SAMPLE EFFICIENT MACHINE LEARNING WITH PHYSICS-INFORMED KERNEL METHODS AND SAMPLING TECHNIQUES

S. Preston*[1, 2, 3], N. Blaskovic Kraljevic[2], I. P. S. Martin[2, 3], P. N. Burrows[3]
[1]Ada Lovelace Centre, Oxfordshire, UK
[2]Diamond Light Source, Oxfordshire, UK
[3]John Adams Institute for Accelerator Science, Oxfordshire, UK

Abstract

It is desirable to reduce the convergence time of optimisers used by large accelerator facilities to make best use of the available development time. A popular technique is Bayesian optimisation (BO) which typically use Gaussian processes (GP) to construct a surrogate of the real machine response to decision variables. GPs belong to a class of algorithms called kernel methods that assign pairwise similarities to all data points with a kernel function. While well-suited to tasks like injection, the kernel matrix must be stored and inverted at inference time, incurring a time-complexity and limiting datasets to a few thousand examples in practice. The modeller is free to design the kernel, subject to some mild regularity conditions. We investigate whether modifications made to the kernel structure, informed by the physics of our problems, can improve sample efficiency. Acquisition function strategies are also discussed to further improve performance.

INTRODUCTION

Recent work at Diamond Light Source has studied how the performance of Bayesian optimisation can be improved when tuning magnets in the linac-to-booster and booster-to-storage ring (BTS) transfer lines by exploiting knowledge of the physics in those regions. The only magnets present for controlling the beam are dipole and quadrupole magnets, making the dynamics linear. Roll errors on the magnets are typically small, such that the coupling between the horizontal and vertical planes can be neglected to a first approximation. Standard GP optimiser kernels like radial basis function (RBF) and Matérn [1] are local in the sense that they contain products of terms that become smaller at coordinates far from observed data points. As a result, predictions are miscalibrated in regions far from the training data and this can hurt the discovery process for new points in the BO loop. In this paper we demonstrate how the choice of kernel impacts the search for optimal inputs.

THEORY

Gaussian Process

A GP is a probabilistic surrogate model with a mean, $\mu(\cdot)$, and kernel function, $k(\cdot,\cdot)$, where each $\cdot$ represents a variable. The kernel is also known as a covariance or similarity function. The mean $\mu(x)$ and marginal variance $k(x,x)$ at a point x define the marginal predictive distribution there. Conditioning on data is equivalent to creating a kernel, $k_i = k(\vec{x}_i, \cdot)$, for every data point, each with the same fixed hyperparameters, and finding scale factors for each kernel such that the superposition of all kernels faithfully reproduces the observed data. In standard BO, the likelihood is Gaussian which means the posterior can be written in closed form and is also Gaussian. This means the hyperparameters of the kernel and the weight vector are simultaneously found by minimising the negative log marginal likelihood (absent a constant).

$$-ln\, p(\vec{y} \mid X) = \frac{1}{2}(\vec{y} - \vec{m}_x)^{\mathrm{T}}(K_\omega + \sigma_n^2\mathbb{I})^{-1}(\vec{y} - \vec{m}_x) + \frac{1}{2} ln \mid K_\omega + \sigma_n^2\mathbb{I} \mid$$

where $X \in \mathbb{R}^{\mathrm{N}\times\mathrm{D}}$ is a design matrix of data point coordinates, $\vec{y}$ are the measurements associated with them, $\vec{m}_x$ is the GP mean at those points, K_ω is the covariance or Gramm matrix such that $K_{\omega_{ij}} = k(\vec{x}_i, \vec{x}_j; \omega)$ where k has hyperparameters ω, σ_n is the aleatoric uncertainty of the system, $\mathbb{I}$ is the identity matrix, and n is the number of data points that have been collected. The final term in the RHS is a fixed constant and is usually dropped during this step.

Kernel

A kernel is a function of two inputs that produces a scalar output. It sits within a space of square-integrable functions known as a reproducing kernel Hilbert space (RKHS).

$$k : \mathbb{R} \times \mathbb{R} \to \mathbb{R}.$$

If the kernels are fixed at data points they become functions of a single free variable. Square-integrability then implies that the integral of the square of the magnitude of the kernel over the entire domain is finite.

$$\int_{-\infty}^{\infty} |k_i(\vec{x})|^2 d\vec{x} < \infty.$$

Hence it can be concluded that any function which is square-integrable and meets some additional criteria like being positive semi-definite are valid kernels and can be used by a GP in a consistent way. Commonly used kernels are the RBF which is smooth and Matérn which has a finite number of continuous derivatives. A property that follows from this is that sums and products of kernels are also valid kernels. An additive kernel is motivated by considering that

* shaun.preston@physics.ox.ac.uk

some multivariate functions $f = f(x_1, ..., x_n)$ are the result of the sum of several different orders of interaction between its input dimensions [2]. If a base kernel for each dimension is chosen, k_d, the kernel evaluated on a data point is

$$k_d = k_d(x_d)$$

where x_d is the dth element of a data point vector (the arrow has been dropped for clarity). The first order of interaction does not account for any interaction between dimensions and is written as

$$k_{add_1}(x, x') = \sigma_1^2 \sum_{i=1}^{D} k_i(x_i, x_i'), \tag{1}$$

where σ_n^2 is the variance of the nth order of interaction. To further illustrate, the second order interaction between dimension pairs is

$$k_{add_2}(x, x') = \sigma_2^2 \sum_{i=1}^{D} \sum_{j=i+1}^{D} k_i(x_i, x_i')\, k_j(x_j, x_j'),$$

A general expression for interactions of any order can be found in [2], however the sums quickly become intractable in high dimensions because the number of terms grows combinatorially. Furthermore, only first order interactions are considered in these studies. As a consequence, global features become learnable despite the base kernels themselves being local, which is shown in Fig. 1. A drawback is that the number of hyperparameters which must be optimised is directly related to the number of dimensions as well as the number of sub-kernels used to form the base kernel in each dimension. Although sample efficiency may improve, the time needed to perform the calculations could grow enough to negate any potential speed-ups.

Linear Dynamics as Additivity

The beam envelope is parametrised by the longitudinal coordinate s which implies that transmission is most sensitive at locations where the beta function is largest, since the maximum separation between the envelope and beam pipe is smallest. Most of the beam losses that occur due to changing the corrector strengths will come from it scraping the vacuum pipe wall and other components. In real machines, there is typically a ceiling on the maximum achievable transmission and this can vary greatly. The sum of charges at beam position monitors (BPMs) can be written

$$f = f(x_{H_1}, ..., x_{H_n}, x_{V_1}, ..., x_{V_n})$$

where x_{H_i} and x_{V_j} are the ith and jth horizontal and vertical corrector strengths, respectively. If the dynamics in the transverse planes are truly decoupled due to negligible errors and fringe effects, an inductive bias for the GP is

$$f \approx f(\vec{x}_H) + f(\vec{x}_V) \tag{2}$$

The objective is now in a suitable form to be optimised with an additive kernel. What remains is to discover the extent to which different orders of interaction between correctors in the same plane affect the objective value.

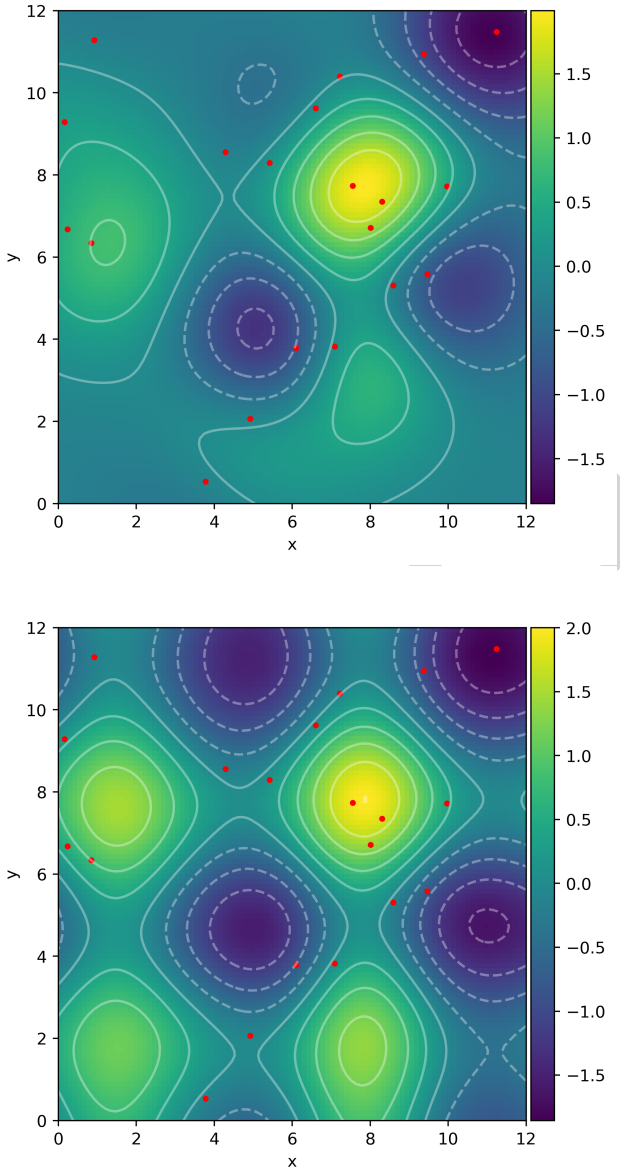

Figure 1: Two GPs are trained on the same random dataset of 20 examples (red dots). The ground truth function is $sin\ x + sin\ y$. A single RBF kernel is used, predictions are inaccurate in regions with high uncertainty (top). First-order interactions with RBF base kernels are used (bottom). White lines represent isocontours of the GP mean surfaces.

METHODOLOGY

The same randomly initialised beam of 2,000 electrons was instantiated at the beginning of a model of the Booster-to-storage ring (BTS) transfer line lattice in the PyAT particle tracking code [5] with a sigma matrix that is a close approximation of the real machine. The Twiss parameters were $\alpha_x = -2.92$, $\alpha_y = 0.75$, $\beta_x = 12.13$m, $\beta_y = 2.94$m and the horizontal and vertical emittance $\epsilon_{xy} = 2.6 \times 10^{-7}$ m rad.

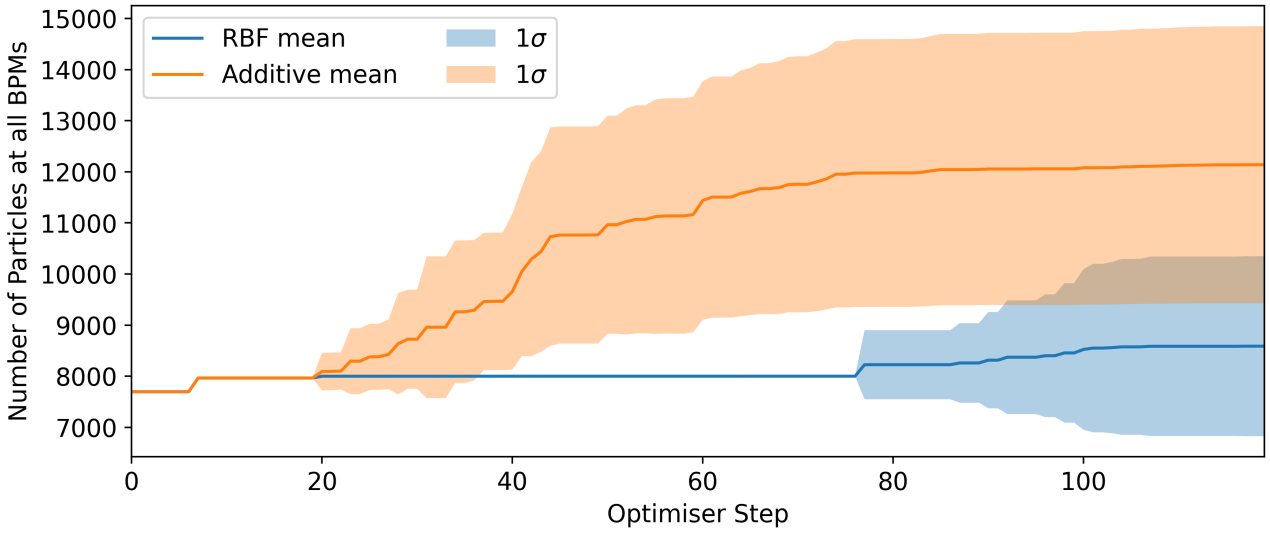

Figure 2: Two separate studies: blue — single RBF kernel; orange — first-order interaction additivity with Matérn base kernel with $\nu = 3/2$. The average best performance of a GP optimiser tasked with maximising beam transmission along the BTS transfer line in a simulation by tuning 14 corrector magnets. Trust region Bayesian optimisation (TuRBO) [3, 4] is used to accelerate the progress in the high-dimensional setting. In each study, ten identical optimisers with the same initial conditions and same 20 samples were run. The solid lines represent the average within a study and the shaded regions represent the respective 1σ confidence bands. Seven BPMs and 2,000 particles imply that 14,000 particles is the maximum theoretical value.

$\pm$25mm rectangular apertures were added between each element to simulate the vacuum pipe wall of the real machine. All 14 corrector magnets were controlled in the studies. An identical batch of 50 initial sample trajectories and their associated corrector strength vectors were recorded and used to seed the optimisers. Upper confidence bound (UCB) was chosen as the acquisition function with $\beta = 2$. A constant zero-mean prior was also used. Each configuration of an optimiser was rerun ten times with identical initial conditions. The variability in the optimiser history comes from the stochasticity of searching the likelihood landscape to optimise hyperparameters, using the Xopt Python package [6]. This gives a summary of the average performance of a particular configuration. For performance reasons, the maximum order of interaction between the corrector strengths in the optimisations was kept at two, although it was occasionally dropped to one. This is necessary because the number of hyperparameters quickly grows with the number of dimensions and base kernel components. Additionally, the likelihood landscape becomes less smooth and the gradient descent algorithm used to find optimal hyperparameters, L-BFGS [7], becomes stuck in poor regions or fails to converge altogether.

RESULTS

The toy problem from Fig. 1 illustrated how global structure could be found with additivity. The benefits of exploiting the uncoupled dynamics to design a custom kernel become even more apparent in higher dimensions. In Fig. 2 it is shown that the additive kernel performs better as it converges to an optimal solution earlier than the standard RBF kernel, however, the variance is larger.

CONCLUSION AND FUTURE WORK

The motivation for an additive kernel structure in the context of particle accelerator injection is explained and benchmarked against a standard kernel method at various configurations. Cautionary tales are given about extending these methods to higher-dimensional domains. It is shown that additive kernels allow local stationary kernels to discover global trends in a system response, improving sample efficiency. In the future, the additive structure will be tested on the real transfer line to see how robust it is to measurement noise from the BPMs. It is also intended to test different acquisition function strategies to determine if further performance gains are achievable. This means benchmarking UCB which is intuitive to use against expected improvement which tends to be more efficient at finding a global optimum, as well as a Thompson sampling approach that combines the two approaches.

REFERENCES

[1] M. Kanagawa *et al.*, "Gaussian processes and kernel methods: a review on connections and equivalences", 2018. doi:10.48550/arXiv.1807.02582

[2] D. K. Duvenaud, H. Nickisch, and C. E. Rasmussen, "Additive gaussian processes", in *Adv. in Neur. Inf. Proc. Syst.*, vol. 24, 2011. doi:10.48550/arXiv.1112.4394

[3] D. Eriksson *et al.*, "Scalable global optimization via local bayesian optimization", in *Adv. in Neur. Inf. Proc. Syst.*, vol. 32, 2019. doi:10.48550/arXiv.1910.01739

[4] R. Roussel *et al.*, "Bayesian optimization algorithms for accelerator physics", *Phys. Rev. Accel. Beams*, vol. 27, no. 8, p. 084801, 2024. doi:10.1103/PhysRevAccelBeams.27.084801

[5] W. A. H. Rogers *et al.*, "pyAT: a python build of accelerator toolbox", in *Proc. IPAC'17*, Copenhagen, Denmark, pp. 3855–3857, May 2017. doi:10.18429/JACoW-IPAC2017-THPAB060

[6] R. Roussel *et al.*, "Xopt: a simplified framework for optimization of accelerator problems using advanced algorithms", in *Proc. IPAC'23*, Venice, Italy, pp. 4847–4850, May 2023. doi:10.18429/JACoW-IPAC2023-THPL164

[7] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization", *Math. Program.*, vol. 45, no. 1, pp. 503–528, 1989. doi:10.1007/BF01589116