Efficient Bayesian Optimization for Target Vector Estimation

Anders Kirk Uhrenholt
University of Glasgow,
School of Computing Science,
Glasgow, Scotland
anders.uhrenholt@glasgow.ac.uk

Bjørn Sand Jensen
University of Glasgow,
School of Computing Science,
Glasgow, Scotland
bjorn.jensen@glasgow.ac.uk

Abstract

We consider the problem of estimating a target vector by querying an unknown multi-output function which is stochastic and expensive to evaluate. Through sequential experimental design the aim is to minimize the squared Euclidean distance between the output of the function and the target vector. Applying standard single-objective Bayesian optimization to this problem is both wasteful, since individual output components are never observed, and imprecise since the predictive distribution for new inputs will be symmetric and have support in the negative domain. We address this issue by proposing a Gaussian process model that takes into account the individual function outputs and derive a distribution over the resulting 2-norm. Furthermore, we derive computationally efficient acquisition functions and evaluate the resulting optimization framework on several synthetic benchmark functions and a real-world problem. The results demonstrate a significant improvement over standard Bayesian optimization methods based on both standard and Warped Gaussian processes.

1 Introduction

Estimating a target vector by querying a multi-output function is a challenging problem when the function is stochastic, expensive to evaluate, and only accessible via its inputs and outputs. A popular methodology for optimizing single-output functions under such constraints is Bayesian optimization which offers a principled approach based on probabilistic modelling [1, 2, 3]. By inferring a predictive distribution over new evaluations the decision as to which input to sample next may be informed by probabilistic measures such as expected improvement over the incumbent value or reduction in posterior entropy. More recently this framework has been extended to multi-output problems where the aim is to identify the set of Pareto-optimal (or non-inferior) output vectors [4, 5, 6, 7, 8, 9].

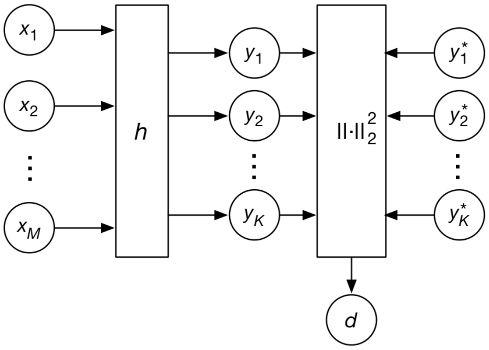

Figure 1: Schematic illustration of the target vector estimation setup. We aim to identify an input $\mathbf{x} = [x_1, \cdots, x_M]^T$ such that the quadratic Euclidean distance between our target vector $\mathbf{y}^* = [y_1^*, \cdots, y_K^*]^T$ and the output of the unknown function h is minimized.

A related problem, which has received less attention within the Bayesian optimization framework, is that of optimizing a multi-output system through an aggregating, single-output objective function such as the 2-norm. In contrast to the Pareto setting we are here willing to sacrifice the deterioration of one output dimension if it yields sufficient improvement for another. This aggregation, however, is problematic for the assumption of Gaussianity underpinning Bayesian optimization. We address this problem in the context

Proceedings of the 22^{nd} International Conference on Artificial Intelligence and Statistics (AISTATS) 2019, Naha, Okinawa, Japan. PMLR: Volume 89. Copyright 2019 by the author(s).

of target vector estimation [10, 11] where the discrepancy between output and target is measured through the sum of squares. This is a class of problems analogous to classical system identification and model inversion where we aim to estimate the parameters, settings or, in a broader sense, a target vector determining the characteristics of the unknown function (or system) under consideration. Examples include the modelling of blood flow in the human vasculature [12], estimation of the optimal hyper-parameters of machine learning models [2], defining the structural design of helicopter rotor blades [10], and estimating emission line intensities of alumina powder [11].

The basic setup is illustrated in Figure 1 and defined as follows: Given a multi-output function $h : \mathcal{X} \rightarrow \mathcal{Y}$ for $\mathcal{X} \subset \mathbb{R}^M, \mathcal{Y} \subset \mathbb{R}^K$, and target vector $\mathbf{y}^*$ we are interested in identifying a $\hat{\mathbf{x}}$ which minimizes $\|h(\hat{\mathbf{x}}) - \mathbf{y}^*\|^2$. In the Bayesian optimization framework this problem has traditionally been solved by directly observing the the distance (or cost) and minimizing it through standard, single-objective Bayesian optimization [12]. This strategy, however, suffers from two major caveats: i) the model is only ever presented with measurements of the aggregated distance, thus never seeing the individual output components of h which could potentially be informative w.r.t. the optimization, and ii) the predictive distribution is known to be imprecise at critical input locations due to symmetry and support in the negative domain.

Given these deficiencies we suggest that a far better noise model may be derived by observing the individual outputs of h and using these to construct a distribution over the norm. This will, in turn, increase the predictive capacity of the model thereby improving the optimization as a whole. Specifically, we contribute with: i) a Gaussian process modelling approach for target vector estimation in which we model the individual outputs of h and derive an approximate noncentral Chi-squared distribution over the objective $\|h(\mathbf{x}) - \mathbf{y}^*\|_2^2$, and ii) the derivation of standard acquisition functions (Expected Improvement and Lower Confidence Bound [2]) having computational complexity equivalent to those in standard Bayesian optimization.[1]

Finally we provide an empirical evaluation of the proposed models and associated acquisition functions by comparing against single-objective Bayesian optimization approaches reliant on two well-known surrogate models, namely Gaussian processes and warped Gaussian processes [14]. The empirical results show a significant improvement for both synthetic benchmark functions and a practical problem when applying the proposed modelling approach.

The remainder of the paper is structured as follows: In Section 2 we outline the theory behind Bayesian optimization with Gaussian process surrogate models. Section 3 describes the proposed optimization scheme for minimizing the squared Euclidean distance. The effectiveness of this scheme is demonstrated empirically in Section 4 for a suite of benchmark functions and a practical problem.

2 Background

2.1 Bayesian optimization

Consider the task of minimizing a smooth, stochastic cost function $h : \mathbb{R}^M \rightarrow \mathbb{R}$ which is blackboxed and expensive to evaluate.[2] Through noisy measurements we are to arrive at a minimal output value in as few iterations as possible. We thus need to trade off learning the response surface of h in order to make an informed choice about where to sample next (exploration) against using our limited sampling budget to locate the optimal input in regions with low cost (exploitation).

In Bayesian optimization we assume the measurements to be evaluations of a latent function with added noise:

$$d = f(\mathbf{x}) + \epsilon,$$

where ϵ follows a known distribution. By placing a suitable prior on f we can infer a predictive distribution for unseen inputs conditioned on all samples collected so far:

$$p(d \mid \mathbf{x}, \mathcal{D}),$$

where $\mathcal{D}$ is the set of previously observed inputs and outputs. In line with [3] we refer to the statistical model, through which the predictive distribution is inferred, as the surrogate model. We can now select the next sample via an acquisition function $\alpha : \mathbb{R}^M \rightarrow \mathbb{R}$ that assigns utility to unseen input, $\mathbf{x}$, based on the associated predictive distribution over d. While the design of acquisition functions is a rich and fast evolving research field we will restrict attention to two of the most popular variants: Expected Improvement (EI) and Lower Confidence Bound (LCB) [2]. In EI we consider how much the current incumbent value, $d_{\min}$, can be expected to improve by picking a given input:

$$\alpha_{\mathrm{EI}}(\mathbf{x}; \mathcal{D}) = \mathbb{E}_{p(d|\mathbf{x},\mathcal{D})}[\min(0, d_{\min} - d)]. \quad (1)$$

[1] An open-source implementation of the methods extending the Emukit Bayesian optimization framework [13] is available at: `https://github.com/akuhren/target_vector_estimation`

[2] All presented theory can equally be framed as a task where a reward function is maximized.

LCB, on the other hand, uses the minimum of a predetermined confidence interval for d when assigning utility to $\mathbf{x}$. It can be viewed as always considering the best possible outcome within that interval [3]. In the most general formulation of LCB it is defined through the predictive distribution's quantile function $F^{-1}(d \mid \mathbf{x}, \mathcal{D}; q)$ yielding the q'th quantile for the distribution of d:

$$\alpha_{\mathrm{LCB}}(\mathbf{x}; q, \mathcal{D}) = -F^{-1}(d \mid \mathbf{x}, \mathcal{D}; q), \tag{2}$$

where the quantile q determines how optimistic (or exploratory) a search strategy to deploy. Under a Gaussian distribution the acquisition function takes the more well-known form

$$\alpha_{\mathrm{LCB}}(\mathbf{x}; \beta, \mathcal{D}) = -\mu(\mathbf{x}) + \beta\sigma(\mathbf{x}),$$

where $\mu(\mathbf{x})$ and $\sigma(\mathbf{x})$ are the mean and standard deviation of the predictive distribution for d, and $\beta = -\Phi^{-1}(q)$ is defined through the standard normal quantile function.

2.2 Gaussian processes

The most commonly used surrogate model, and the one used in the experiments carried out in this paper, is the Gaussian process (GP) model [15]. The GP assumption implies that any finite subset of function evaluations will follow a multivariate normal distribution:

$$f(\mathbf{X}) = \mathbf{f} \mid \mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}),$$

where $\boldsymbol{\mu}$ and $\mathbf{K}$ are defined through appropriate mean function $m : \mathbb{R}^M \to \mathbb{R}$ and covariance function $\kappa : \mathbb{R}^M \times \mathbb{R}^M \to \mathbb{R}$. Assuming that our observations $\mathbf{d}$ have isotropic Gaussian likelihood with a GP prior on the mean we have

$$\mathbf{d} \mid \mathbf{f}, \sigma^2 \sim \mathcal{N}(\mathbf{f}, \sigma^2\mathbf{I}), \tag{3}$$

and by the marginalization properties of the normal distribution it follows that

$$\mathbf{d} \mid \mathbf{X}, \sigma^2 \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K} + \sigma^2\mathbf{I}).$$

Since the output d' for an unseen test point $\mathbf{x}'$ is also assumed to be generated from the Gaussian process, it will be jointly Gaussian with the already observed variables. This yields a closed-form expression for the predictive distribution of d' when conditioned on our N observations $\mathcal{D}$:

$$\begin{aligned} p(d' \mid \mathbf{x}', \mathcal{D}) &= \mathcal{N}(d' \mid \mu(\mathbf{x}'), \sigma^2(\mathbf{x}')), \\ \mu(\mathbf{x}') &= m(\mathbf{x}') + \mathbf{k}^T(\mathbf{K} + \sigma^2\mathbf{I})^{-1}(\mathbf{d} - \boldsymbol{\mu}), \\ \sigma^2(\mathbf{x}') &= \kappa(\mathbf{x}', \mathbf{x}') - \mathbf{k}^T(\mathbf{K} + \sigma^2\mathbf{I})^{-1}\mathbf{k}, \end{aligned}$$

where $\mathbf{k} = [\kappa(\mathbf{x}^{(1)}, \mathbf{x}'), \cdots, \kappa(\mathbf{x}^{(N)}, \mathbf{x}')]^T$.

2.3 Warped Gaussian processes

The generic GP outlined above assumes a Gaussian likelihood for our observations. However, when the observations are produced through the squared 2-norm this assumption is clearly erroneous, as previously argued. A common approach to achieve a non-Gaussian, yet analytically tractable, predictive distribution is through the use of warped GP's [14]. Here the observations are first warped to a latent space before applying the (noise free) GP. Following [14], we define

$$z = g(d), \qquad p_z(z \mid \mathbf{x}, f, \sigma^2) = \mathcal{N}(z \mid f(\mathbf{x}), \sigma^2),$$

such that the conditional distribution for d is given by

$$p_d(d \mid \mathbf{x}, f) = p_z(g(d) \mid \mathbf{x}, f) \cdot \left| \frac{\partial g(d)}{\partial d} \right|,$$

for the monotonic, nonlinear function g. This allows us to model non-Gaussian distributions within the established GP framework.

3 Bayesian Optimization of the 2-norm

3.1 Modelling the 2-norm

We define the stochastic variable $d \mid \mathbf{x} = \|h(\mathbf{x}) - \mathbf{y}^*\|_2^2$ with the aim of modelling the true distribution $p(d \mid \mathbf{x})$ and using this in the Bayesian optimization scheme to minimize d. In a standard setting reliant on a GP surrogate model we would simply assume $p_{\mathcal{N}}(d \mid \mathbf{x}, \mathcal{D}) = \mathcal{N}(d \mid \mu(\mathbf{x}), \sigma^2(\mathbf{x}))$ as outlined in the previous section.

However, this is imprecise since we know the true $p(d \mid \mathbf{x})$ to be asymmetric and without negative support. Furthermore, we are discarding potentially important information about the individual outputs of h by aggregating the squared differences to $\mathbf{y}^*$ before presenting the observation to the surrogate model.

Instead assume that the individual outputs of the multi-output function h can be reasonably modelled by a set of uncorrelated functions drawn from one or several GP priors. That is, for a given input, $\mathbf{x}$, we obtain a predictive distribution for each of the K outputs:

$$p(y_k \mid \mathbf{x}, \mathcal{D}) = \mathcal{N}(y_k \mid \mu_k(\mathbf{x}), \sigma_k^2(\mathbf{x})), \;\; 1 \leq k \leq K.$$

An unbiased approximation of the distribution over the squared 2-norm d is then given by

$$p_{\chi^2}(d \mid \mathbf{x}, \mathcal{D}) \approx NC\chi^2(d\gamma^{-2} \mid K, \lambda)\gamma^{-2}, \tag{4}$$

$$\lambda = \gamma^{-2} \sum_{k=1}^{K} (\mu_k(\mathbf{x}) - y_k^*)^2,$$

$$\gamma = \sqrt{\frac{1}{K} \sum_{k=1}^{K} \sigma_k^2(\mathbf{x})},$$

where $NC\chi^2$ is the noncentral Chi-squared distribution which is governed by the number of outputs K (commonly denoted *degrees of freedom*) and noncentrality parameter λ. Refer to Supplementary Material A for the derivation of Eq. 4. To incorporate this distribution in the acquisition functions we need an expression for its cumulative distribution function (CDF) which we will denote $F_{K,\lambda}$. For this purpose we will use the results of [16] in which it is shown that given a random variable t with distribution $p(t) = NC\chi^2(t \mid K, \lambda)$, we can transform it into an approximately normally distributed variable as follows:

$$z = \left(\frac{t}{K + \lambda}\right)^{\ell},$$

$$\ell = 1 - \frac{r_1 r_3}{3 r_2^2}, \qquad r_s = 2^{s-1}(s-1)!(K + s\lambda),$$

yielding $p(z \mid K, \lambda) \approx \mathcal{N}(z \mid \alpha, \rho^2)$ with

$$\alpha = 1 + \ell(\ell - 1)\left(\frac{r_2}{2r_1^2} - (2-\ell)(1-3\ell)\frac{r_2^2}{8r_1^4}\right),$$

$$\rho = \frac{\ell r_2^2}{r_1}\left(1 - \frac{(1-\ell)(1-3\ell)}{4r_1^2} r_2\right).$$

Through this approximation we obtain a closed-form approximation of the CDF of t by

$$F_{K,\lambda}(t) \approx \Phi\left(\frac{z - \alpha}{\rho}\right),$$

where Φ is the standard normal CDF. As such, we obtain the following expression for the CDF of d:

$$F(d \mid \mathbf{x}, \mathcal{D}) \approx F_{K,\lambda}(d\gamma^{-2}). \tag{5}$$

This expression is furthermore invertible, allowing us to extract an approximate quantile function $F^{-1}(d \mid \mathbf{x}, \mathcal{D}; q)$ which we will use when defining the LCB acquisition function.

Comparing the standard GP and the proposed 2-norm model, we see that the latter seeks to explain the behaviour of each individual output dimension of h and through these derive a distribution closer aligned with what we would expect for $p(d \mid \mathbf{x})$. To illustrate the difference between the models, refer to Figure 2 where the predictive distribution for $p_{\mathcal{N}}$ and p_{χ^2} are compared for the one-dimensional function $h(x) = -3x\sin(3x/4) + e^{-2x} + \epsilon$, $\epsilon \sim \mathcal{N}(0, 0.5)$ with target $y^* = -4$. The mean of $d \mid x = \|h(x) - y^*\|_2^2$ has two minima in the domain $[0, 2\pi]$ which the surrogate model is to identify. Figure 2 (c) shows the predictive distribution over $p_{\mathcal{N}}(d \mid x, \mathcal{D})$ for a GP that has been fitted directly to the noisy observations of d for 6 equidistant training points. In Figure 2 (d) a GP has been fitted to the observations of $y = h(x)$ and the predictive distribution for $p_{\chi^2}(d \mid x, \mathcal{D})$ has been inferred by Eq. (4). While both models have found a good estimation of the median, $p_{\mathcal{N}}$ is symmetrical and yields negative support at low values reflecting a flawed expectation about the behavior of d in unexplored regions.

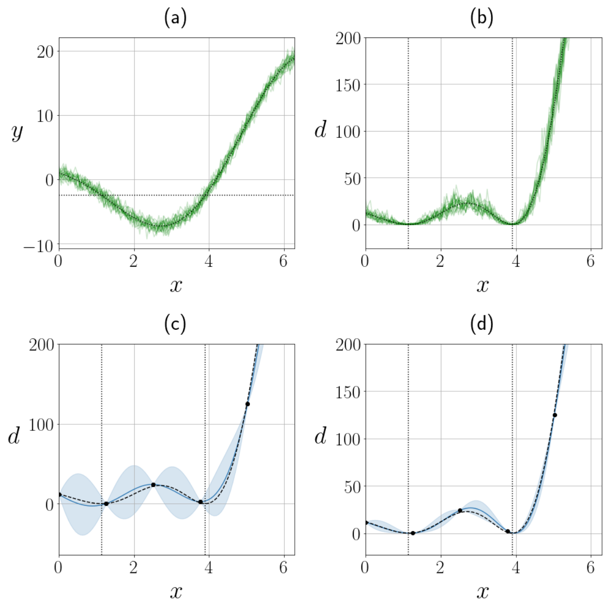

Figure 2: Comparison of the predictive distributions. (a) 20 samples of a single-output stochastic function $h(x)$ with target $y^* = -4$. (b) 20 samples drawn from the 2-norm $d \mid x = \|h(x) - y^*\|_2^2$ with the two minima marked with vertical, dotted lines. (c)-(d) The predictive distribution for respectively $p_{\mathcal{N}}(d \mid x, \mathcal{D})$ and $p_{\chi^2}(d \mid x, \mathcal{D})$.

The benefit of incorporating information about the individual outputs for the multi-objective case is illustrated in Figure 3 where we consider the linear multi-output function $h(x) = [-2\pi x, x]^T + \epsilon, \epsilon \sim \mathcal{N}(\mathbf{0}, 0.2\mathbf{I})$ with target $\mathbf{y}^* = [\pi, \pi]^T$. The quadratic 2-norm $d \mid \mathbf{x} = \|h(\mathbf{x}) - \mathbf{y}^*\|_2^2$ yields a convex function with heteroscedastic noise since the variance increases away from the minimum. In Figure 3 (d) a GP has been fitted directly to 6 equidistant observations of d in the domain $[0, 2\pi]$, but due to the high noise the model has misinterpreted the quadratic trend. In Figure 3 (e) two GP's have been fitted to the observations of $[y_1, y_2]^T = h(x)$ for the same inputs. Since each output is linear with homoscedastic noise it is consid-

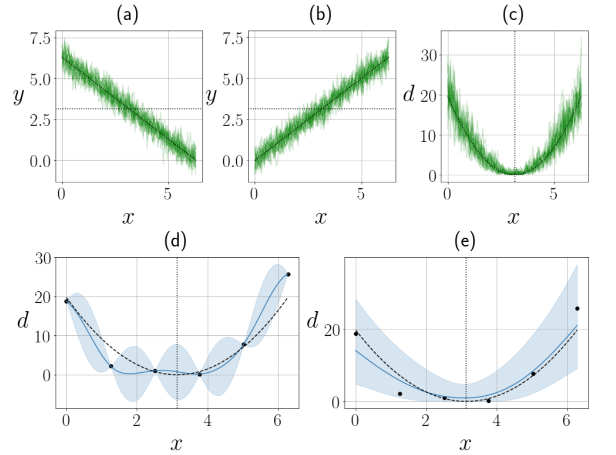

Figure 3: Comparison of the predictive distributions for a multi-output function. (a)-(b) 20 samples of a two-dimensional stochastic function $h(x)$ with target $\mathbf{y}^* = [\pi, \pi]$. (c) 20 Samples drawn from the 2-norm $d \mid x = \|h(x) - \mathbf{y}^*\|_2^2$ with the global minimum marked with a vertical, dotted line. (d)-(e) The predictive distributions for $p_{\mathcal{N}}(d \mid x, \mathcal{D})$ and $p_{\chi^2}(d \mid x, \mathcal{D})$.

erably easier for the GP's to find a proper fit and so the inferred distribution $p_{\chi^2}(d \mid x, \mathcal{D})$ yields a more robust estimate.

3.2 Acquisition functions

In this section we derive the new EI and LCB acquisition functions for $p_{\chi^2}(d \mid \mathbf{x}, \mathcal{D})$ for the 2-norm model proposed in Section 3.1.

Expected improvement: For EI we start from the definition given in Eq. (1) and show that by taking the expectation over the approximated $NC\chi^2$ distribution we arrive at a closed-form expression that is easy to evaluate. First define $t = d\gamma^{-2}$. We then have

$$\begin{aligned}
\alpha_{\mathrm{EI}}(\mathbf{x}; \mathcal{D}) &= \int_0^{d_{\min}} (d_{\min} - d) p_{\chi^2}(d \mid \mathbf{x}, \mathcal{D}) \, \mathrm{d}d \\
&= \int_0^{d_{\min}/\gamma^2} (d_{\min} - t\gamma^2) NC\chi^2(t \mid K, \lambda) \, \mathrm{d}t \\
&= d_{\min} F_{K,\lambda}(d_{\min}/\gamma^2) \\
&\qquad - \gamma^2 \mathbb{E}\left[t \mid t < d_{\min}/\gamma^2\right] F_{K,\lambda}(d_{\min}/\gamma^2).
\end{aligned}$$

Next we apply the results from [17] stating that for $z \sim NC_{\chi}^2(z \mid K, \lambda)$ the truncated mean is given by $\mathbb{E}[z \mid z < a] = K \cdot F_{K+2,\lambda}(a) + \lambda F_{K+4,\lambda}(a)$. This lets us arrive at the closed-form expression:

$$\begin{aligned}
\ldots = d_{\min} & F_{K,\lambda}(d_{\min}/\gamma^2) \\
& - \gamma^2 (K \cdot F_{K+2,\lambda}(d_{\min}/\gamma^2) \\
& \qquad + \lambda \cdot F_{K+4,\lambda}(d_{\min}/\gamma^2)).
\end{aligned}$$

Note that the expression from [17] is pivotal for our derivation of the EI acquisition function. The lack of an analogous expression for the generalized Chi-squared distribution is what necessitates the assumption of our observation noise being isotropic.

Lower confidence bound: For LCB we use the definition given in Eq. (2) reliant on the negative quantile function for the predictive distribution. We first invert the closed-form approximation for $F(d \mid \mathbf{x}, \mathcal{D})$ in Eq. (5) in order to obtain

$$F^{-1}(d \mid \mathbf{x}, \mathcal{D}; q) = \sqrt[\ell]{\Phi^{-1}(q)\rho + \alpha} \cdot (K + \lambda)\gamma^2,$$

where Φ^{-1} is the standard normal quantile function and ℓ, ρ, and α are as defined in Section 3.1. To match the signature of the Gaussian LCB, which uses β as exploration parameter, we can set $q = \Phi(-\beta)$ as explained in Section 2.1. As such we obtain

$$\alpha_{\mathrm{LCB}}(\mathbf{x}; \beta, \mathcal{D}) = -\sqrt[\ell]{\alpha - \beta\rho} \cdot (K + \lambda)\gamma^2.$$

The two acquisition functions benefit from being both easy to calculate and differentiable such that they can be maximized numerically.

In Figure 4 we compare the normalized EI utilities for $p_{\mathcal{N}}(d \mid x, \mathcal{D})$ and $p_{\chi^2}(d \mid x, \mathcal{D})$ for the single-output function from Figure 2. Due to the high amount of negative support for $p_{\mathcal{N}}(d \mid x, \mathcal{D})$ between low-value training points, the associated acquisition function vastly overestimates the utility in these regions. Figure 5 shows the EI utility for the multi-output function from Figure 3. Because $p_{\mathcal{N}}(d \mid x, \mathcal{D})$ has not captured the quadratic trend of the 2-norm, the associated acquisition function identifies multiple regions of interest for future sampling that will not yield any

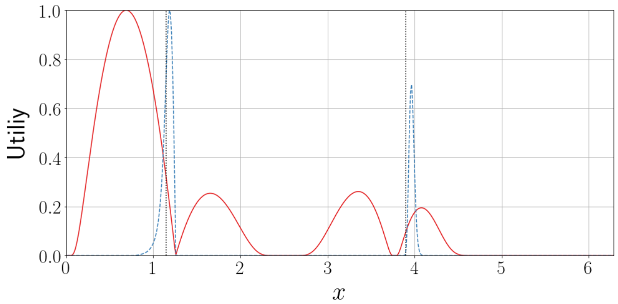

Figure 4: Comparison of the normalized acquisition utility for the one-dimensional function from Figure 2. The red, solid line is the EI acquisition for $p_{\mathcal{N}}(d \mid x, \mathcal{D})$ and the blue, stripped line is the EI acquisition for $p_{\chi^2}(d \mid x, \mathcal{D})$. The vertical, dotted lines mark the true minima.

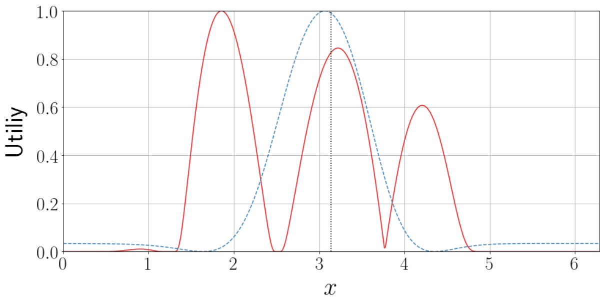

Figure 5: Comparison of the normalized acquisition utility for the multi-output function from Figure 3. The red, solid line is the EI acquisition for $p_{\mathcal{N}}(d \mid x, \mathcal{D})$ and the blue, stripped line is the EI acquisition for $p_{\chi^2}(d \mid x, \mathcal{D})$. The vertical, dotted line mark the true minimum.

improvement. For brevity we have only compared the EI utilities for each of the distributions but the same tendencies were observed for the LCB acquisition functions.

4 Experiments

The proposed 2-norm model was tested against both a standard and a warped GP model on a suite of synthetic benchmark functions as well as a real-world problem. For each surrogate model we evaluated the optimization procedure for both the EI and LCB acquisition functions, resulting in 6 distinct optimization setups (BO setups). All GP's used a Mátern 5/2 kernel with lengthscale and variance as free hyperparameters which, along with the scale of the observation noise, were fitted between iterations using evidence maximization.

The warped GP's turned out to consistently underperform so to avoid clutter we have collected the methodology and results for these BO setups in Supplementary Material B.

4.1 Synthetic functions

The 6 BO setups were tested on 9 single-objective functions from [18] and 5 multi-objective functions from [19]. In order to make the problems stochastic we added normally distributed noise on all function evaluations before returning them to the surrogate model. We scaled the noise separately for each of the K output components to avoid dimensions with smaller range being more heavily influenced by the noise addition. Specifically we used

$$\mathbf{y} = h(\mathbf{x}) + \epsilon, \qquad \epsilon \sim \mathcal{N}(\mathbf{0}, \operatorname{diag}(\mathbf{v})),$$
$$\mathbf{v} = [\Delta(h_1), \cdots, \Delta(h_K)]^T \cdot 10^{-2},$$

where $\Delta(h_k)$ is the difference between the highest and lowest value of the k'th output component from 10,000 random samples of h. The above generalizes to the single-dimensional case where $\mathbf{y}$ and $\mathbf{v}$ are scalars.

For each function h we first sampled a random target vector, $\mathbf{y}^* \in \mathbb{R}^K$, which then were to be estimated. Each of the 6 BO setups were run for 30 iterations with the same 5 initial points selected by Latin hypercube sampling. This process was repeated 8 times per objective function while keeping $\mathbf{y}^*$ fixed. Table 1 lists the means and standard deviations over the best collected points by the end of optimization for each BO setup and objective. We saw an improved performance for all objective functions when optimizing according to the proposed Chi-squared distribution. However, no consistent difference in performance was observed between EI and LCB across objective functions. As is the case for standard, single-objective Bayesian optimization the suitability of a

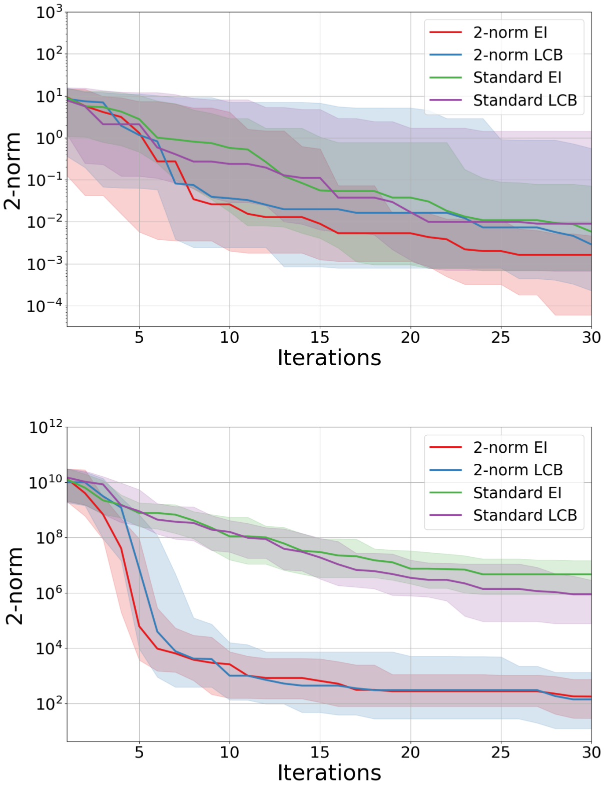

Figure 6: Optimization traces for the objective functions Ackley (top) and Bohachevsky (bottom). The traces comprise 8 runs of 30 iterations each in which the surrogate models minimize the squared Euclidean distance to a fixed target. The solid lines are the median distance to the target and the shaded regions are the interquartile ranges.

Objective		**2-norm EI**	**2-norm LCB**	**Standard EI**	**Standard LCB**
BNH	Mean	1.02e-01	**4.67e-02**	1.62e-01	1.96e-01
	Std	1.35e-01	3.58e-02	1.97e-01	1.98e-01
SRN	Mean	**7.40e+00**	1.12e+01	1.27e+02	5.98e+02
	Std	5.39e+00	1.15e+01	1.85e+02	7.74e+02
OSY	Mean	3.53e+01	**1.79e+01**	4.76e+02	3.61e+02
	Std	4.70e+01	1.95e+01	5.92e+02	2.69e+02
TwoBarTrussDesign	Mean	**2.34e+12**	2.51e+12	2.98e+12	1.09e+13
	Std	2.13e+12	3.36e+12	2.64e+12	2.14e+13
WeldedBeamDesign	Mean	1.49e+02	**3.19e+01**	1.50e+02	1.42e+02
	Std	1.33e+02	3.26e+01	2.31e+02	1.51e+02
Rosenbrock	Mean	6.41e-02	**2.00e-02**	1.78e-01	1.56e-01
	Std	5.70e-02	2.11e-02	3.24e-01	1.89e-01
Ackley	Mean	**5.64e-04**	8.51e-04	4.93e-03	5.86e-04
	Std	1.01e-03	1.19e-03	1.03e-02	1.04e-03
Bohachevsky	Mean	6.00e+02	**1.89e+02**	1.05e+04	3.35e+03
	Std	9.21e+02	1.99e+02	2.21e+04	7.06e+03
Griewank	Mean	**1.24e-01**	1.26e-01	5.83e+00	2.44e+00
	Std	1.54e-01	2.89e-01	9.56e+00	4.13e+00
H1	Mean	2.57e-05	1.98e-05	**7.20e-06**	2.58e-05
	Std	4.13e-05	2.50e-05	7.44e-06	5.16e-05
Himmelblau	Mean	3.56e+00	**1.35e+00**	1.48e+01	3.28e+01
	Std	5.10e+00	1.39e+00	2.91e+01	6.09e+01
Rastrigin	Mean	**2.55e-02**	2.01e+00	9.81e-01	2.34e+00
	Std	3.11e-02	5.69e+00	1.32e+00	2.37e+00
Schaffer	Mean	1.33e-02	**3.10e-03**	8.31e-02	2.04e-02
	Std	1.01e-02	5.09e-03	1.52e-01	2.04e-02
Schwefel	Mean	2.14e+01	**1.19e+00**	9.19e+01	3.76e+02
	Std	4.21e+01	1.53e+00	9.86e+01	5.02e+02

Table 1: Distance after 30 iterations for each of the optimization setups. We denote the proposed surrogate model with a predictive Chi-squared distribution by **2-norm** and the common GP surrogate model by **Standard GP**. The two rows for each function list mean and standard deviation for 8 repetitions with the same target vector. Lowest mean values are highlighted.

given acquisition function seems to be dependent on the response surface of the objective.

While the proposed surrogate model in the aggregate outperformed the standard GP the difference was much more pronounced in cases where the response surface was easy to model. This is illustrated in the optimization traces for the Ackley and Bohachevsky objectives shown in Figure 6. The former has a rugged response surface with a large well in the center making it a difficult function to model with a GP relying on a stationary kernel such as the Mátern 5/2. The latter, in contrast, has a smooth surface with no sudden jumps. This indicates to us that the advantage of the proposed method is highly dependent on the individual outputs of the objective function being well-modelled by the surrogate GP.

4.2 Audio target estimation

To evaluate the proposed method on a real-world problem, we considered the task of reverse engineering a musical synthesizer [20, 21, 22, 23]. A synthesizer produces sound by generating waveforms in one or more oscillators and routing the audio streams through a processing pipeline which may include mixing of separate streams, filtering, adding of noise, and saturation. By changing the configuration of this pipeline the musician can design the character of the output sound. A common task, and the one considered here, is to be presented with some target sound and then finding a configuration that approximates this target.

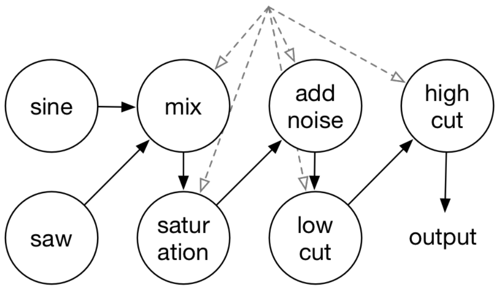

Figure 7: Diagram of the synthesizer's processing pipeline.

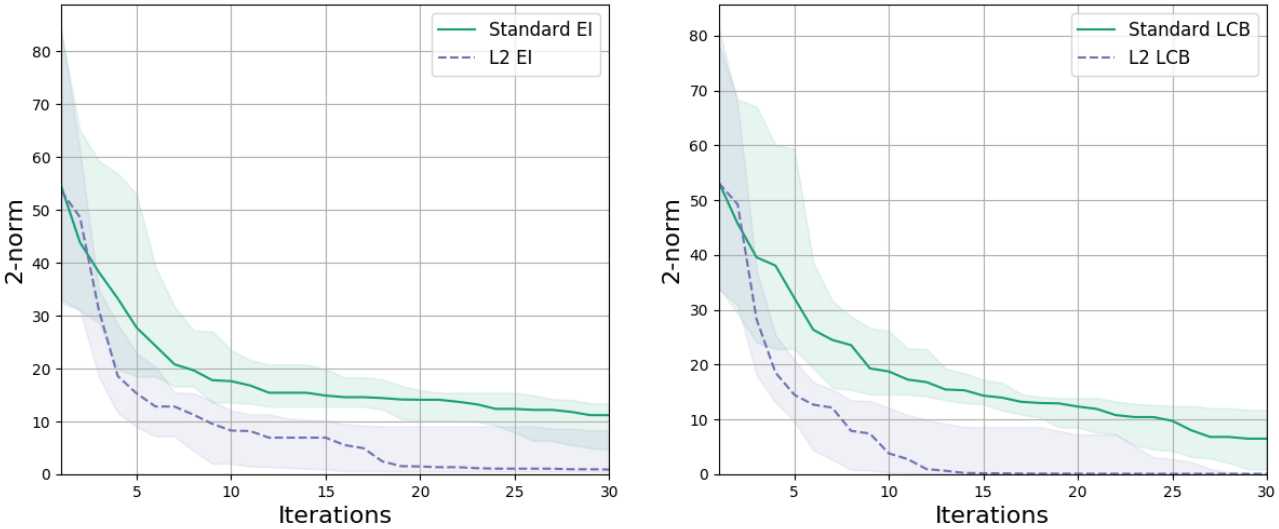

Figure 8: Optimization traces for the 2-norm and the standard GP model for the problem of reverse engineering a synthesizer. Each trace comprises 8 runs of optimization for the same target with the lines being the medians of the incumbent values and the shaded regions being the interquartile ranges.

The internal workings of a synthesizer constitutes the manufacturer's signature sound developed over years or sometimes decades and the processing algorithms can therefore be expected to be both highly complex and unavailable to consumers and potential copycats. Mapping from output sound back to configuration is thus a non-trivial problem apt for the Bayesian optimization framework. And defining the discrepancy between the produced sound and the target through the 2-norm makes this problem suitable for the method proposed in this paper.

The experiments were carried out using a simple, custom built synthesizer which offered 5 free parameters: The mix of a sine and a sawtooth oscillator, the amount of digital saturation, the volume of white noise added, and the frequencies for low cut and high cut filtering. Refer to Figure 7 for a diagram of the processing pipeline. The frequencies produced by the oscillators were fixed so that the output sound was only dependent on the parameter configuration. To get the feature space down to a feasible size we projected the log transformed FFT of the output signal down to 10 dimensions using Principal Analysis Component (PCA) which had been pre-estimated on 200 random outputs. Using the established terminology we thus have $\mathcal{X} \subset \mathbb{R}^5$ as the set of configuration parameters, $\mathcal{Y} \subset \mathbb{R}^{10}$ as the set of output sound, while $h : \mathcal{X} \to \mathcal{Y}$ encompasses the sound generation and subsequent feature extraction.

The experiment was carried out as for the benchmark functions by first producing a target sound for a random configuration which each of the BO setups then were to approximate by minimizing the squared 2-norm in $\mathcal{Y}$-space. We ran the optimization 8 times with different starting points and compared the optimization traces. The results are depicted in Figure 8. As before we see better performance by the two BO setups reliant on the 2-norm model with the LCB acquisition function on average showing faster convergence as well as final result. The advantage is established within the first 5 iterations for both acquisition functions and remains stable throughout the optimization.

5 Conclusion

In this paper we have addressed the problem of estimating a target vector by querying a multi-output function that is blackboxed, stochastic, and expensive to evaluate. We have put forth an approach in which each output component is modelled separately by a Gaussian process such that the sum of squares between target and function evaluation approximately follows a noncentral Chi-squared distribution. We have developed closed-form and computationally efficient acquisition functions for Expected Improvement and Lower Confidence Bound based on the adjusted predictive distribution which are better suited for Bayesian optimization for minimizing distances. An empirical comparison between the proposed model and standard methods shows a significant improvement throughout the optimization both when tested on synthetic benchmark functions and on a practical problem in the audio domain.

References

[1] J. Mockus. On Bayesian methods for seeking the extremum. In *Optimization Techniques IFIP Technical Conference*, pages 400–404. Springer, 1975.

[2] J. Snoek, H. Larochelle, and R. P. Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, pages 2951–2959, 2012.

[3] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2016.

[4] J. Knowles. ParEGO: a hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 10(1):50–66, 2006.

[5] Q. Zhang, W. Liu, E. Tsang, and B. Virginas. Expensive multiobjective optimization by MOEA/D with Gaussian process model. *IEEE Transactions on Evolutionary Computation*, 14, 2010.

[6] S. Jeong and S. Obayashi. Efficient global optimization (EGO) for multi-objective problem and data mining. In *2005 IEEE Congress on Evolutionary Computation*, volume 3, pages 2138–2145. IEEE, 2005.

[7] M. Emmerich and J. Klinkenberg. The computation of the expected improvement in dominated hypervolume of Pareto front approximations. *Rapport technique, Leiden University*, 34, 2008.

[8] D. Hernández-Lobato, J. Hernandez-Lobato, A. Shah, and R. P. Adams. Predictive entropy search for multi-objective Bayesian optimization. In *International Conference on Machine Learning*, pages 1492–1501, 2016.

[9] P. Feliot, J. Bect, and E. Vazquez. A Bayesian approach to constrained single-and multi-objective optimization. *Journal of Global Optimization*, 67(1-2):97–133, 2017.

[10] M. S. Murugan, S. Suresh, R. Ganguli, and V. Mani. Target vector optimization of composite box beam using real-coded genetic algorithm: a decomposition approach. *Structural and Multidisciplinary Optimization*, 33(2):131–146, 2007.

[11] D. Wienke, C. Lucasius, and G. Kateman. Multicriteria target vector optimization of analytical procedures using a genetic algorithm: Part i. theory, numerical simulations and application to atomic emission spectroscopy. *Analytica Chimica Acta*, 265(2):211–225, 1992.

[12] P. Perdikaris and G. E. Karniadakis. Model inversion via multi-fidelity Bayesian optimization: A new paradigm for parameter estimation in haemodynamics, and beyond. *Journal of The Royal Society Interface*, 13(118):20151107, 2016.

[13] The Emukit authors. Emukit: Emulation and uncertainty quantification for decision making. `https://github.com/amzn/emukit`, 2018.

[14] E. Snelson, Z. Ghahramani, and C. E. Rasmussen. Warped Gaussian processes. In *Advances in Neural Information Processing Systems*, pages 337–344, 2004.

[15] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, USA, January 2006.

[16] M. Sankaran. On the non-central chi-square distribution. *Biometrika*, 46(1/2):235–237, 1959.

[17] E. Marchand. Computing the moments of a truncated noncentral Chi-square distribution. *Journal of Statistical Computation and Simulation*, 54(4):387–391, 1996.

[18] F. Fortin, F. De Rainville, M. Gardner, M. Parizeau, and G. Christian. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13:2171–2175, jul 2012.

[19] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[20] A. Horner, J. Beauchamp, and L. Haken. Machine tongues XVI: Genetic algorithms and their application to FM matching synthesis. *Computer Music Journal*, 17(4):17–29, 1993.

[21] R. Garcia. Growing sound synthesizers using evolutionary methods. In *ALMMA 2002: Artificial Life Models for Musical Applications Workshop*, (Cosenza, Italy, 2001).

[22] Y. Lai, S. Jeng, D. Liu, and Y. Liu. Automated optimization of parameters for FM sound synthesis with genetic algorithms. In *International Workshop on Computer Music and Audio Technology*, 2006.

[23] S. Heise, M. Hlatky, and J. Loviscach. Automatic cloning of recorded sounds by software synthesizers. In *Audio Engineering Society Convention 127*. Audio Engineering Society, 2009.