt{\pi} / \left[2^{v/2}\Gamma\left((v+1)/2\right)\right] \tag{30}$$

Table I gives some typical values of the approximate correction factor. Particularly, the first raw absolute moment of the residual $(v = 1)$ is the absolute residual, while the second raw absolute moment of the residual $(v = 2)$ is the squared residual. An approximately unbiased estimate of the noise standard deviation $g_i$ at training point $\mathbf{x}_i$ is

$$g_i = \mathrm{E}\left\{|r_i|\right\} s\left(1\right) \approx \sqrt{\pi/2}\mathrm{E}\left\{|r_i|\right\} \tag{31}$$

Similarly, an approximately unbiased estimator of the noise variance $g_i^2$ at $\mathbf{x}_i$ is

$$g_i^2 = \mathrm{E}\left\{|r_i|^2\right\} s\left(2\right) \approx \mathrm{E}\left\{r_i^2\right\} \tag{32}$$

As a result, a new data $D' = \{\mathbf{x}_i, z_i\}_{i=1}^n$ with $z_i = |r_i|^v$ can be built to train a second standard GP to estimate the most likely noise levels $\tilde{g}_i^v = \mu_{\mathrm{z}_i}s(v)$ at training point $\mathbf{x}_i$ and $\tilde{g}_*^v = \mu_{\mathrm{z}_*}s(v)$ at test point $\mathbf{x}_*$. Interestingly, it is seen that $z_i = |r_i|^v$ is just what we need to transform the residuals $r_i$ before using a regression technique to estimate the noise function $g(\mathbf{x}_i)$. However, it should be noted that in the second GP, we are using a Gaussian approximation to the transformed residuals $z_i = |r_i|^v$ that are in general non-Gaussian and even non-negative. Such approximation can be reasonable in the improved MLGHP as we care only the mean function of the second GP (it defines the most likely noise levels), rather than the full distribution of it. Thus, the most likely noise levels are required to be refined to $\tilde{g}_i^v = \max(0, \mu_{\mathrm{z}_i}s(v))$ or $\tilde{g}_*^v = \max(0, \mu_{\mathrm{z}_*}s(v))$ to ensure a nonnegative noise level. Besides the input-dependent noise level being better estimated, the EM-like iteration algorithm required in the current MLHGP for iteratively learning the function value and the noise level is avoided. The improved MLHGP is elucidated in Algorithm 2. In principle, any order of the raw absolute moment of Bayesian residuals is acceptable to estimate the input-dependent noise level, but in practice lower orders ($v = 1$ or $v = 2$) are preferable because they are easy to compute and numerically more stable.

**Algorithm 2:** Improved Most Likely Heteroscedastic Gaussian Process.

```
1. Train a standard GP $G_1$ on the training dataset
   $D = \{\mathbf{x}_i, y_i\}_{i=1}^n$ and estimate the posterior
   distribution over training outputs
   $y_i|\mathbf{x}_i, \boldsymbol{\theta}_y, D \sim \mathcal{N}(\mu_{y_i}, \sigma_{y_i}^2)$;
2. Calculate regression residuals $r_i = y_i - \mu_{y_i}$ and build
   a new training dataset $D' = \{\mathbf{x}_i, z_i\}_{i=1}^n$ with
   $z_i = |r_i|^v$;
3. Train another standard GP $G_2$ on the new dataset $D'$
   and estimate the input-dependent noise levels
   $z_i|\mathbf{x}_i, \boldsymbol{\theta}_z, D' \sim \mathcal{N}(\mu_{z_i}, \sigma_{z_i}^2)$;
4. Update the most likely noise levels
   $\tilde{g}_i^v = \max(0, \mu_{\mathrm{z}_i}s(v))$ with
   $s(v) \approx \sqrt{\pi}/[2^{v/2}\Gamma((v+1)/2))]$;
5. Make prediction on future observations
   $y_*|\mathbf{x}_*, \boldsymbol{\theta}_f, \tilde{\mathbf{g}}, \tilde{g}_*, D \sim \mathcal{N}(\mu_{y_*}, \sigma_{y_*}^2)$.
```

## IV. EXPERIMENTS

In this section, the performance of five GPs will be compared, which are: GP—the standard Gaussian process, VHGP—the variational heteroscedastic Gaussian process, MLHGP—the current most likely heteroscedastic Gaussian process, IMLHGP1—the improved MLHGP using absolute residual $(v = 1)$, IMLHGP2—the improved MLHGP using squared residual $(v = 2)$. The first GP is said to be homoscedastic, while the other four GPs are heteroscedastic. The predictive performance of the five GPs is assessed by using four benchmark experiments, consisting of two synthetic cases and two real-world datasets, that have been employed to verify other HGPs [7], [8], [14], [15].

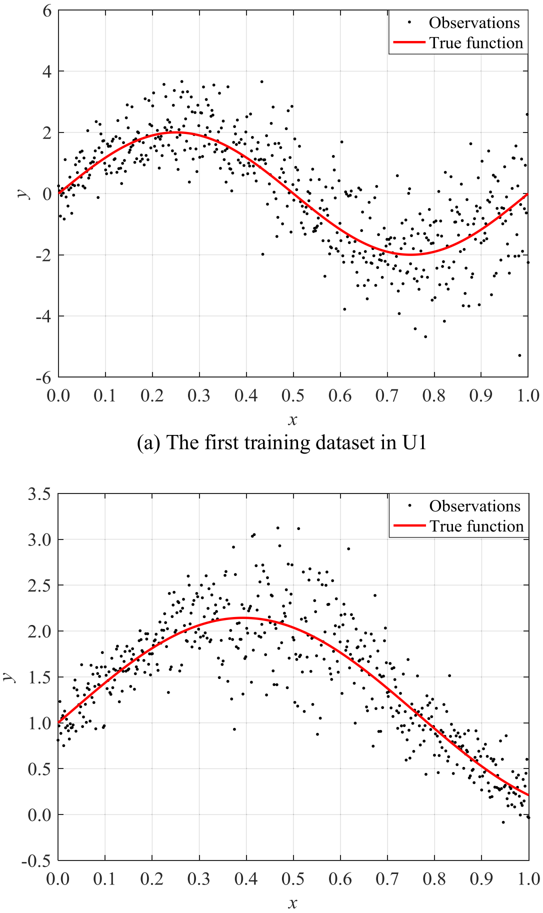
### A. Benchmark Experiments

The two synthetic benchmark experiments (U1 and U2) are both one-dimensional nonlinear regression problems with input-dependent noise. In the first synthetic experiment [7] the noise rate increases linearly with the input; but in the second one [8] the noise rate depends nonlinearly on the input. For the sake of simplicity, the observed output $y_i$ is rewritten as

$$y_i = f_i + g_i e_i \text{ with } e_i \sim \mathcal{N}\left(0, 1\right) \tag{33}$$

TABLE II
DESCRIPTION OF TWO SYNTHETIC BENCHMARK EXPERIMENTS

| Experiments | True Function<br>$f(x)$ | Noise standard deviation<br>$g(x)$ |
|---|---|---|
| U1 | $2\sin(2\pi x)$ | $0.5 + x$ |
| U2 | $[1+\sin(4x)]^{1.1}$ | $0.2 + 0.3\exp[-30(x-0.5)^2]$ |



(a) The first training dataset in U1

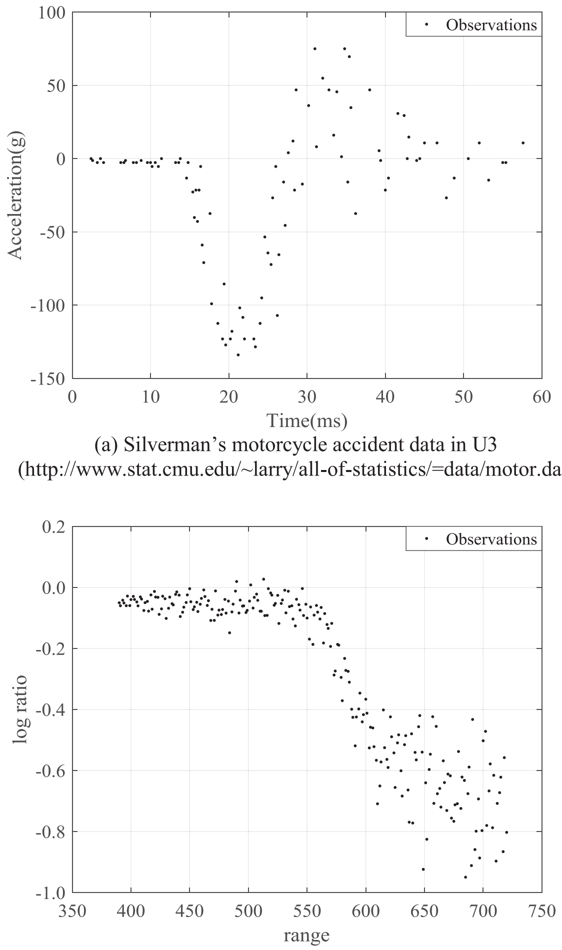
(b) The first training dataset in U2

Fig. 1. Examples of training datasets in the two synthetic benchmark experiments, with solid red lines depicting the true function values.

where $f_i$ is the true function value at input $\mathbf{x}_i$, $g_i$ is the true noise standard deviation at the same location, and $e_i$ is a standard normal random variable. More detailed information about the two synthetic experiments is given in Table II.

For each of the synthetic experiments, 100 training datasets were generated using the same program but different random seeds, with each training dataset consisting of $n = 500$ samples uniformly drawn from the input range [0,1]. Fig. 1 gives examples of the training datasets in the two synthetic experiments. A test dataset with $N = 1000$ samples was also generated to evaluate the performance of a trivial GP model.

Benchmark experiments were also conducted on Silverman's motorcycle accident dataset [2] and Sigrist's lidar dataset [3]. The motorcycle dataset consists of 94 observations (Fig. 2(a)), while the lidar dataset is composed of 221 observations (Fig. 2(b)). For the two real-world datasets, 100 training datasets were generated, using 90% of the observations for training and the remaining 10% for testing.



(a) Silverman's motorcycle accident data in U3
(http://www.stat.cmu.edu/~larry/all-of-statistics/=data/motor.dat)

(b) Sigrist's lidar data in U4
(http://www2.stat.duke.edu/~wjang/teaching/S05-293/data/lidar.html)

Fig. 2. Real-world heteroscedastic datasets for benchmark experiments.

## B. Predictive Performance Assessment

To quantify predictive performance of the five GPs in dealing with input-dependent regression problems, the standardized mean squared error (SMSE) with respect to the true function values in relation to a trivial GP model is first calculated as

$$\mathrm{SMSE}\,(f) = \frac{1}{N}\sum_{i=1}^{N}\frac{\left(\mu_{f_{*,i}} - f_{*,i}\right)^2}{\mathrm{var}\,(\mathbf{f}_*)} \tag{34}$$

where $\mu_{f_{*,i}}$ is the posterior mean of the function value $f_{*,i}$ at test input $\mathbf{x}_{*,i}$, $\mathrm{var}(\mathbf{f}_*)$ is the variances of the true function values at all test points $X_* = (\mathbf{x}_{*,i}, \ldots, \mathbf{x}_{*,N})^{\mathrm{T}}$, and $N$ is the test dataset size. As for real-world data, the true function values $f_{*,i}$ may not be available and one may have to use their noisy values $y_{*,i}$ (the testing outputs) as alternatives. In this regard, $\mathrm{SMSE}(f)$ should be replaced by $\mathrm{SMSE}(y)$. Then the SMSE with respect to the true noise standard deviation is computed by

$$\mathrm{SMSE}\,(g) = \frac{1}{N}\sum_{i=1}^{N}\frac{\left(\mu_{g_{*,i}} - g_{*,i}\right)^2}{\mathrm{var}\,(\mathbf{g}_*)} \tag{35}$$

where $\mu_{g_{*,i}}$ is the posterior mean of the noise standard deviation $g_{*,i}$ at input $\mathbf{x}_{*,i}$, and $\mathrm{var}(\mathbf{g}_*)$ is the variances of the true noise standard deviations at all test points. As for real-world data, the SMSE($g$) is not available. Finally, the average negative log probability density (NLPD) of the test outputs in regard to a trivial GP model is evaluated as

$$\begin{aligned}\mathrm{NLPD}\,(y) &= -\frac{1}{N}\sum_{i=1}^{N}\mathrm{log}p\,(y_{*,i}|\mathbf{x}_{*,i},D)\\ &= \frac{1}{2N}\sum_{i=1}^{N}\log\left(2\pi\sigma^2_{y_{*,i}}\right)\frac{1}{N}\sum_{i=1}^{N}\frac{\left(y_{*,i}-\mu_{y_{*,i}}\right)^2}{2\sigma^2_{y_{*,i}}}\end{aligned} \tag{36}$$

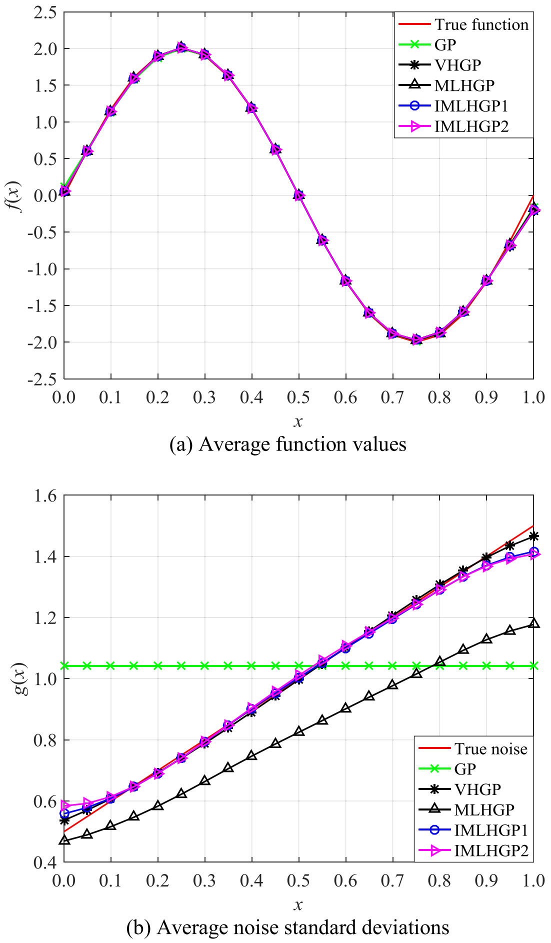
where $\mu_{y_{*,i}}$ and $\sigma^2_{y_{*,i}}$ are respectively the posterior mean and variance of the test output $y_{*,i}$ at $\mathbf{x}_{*,i}$.

The SMSE($f$), SMSE($g$) and NLPD($y$) are three quantities to measure regression losses when a trivial GP model is preferred. Lower losses indicate better predictive performance. In the next section, the predictive performance of the five GPs on the four benchmark experiments will be evaluated in detail through the following criteria: the SMSE($f$) loss on recovering the unknown function; the SMSE($g$) loss on recovering the noise standard deviation if available; and the NLPD($y$) loss on predicting the future observation.

### *C. Results*

For each benchmark experiment, the five GPs are successively applied on the 100 training datasets to recover the unknown function and the input-dependent noise. The hyperparameters associated with the five GPs are all determined by a conjugate gradient optimizer. As a local search strategy, the gradient-based optimizer may yield a local optimum for the hyperparameters. To reduce the risk of getting trapped in local minima, a multi-starting point strategy [32] is adopted in conjunction with the conjugate gradient optimizer for hyperparameter estimation. New emerging nature-inspired metaheuristic algorithms such as cuckoo search [33] and bat algorithm [34] would be more promising in searching global optimum solution of the hyperparameters.

For the two synthetic benchmark experiments, the average function values, the noise standard deviations, the SMSE($f$) losses on recovering the noise-free function, the SMSE($g$) losses on recovering the noise standard deviation, and the NLPD($y$) losses on predicting the future observations are obtained, respectively, as shown in Figs. 3 and 4 for U1 and Figs. 5 and 6 for U2. It can be observed that standard GP and HGPs exhibit very similar performance on recovering the function values. The average function values recovered by the five GPs are almost identical, which are all very close to the true function values as Figs. 3(a) and 5(a) show. The SMSE($f$) losses in regard to function recovery from the five GPs are nearly at the same level, with similar medians and variabilities (boxplot widths) as shown in Figs. 4(a) and 6(a).



(a) Average function values

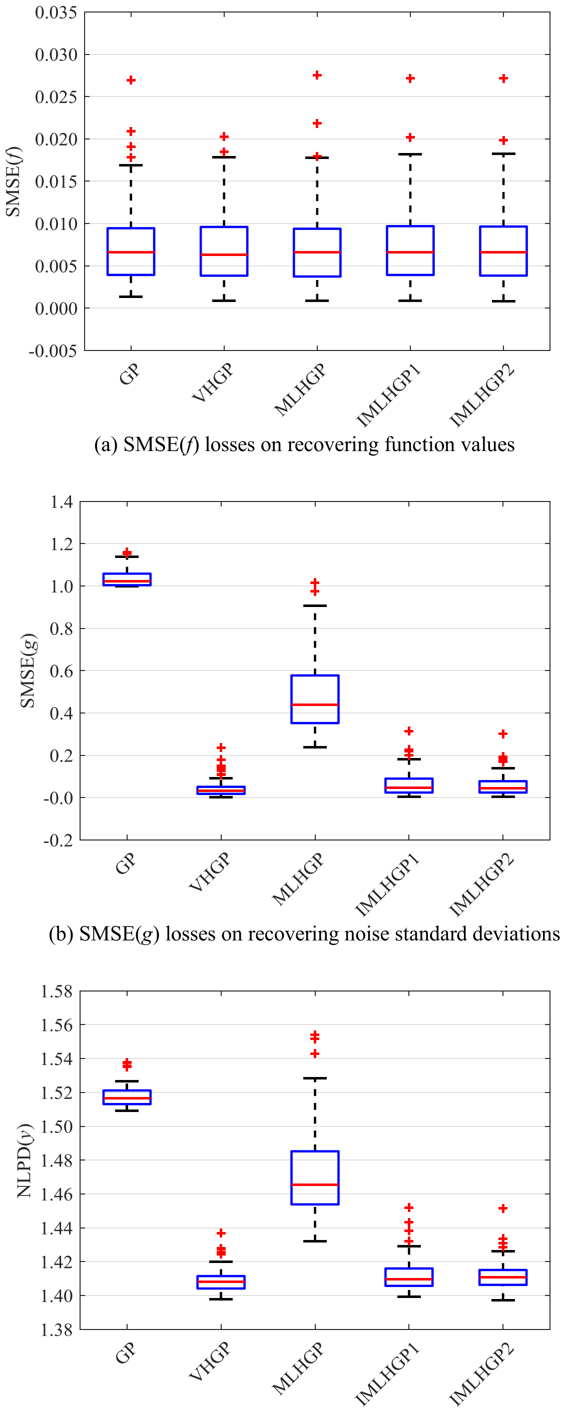
(b) Average noise standard deviations

Fig. 3. Average function values and noise standard deviations over 100 random trials by the five GPs in U1.

By contrast, the examined HGPs significantly outperform the standard GP on recovering the noise level and on predicting future observations. As Figs. 3(b) and 5(b) show, the standard GP tends to overestimate weaker noise but to underestimate stronger noise in the two benchmark experiments. The improper assumption is released in HGPs, giving rise to better predictive performance on recovering the noise level and on predicting the future observations as shown in Figs. 4(b–c) and 6(b–c).

The MLHGP outperforms the standard GP on recovering the noise level and on predicting the future observations, whereas its performance is very variable, resulting in larger and more deconcentrated SMSE($g$) and NLPD($y$) losses than other HGPs. In one training dataset case, the current MLHGP is likely to perform even worse than the standard GP, giving outlier SMSE($g$) and NLPD($y$) losses larger than those from the standard GP. The MLHGP tends to underestimate the overall noise level, and such observation was also made by other researchers [18].

The improved MLHGPs, including IMLHGP1 $(v = 1)$ and IMLHGP2 $(v = 2)$ clearly outperform the standard GP and MLHGP on recovering the noise level and on predicting future observations. They give the average noise standard deviations that are much closer to the true noise level. Moreover, their

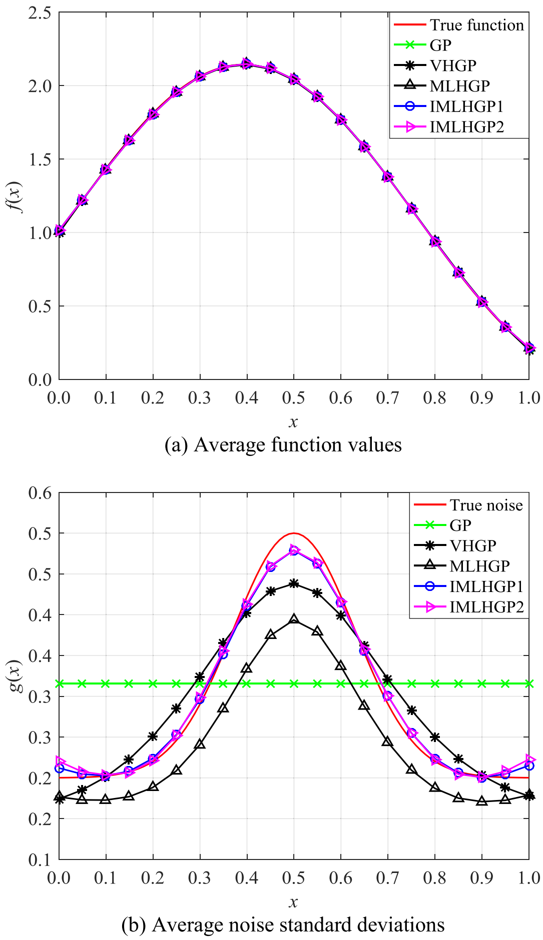


(a) SMSE($f$) losses on recovering function values

(b) SMSE($g$) losses on recovering noise standard deviations

(c) NLPD($y$) losses on predicting future observations

Fig. 4. Performance test results of the five GPs in U1 by running 100 random trials.



(a) Average function values

(b) Average noise standard deviations

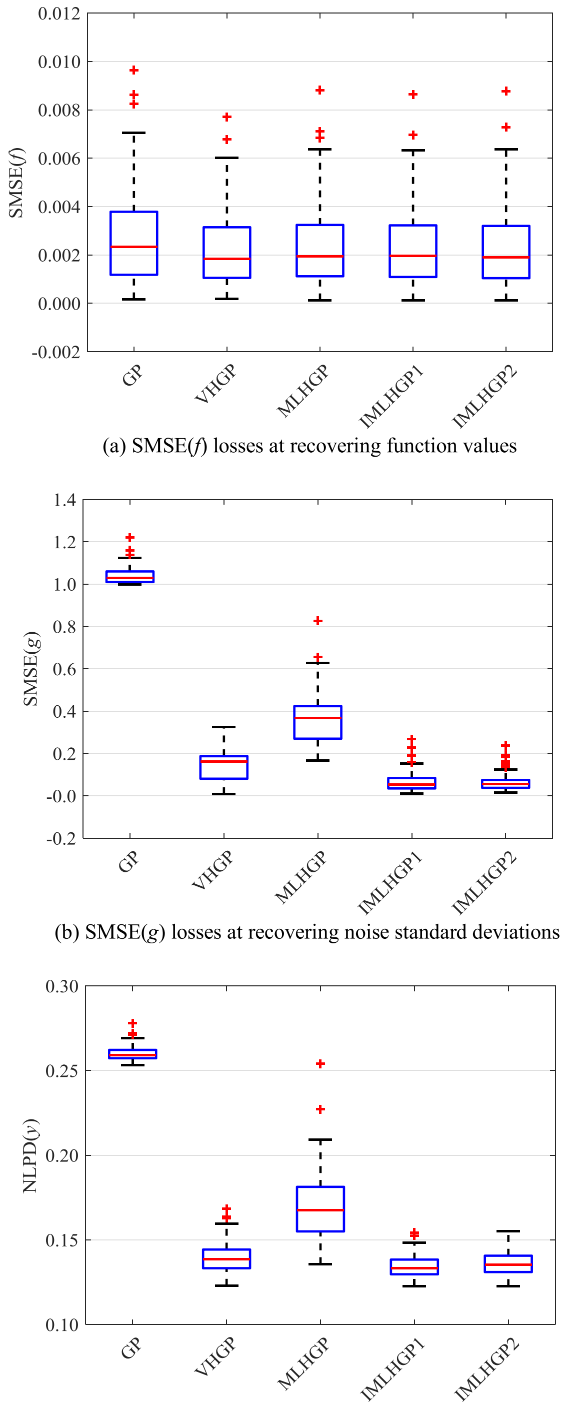
Fig. 5. Average function values and noise standard deviations over 100 random trials by the five GPs in U2.

SMSE(g) and NLPD(y) losses are much smaller and more concentrated than MLHGP. The improved MLHGPs perform better than the standard GP and MLHGP in both synthetic benchmark experiments. Nevertheless, they do not necessarily outperform VHGP. For example, VHGP exhibits the best performance on recovering the noise level and on predicting the future observations in the first synthetic experiment, while the improved MLHGPs outperform VHGP only in the second synthetic experiment.

For the benchmark experiments with real-world datasets, only the SMSE($y$) losses on recovering the noisy function and the NLPD($y$) losses on predicting future observations are available, as shown in Fig. 7 for the motorcycle dataset and Fig. 8 for the lidar dataset. It is observed that the standard GP and HGPs exhibit similar performance on recovering the function values even when the datasets come from the real world. The SMSE($y$) losses regarding the noisy function recovery obtained from the five GPs are close to each other, with similar medians and variabilities as shown in Figs. 7(a) and 8(a).

The examined HGPs again noticeably outperform the standard GP on predicting future observations, as shown in Figs. 7(b) and 8(b). The NLPD($y$) losses from the four HGPs are much smaller than those from the standard GP. Even though the motorcycle and lidar datasets are not massive, it is still able to observe that the improved MLHGPs outperform the current version on predicting future observations and their NLPD($y$) losses are even comparable to those from VHGP.

Table III provides the average training times of the five GPs on the four benchmark experiments. It is found that the training

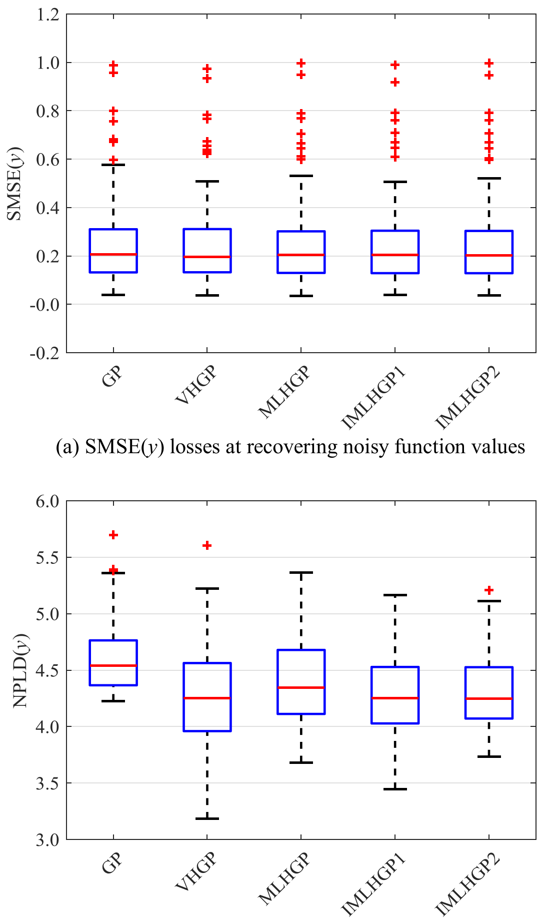


(a) SMSE(*f*) losses at recovering function values

(b) SMSE(*g*) losses at recovering noise standard deviations

(c) NLPD(*y*) losses at predicting future observations

Fig. 6. Performance test results of the five GPs in U2 by running 100 random trials.



(a) SMSE(*y*) losses at recovering noisy function values

(b) NLPD(*y*) losses at predicting future observations

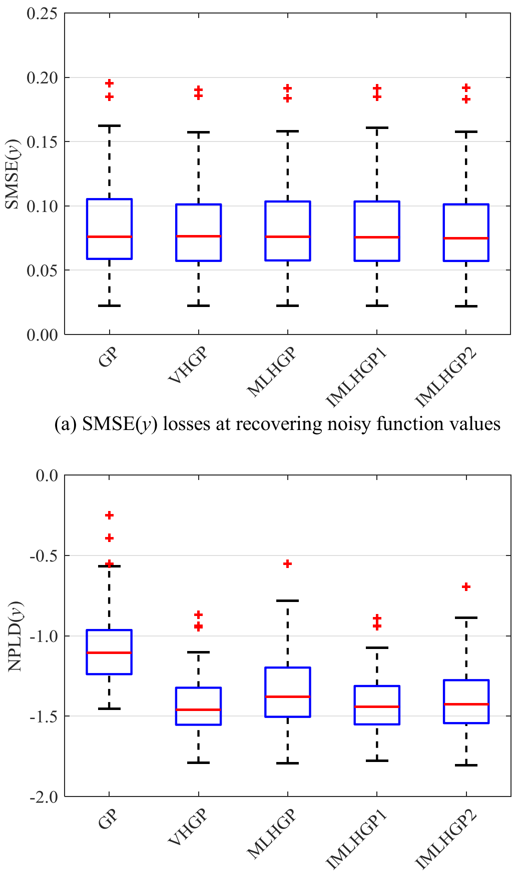
Fig. 7. Performance test results of the five GPs in U3 by running 100 random trials.

TABLE III
TRAINING TIMES OF FIVE GP MODELS FOR BENCHMARK EXPERIMENTS (IN SECONDS)

| Model | U1 | U2 | U3 | U4 |
|---|---|---|---|---|
| GP | 0.30 | 0.32 | 0.03 | 0.06 |
| VHGP | 11.67 | 11.96 | 0.92 | 1.99 |
| MLHGP | 2.23 | 2.31 | 0.23 | 0.45 |
| IMLHGP1 | 0.64 | 0.62 | 0.06 | 0.12 |
| IMLHGP2 | 0.63 | 0.65 | 0.06 | 0.13 |

The training times have been averaged over 100 random trials, running on a Dell Precision T5810, with CPU Intel Xeon E5-1620 at 3.5GHz and memory 16.0 GB.

times of the five GPs can be quite different, though their basic computational complexity all scales in the form of $\mathcal{O}(n^3)$ with $n$ being the number of training data points. Among the five GPs, the standard GP is found to be computationally the most efficient, taking the least average time to obtain a trivial regression model. The improved MLHGPs (IMLHGP1 and IMLHGP2) are also very attractive, taking about twice the time of a standard GP. The MLHGP is required to perform a sophisticated EM-like iteration learning procedure such that it costs more than seven times the training time of a standard GP on the four benchmark experiments. The VHGP is computationally the most expensive among the five GPs, costing at least thirty times the training effort of a standard GP. This can be attributed to the fact that there are numerous unknown hyperparameters to be learned, including not only model hyperparameters but also many variational parameters.



(a) SMSE($y$) losses at recovering noisy function values

(b) NLPD($y$) losses at predicting future observations

Fig. 8. Performance test results of the five GPs in U4 by running 100 random trials.

## V. Conclusion

In this paper, an improved MLHGP algorithm is proposed to deal with a kind of nonlinear regression problems with input-dependent noise. The improved model follows the same idea in the current MLHGP that adopts a point estimate to replace the full noise posterior distribution. The improved MLHGP, however, affords an approximately unbiased estimate for the most likely noise level, differing from the biased estimate in the current model. The approximately unbiased noise estimate is elicited by the method of moments for Bayesian residuals. This refinement brings about a significant improvement in the noise estimate and exempts the EM-like learning from the current MLHGP.

To validate the feasibility and effectiveness of the improved MLHGP, we have compared its performance with the standard GP, the VHGP and the current MLHGP by addressing two synthetic and two real-world benchmark experiments, in terms of regression losses on function and noise recoveries, future prediction and computational cost. The experiment results show that the improved MLHGP clearly outperforms the current MLHGP in algorithmic accuracy, stability and computational cost. Though the improved MLHGP may not necessarily outperform the variational approach, it is much simpler in implementation and more computationally efficient.

While the improved MLHGP algorithm is quite powerful for pursuing regression problems with input-dependent noise, the computational constraint of it remains a major hurdle to practical applications where the datasets are extremely large. In addition, there exist more challenging regression problems in practical applications, such as non-Gaussian noises [8], output constraints [35] and observation outliers [36]. It would be desirable in the future to attempt sparse approximations or non-Gaussian likelihoods in the face of these highly demanding applications.

## References

[1] C. K. Williams and C.E. Rasmussen, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[2] B. W. Silverman, “Some aspects of the spline smoothing approach to non-parametric regression curve fitting,” *J. Roy. Statistical Soc.*, vol. 47, no. 1, pp. 1–21, 1985.

[3] M. W. Sigrist, “Air monitoring by laser photoacoustic spectroscopy,” in: M. W. Sigrist (Ed.), *Air Monitoring by Spectroscopic Techniques, Chemical Analysis Series*, vol. 127, Hoboken, NJ, USA: Wiley, pp. 163–238, 1994.

[4] D. G. Altman, “Construction of age-related reference centiles using absolute residuals,’’ *Stat. Med.*, vol. 12, no. 10, pp. 917–924, 1993.

[5] J. M. Bland, “The half-normal distribution method for measurement error: two case studies,” Dept. Health Sciences, York Univ., Technical Report, 2005. [Online]. Available: https://www-users.york.ac.uk/∼mb55/talks/halfnor.pdf

[6] H. Shi, K. Worden, and E.J. Cross, “A cointegration approach for heteroscedastic data based on a time series decomposition: an application to structural health monitoring,” *Mech. Syst. Signal Process.*, vol. 120, pp. 16–31, 2019.

[7] P. W. Goldberg, C. K. Williams, and C. M. Bishop, “Regression with input-dependent noise: A Gaussian process treatment,” in *Advances in Neural Information Processing Systems*, vol. 10, M. I. Jordan, M. J. Kearns, and S. A. Solla, eds., Hoboken, NJ, USA: MIT Press, 1998, pp. 493–499.

[8] C. Y. Wang, “Gaussian process regression with heteroscedastic residuals and fast MCMC methods,” Ph.D. dissertation, Dept. Statistics, Toronto Univ., Toronto, Canada, 2014.

[9] L. Muñoz-González, M. Lázaro-Gredilla, and A. R. Figueiras-Vidal, “Heteroscedastic Gaussian process regression using expectation propagation,” in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process.*, Santander, Spain, 2011, pp. 1–6.

[10] L. Muñoz-González, M. Lázaro-Gredilla, and A.R. Figueiras-Vidal, “Divisive Gaussian processes for nonstationary regression,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 11, pp. 1991–2003, Nov. 2014.

[11] V. Tolvanen, P. Jylänki, and A. Vehtari, “Expectation propagation for nonstationary heteroscedastic Gaussian process regression,” in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process.*, Reims, France, 2014, pp. 1–6.

[12] L. Muñoz-González, M. Lázaro-Gredilla, and A. R. Figueiras-Vidal, “Laplace approximation for divisive gaussian processes for nonstationary regression,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 3, pp. 618–624, Mar. 2016.

[13] M. Lázaro-Gredilla and M. K. Titsias, “Variational heteroscedastic Gaussian process regression,” in *Proc. 28th Int. Conf. Mach. Learn.*, Bellevue, WA, USA, 2011, pp. 841–848.

[14] K. Kersting, C. Plagemann, P. Pfaff, and W. Burgard, “Most likely heteroscedastic Gaussian process regression,” in *Proc. 24th Int. Conf. Mach. Learn.*, New York, NY, USA, 2007, pp. 393–400.

[15] N. Quadrianto, K. Kersting, M. D. Reid, T. S. Caetano, and W. L. Buntine, “Kernel conditional quantile estimation via reduction revisited,” in *Proc. 9th IEEE Int. Conf. Data Mining*, Miami, FL, USA, 2009, pp. 938–943.

[16] Q. V. Le, A. J. Smola, and S. Canu, “Heteroscedastic gaussian process regression,” in *Proc. 22nd Int. Conf. Mach. Learn.*, Bonn, Germany, 2005, pp. 489–496.

[17] S. Urban, M. Ludersdorfer, and P. Van Der Smagt, “Sensor calibration and hysteresis compensation with heteroscedastic gaussian processes,” *IEEE Sensors J.*, vol. 15, no. 11, pp. 6498–6506, 2015.

[18] M. Binois, R. B. Gramacy, and M. Ludkovski, “Practical heteroskedastic Gaussian process modeling for large simulation experiments,” *J. Comput. Graph. Stat.*, vol. 27, no. 4, pp. 808–821, 2018.

[19] A. J. McHutchon, “Nonlinear modelling and control using Gaussian processes,” Ph.D. dissertation, Dept. Eng., Cambridge Univ., Cambridge, UK, 2014.

[20] O. Menzer, A. M. Moffat, W. Meiring, G. Lasslop, E. G. Schukat-Talamazzini, and M. Reichstein, “Random errors in carbon and water vapor fluxes assessed with Gaussian Processes,” *Agricultural Forest Meteorol.*, vol. 178, pp. 161–172, 2013.

[21] A. Boukouvalas, D. Cornford, and M. Stehlík, “Optimal design for correlated processes with input-dependent noise,” *Comput. Statist. Data Anal.*, vol. 71, pp. 1088–1102, 2014.

[22] M. Binois, R. B. Gramacy, and M. Ludkovski, “Practical heteroscedastic gaussian process modeling for large simulation experiments,” *J. Comput. Graphical Statist.*, vol. 27, no. 4, 808–821, 2018.

[23] M. Binois, J. Huang, R. B. Gramacy, and M. Ludkovski, “Replication or exploration? Sequential design for stochastic simulation experiments,” *Technometrics*, vol. 61, no. 1, pp. 7–23, 2019.

[24] M. Chung *et al.*, “Parameter and uncertainty estimation for dynamical systems using surrogate stochastic processes,” *SIAM J. Scientific Comput.*, vol. 41, no. 4, pp. A2212–A2238, 2019.

[25] S. Zhu, X. Luo, X. Yuan, and Z. Xu, “An improved long short-term memory network for streamflow forecasting in the upper Yangtze River,” *Stochastic Environmental Research and Risk Assessment*, 2020. [Online]. Available: https://doi.org/10.1007/s00477-020-01766-4

[26] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed., New York; Berlin, Germany: Springer, 2006.

[27] B. P. Carlin and T. A. Louis, *Bayes and Empirical Bayes methods for Data Analysis*. London, U.K.: Chapman and Hall/CRC, 2010.

[28] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D.B. Rubin, *Bayesian Data Analysis*. London, U.K.: Chapman and Hall/CRC, 2013.

[29] J. O. Rawlings, S. G. Pantula, and D. A. Dickey, *Applied Regression Analysis: a Research Tool*. New York; Berlin, Germany: Springer Science and Business Media, 2001.

[30] R. V. Hogg, E. A. Tanis, and D. L. Zimmerman, *Probability and statistical inference*, 8th ed., NJ, USA: Pearson/Prentice Hall, 2010.

[31] A. Winkelbauer, “Moments and absolute moments of the normal distribution,” 2014. [Online]. Available: http://arxiv.org/pdf/1209.4340.pdf

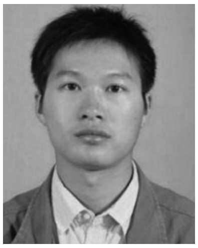
[32] H. P. Wan and Y. Q. Ni, “Bayesian multi-task learning methodology for reconstruction of structural health monitoring data,” *Structural Health Monit.*, vol. 18, no. 4, pp. 1282–1309, 2019.

[33] V. Stojanovic, N. Nedic, D. Prsic, L. Dubonjic, and V. Djordjevic, “Application of cuckoo search algorithm to constrained control problem of a parallel robot platform,” *Int. J. Adv. Manuf. Technol.*, vol. 87, pp. 2497–2507, 2016.

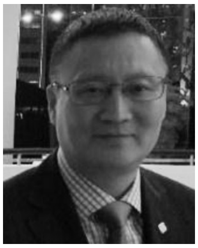
[34] V. Stojanovic and N. Nedic, “A nature inspired parameter tuning approach to cascade control for hydraulically driven parallel robot platform,” *J. Optim. Theory Appl.*, vol. 168, no. 1, pp. 332–347, 2016.

[35] V. Stojanovic and N. Nedic, “Robust identification of OE model with constrained output using optimal input design,” *J. Franklin Inst.*, vol. 353, no. 2, pp. 576–593, 2016.

[36] V. Stojanovic and N. Nedic, “Joint state and parameter robust estimation of stochastic nonlinear systems,” *Int. J. Robust Nonlinear Control*, vol. 26, no. 14, pp. 3058–3074, 2016.



**Qiu-Hu Zhang** received the B.Eng. and M.Sc. degrees from the Hefei University of Technology, Hefei, China, in 2009 and 2014, respectively. He is currently working toward the Ph.D. degree with The Hong Kong Polytechnic University, Hong Kong. His research interests include Bayesian machine learning, Bayesian decision theory, and structural health monitoring.



**Yi-Qing Ni** received the B.Eng. and M.Sc. degrees from Zhejiang University, Hangzhou, China, in 1983 and 1986, respectively, and the Ph.D. degree from The Hong Kong Polytechnic University, Hong Kong, in 1997. He is currently a Chair Professor of Smart Structures and Rail Transit, Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University. He is the Director of National Engineering Research Center on Rail Transit Electrification and Automation (Hong Kong Branch) and a Vice President of International Society for Structural Health Monitoring of Intelligent Infrastructure (ISHMII). His research areas cover structural health monitoring, sensors and actuators, signal processing, and smart materials and structures.