Quantifying mismatch in Bayesian optimization

Eric Schulz
University College London
e.schulz@cs.ucl.ac.uk

Maarten Speekenbrink
University College London
m.speekenbrink@ucl.ac.uk

José Miguel Hernández-Lobato
University of Cambridge
jmh233@cam.ac.uk

Zoubin Ghahramani
University of Cambridge
zoubin@eng.cam.ac.uk

Samuel J. Gershman
Harvard University
gershman@fas.harvard.edu

Abstract

How does misspecifying prior smoothness assumptions about the target function affect the performance of Bayesian optimization routines? We show that misspecifying smoothness leads to an increase in regret, that this effect gets worse in higher dimensions, and that it remains substantial even if hyper-parameters are optimized.

1 Introduction

The goal of Bayesian optimization is to find the maximum of a target function $f(x)$ on some bounded set $\mathcal{X}$, which we will take to be a subset of $\mathbb{R}^D$ [1, 10, 9] by using a Gaussian Process ($\mathcal{GP}$) as a probabilistic model for $f(x)$ and then exploiting this model via an acquisition function to decide where in $\mathcal{X}$ to evaluate the function next [2]. Even though many acquisition functions have been compared before [cf. 1, 10] and theories about how $\mathcal{GP}$s behave in mismatched learning scenarios have been put forward [see 14, 11], little is known about how choosing a particular kernel affects Bayesian optimization routines in practice.

2 Gaussian Process

Let $f(\boldsymbol{x})$ be a function mapping an input $\boldsymbol{x} = (x_1, \ldots, x_d)^\top$ to an output y. A $\mathcal{GP}$ defines a distribution $p(f)$ over such functions [8]. A $\mathcal{GP}$ is parametrized by a mean function $m(\boldsymbol{x})$ and a kernel function, $k(\boldsymbol{x}, \boldsymbol{x}')$:

$$m(\boldsymbol{x}) = \mathbb{E}\left[f(\boldsymbol{x})\right] \tag{1}$$

$$k(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}\left[(f(\boldsymbol{x}) - m(\boldsymbol{x}))(f(\boldsymbol{x}') - m(\boldsymbol{x}'))\right] \tag{2}$$

At time t, we have collected observations $\mathbf{y}_{1:t} = [y_1, y_2, \ldots, y_t]^\top$ at inputs $\boldsymbol{x}_{1:t} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_t)$. For each outcome y_t, we assume $y_t = f(\boldsymbol{x}_t) + \epsilon_t$ with $\epsilon \sim \mathcal{N}(0, \sigma^2)$. Given a $\mathcal{GP}$ prior on the functions $f(\boldsymbol{x}) \sim \mathcal{GP}\left(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')\right)$, the posterior over f is a $\mathcal{GP}$ with

$$m_t(\boldsymbol{x}) = k_{1:t}(\boldsymbol{x})^\top (\boldsymbol{K}_{1:t} + \sigma^2 \boldsymbol{I}_t)\boldsymbol{y}_{1:t} \tag{3}$$

$$k_t(\boldsymbol{x}, \boldsymbol{x}') = k(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{k}_{1:t}(\boldsymbol{x})^\top (\boldsymbol{K}_{1:t} + \sigma^2 \boldsymbol{I}_t)^{-1} \boldsymbol{k}_{1:t}(\boldsymbol{x}') \tag{4}$$

where $\boldsymbol{k}_{1:t}(\boldsymbol{x}) = [k(\boldsymbol{x}_1, \boldsymbol{x}), \ldots, k(\boldsymbol{x}_t, \boldsymbol{x})]^\top$, $\boldsymbol{K}_{1:t}$ is the positive definite kernel matrix $[k(\boldsymbol{x}_i, \boldsymbol{x}_j)]_{i,j=1,\ldots,t}$, and $\boldsymbol{I}_t$ is a t by t identity matrix.

29th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

3 Acquisition Functions

Upper Confidence Bound Sampling Upper confidence bound (UCB) sampling picks the next point that currently has the highest upper confidence bound [12]:

$$\text{UCB}(\boldsymbol{x}) = \mu(\boldsymbol{x}) + \beta\sigma(\boldsymbol{x}) \tag{5}$$

UCB sampling is an optimistic strategy that samples based on an explicit exploration-exploitation trade-off, and has been proven to produce sub-linear regret, $r_t = f(\boldsymbol{x}_{\text{max}}) - f(\boldsymbol{x}_t)$ [13].

Probability of Improvement The Probability of Improvement (POI) sampler picks the next point that currently shows the highest probability of being better than an incumbent point which is normally set to $\boldsymbol{x}^+ = \text{argmax}_{\boldsymbol{x}_i \in \boldsymbol{x}_{1:t}} f(\boldsymbol{x}_i)$—i.e., the best outcome observed so far [5]:

$$\text{POI}(\boldsymbol{x}) = P(f(\boldsymbol{x}) \geq f(\boldsymbol{x}^+ + \xi) = \Phi\left(\frac{\mu(\boldsymbol{x}) - f(\boldsymbol{x}^+ - \xi)}{\sigma(\boldsymbol{x})}\right) \tag{6}$$

where $\Phi(\cdot)$ is the normal CDF and $\xi \geq 0$ is a trade-off parameter controlling exploration.

Expected Improvement The Expected Improvement (EXI) assesses each point by how much in expectation it promises to be better than an incumbent point $\boldsymbol{x}^+$ [7]:

$$\text{EXI}(\boldsymbol{x}) = \begin{cases} (\mu(\boldsymbol{x}) - f(\boldsymbol{x}^+) - \xi)\Phi(Z) + \sigma(\boldsymbol{x})\phi(Z), & \text{if } \sigma(\boldsymbol{x}) \geq 1 \\ 0, & \text{if } \sigma(\boldsymbol{x}) = 0 \end{cases} \tag{7}$$

where $\phi(\cdot)$ is the normal PDF and $Z = (\mu(\boldsymbol{x}) - f(\boldsymbol{x}^+) - \xi)/\sigma(\boldsymbol{x})$.

Predictive Entropy Search Predictive entropy search (PES) is an information-based acquisition function that samples points that promise to maximally reduce the differential entropy of an unknown optimizer $\boldsymbol{x}^* = \text{argmax}_{\boldsymbol{x} \in \mathcal{X}} f(\boldsymbol{x})$ [3, 4]. After having observed data $\mathcal{D}_n$, the posterior distribution of $\boldsymbol{x}^*$ is $p^*(x|\mathcal{D})$. The predictive entropy search sampler then is

$$\text{PES}(\boldsymbol{x}) = H(y|\mathcal{D}_n, \boldsymbol{x}) - \mathbb{E}_{\boldsymbol{x}^*|\mathcal{D}_n}\left[H(y|\mathcal{D}_n, \boldsymbol{x}, \boldsymbol{x}^*)\right] \tag{8}$$

where $H(\boldsymbol{x}^*|\mathcal{D}_n)$ denotes the differential entropy of $p^*(x|\mathcal{D})$. We will approximate the differential entropy by Monte Carlo sampling from the posterior $\mathcal{GP}$ [4].

Minimum Regret Search Minimum regret point (MRP) sampling samples the point producing the lowest simple regret [6]. The expected simple regret (ER) of selecting $\boldsymbol{x}$ under $p(f)$ can be expressed as $\mathbb{E}_{p(f)}[R_f(\boldsymbol{x})] = \mathbb{E}_{p(f)}[f(\boldsymbol{x}_{\text{max}}) - f(\boldsymbol{x})]$. MRP picks the point which has the minimal expected regret under $p_n(f)$, $\tilde{x} = \text{argmin}_{\boldsymbol{x}} ER(p_n)(\boldsymbol{x})$.

$$\text{MRP}(\boldsymbol{x}) = \min \text{ER}(p_n)(\tilde{x}) - \mathbb{E}_{y|p_n(f),\boldsymbol{x}}[\min_{\tilde{\boldsymbol{x}}} \text{ER}(p_n^{[\boldsymbol{x},y]})(\tilde{\boldsymbol{x}})] \tag{9}$$

where $p_n^{[\boldsymbol{x},y]} = p(f|\mathcal{D} \cup \{\boldsymbol{x}^q, y\})$ is the updated probability distribution of f after having queried $\boldsymbol{x}^q$ and observing $y = f(\boldsymbol{x}^q) + \epsilon$. MRP sampling can also be modified to account for the inherent uncertainty about $\tilde{\boldsymbol{x}}$ via marginalization leading to Minimum Regret Search (MRS) as described in detail by [6].

$$\text{MRS}(\boldsymbol{x}) = \mathbb{E}_{\tilde{\boldsymbol{x}} \sim p_n^*}[\text{ER}(p_n)(\tilde{x})] - \mathbb{E}_{y|p_n(f),\boldsymbol{x}^q}[\mathbb{E}_{p^*_{\mathcal{D}_n \cup \{(\boldsymbol{x}^q, y)\}}}[\text{ER}(p_n^{[\boldsymbol{x},y]})(\tilde{\boldsymbol{x}})]] \tag{10}$$

4 Encoding Smoothness

The Matérn class of kernel functions encodes the expected smoothness of the target function explicitly. The Matérn covariance between two points separated by τ distance units is

$$k_\nu(\tau) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\frac{\tau}{\rho}\right) K_\nu \left(\sqrt{2\nu}\frac{\tau}{\rho}\right) \tag{11}$$

where Γ is the gamma function, K_v is the modified Bessel function of the second kind, and ρ and ν are non-negative covariance parameters. A $\mathcal{GP}$ with a Matérn covariance has sample paths that are $\nu - 1$ times differentiable. When $\nu = p + 0.5$, the Matérn kernel can be written as a product of an exponential and a polynomial of order p.

$$k_{p+0.5}(\tau) = \sigma^2 \exp\left(-\frac{\sqrt{2\nu}\tau}{\rho}\right) \frac{\Gamma(p+1)}{(2p+1)} \times \sum_{i=0}^{p} \frac{(p+i)!}{i!(p-i)!} \left(\frac{\sqrt{8\nu}\tau}{\rho}\right)^{p-i} \tag{12}$$

We will compare two extreme cases in our simulations, the Ornstein-Uhlenbeck process, $k(\tau) = \sigma^2 \exp(-\tau/\rho)$, that results when setting $p = 0$ and the radial basis function kernel that results when $p \to \infty$, $k(\tau) = \sigma^2 \exp(-\tau^2/2\rho^2)$. Figure 1 shows prior samples of differently smooth $\mathcal{GP}$s.

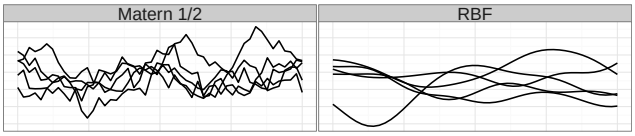

Figure 1: Prior $\mathcal{GP}$-samples from Matérn 1/2 and RBF kernel.

5 Simulations

We created a function $y_t = f(\boldsymbol{x}_t) + \epsilon$ by sampling $f \sim \mathcal{GP}(\mathbf{0}, k_{\text{mat}}(\boldsymbol{x}, \boldsymbol{x}'))$ and $\epsilon \sim \mathcal{N}(0, 10^{-3})$ from a teacher kernel and then used a student kernel to optimize it. This procedure was repeated 100 times for every student-teacher-acquisition function combination.

5.1 Simple regret

In the first simulation, we fixed the length-scale parameter to $\rho = 0.1$ for all student and teacher kernels. Figure 2 shows the median regret (left) and the median distance to the best input point (right) over 100 trials. Maximizing an unsmooth function seems to be harder than maximizing a

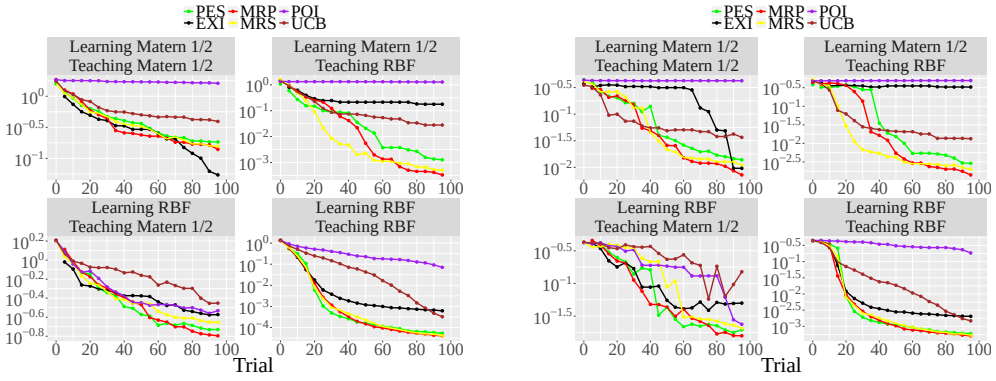

Figure 2: Median simple regret (left) and median distance to best input space (right).

smooth function. Beyond that, mismatched optimization leads to worse performance as compared to expecting the right level of smoothness. The worst possible regret results when expecting a smooth function and in reality optimizing a very rough function. Information-based acquisition functions performed best overall. Mismatch mattered more than specifying the right acquisition function.

5.2 Regret in higher dimensions

We repeated the same simulation but this time over 50 trials and for $d = \{2, 3, 4\}$ dimensions of the input space over which the function had to be optimized. Figure 3 shows the root mean squared difference between each acquisition function's performance in the mismatched setting compared to

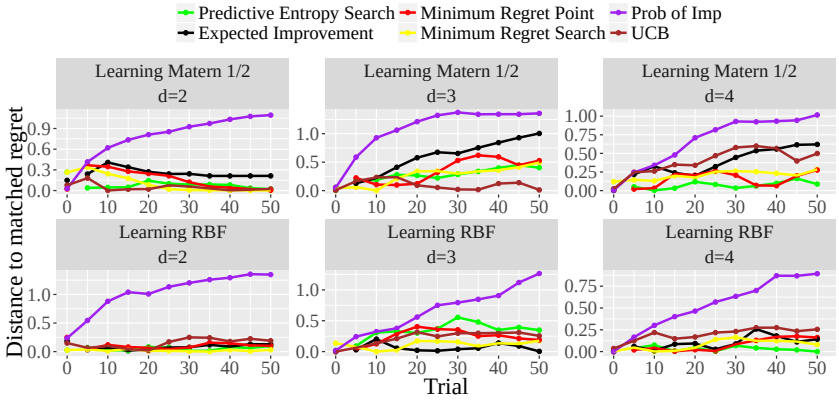

Figure 3: Median counter-factual regret over 50 trials for different input-dimensions.

the same acquisition function's performance in the matched setting for each dimension. This provides a relative measure of counter-factual regret, i.e. by how much could each of the acquisition function have done better on each trial if it had been specified with the right smoothness prior. We can see that, whereas most acquisition functions seem to recover from misspecified assumptions in the bi-variate case, the effect of misspecifications seems to exacerbate over time in higher dimensions.

5.3 Optimizing hyper-parameters

Next, we assessed the effect of mismatch when the student kernel's hyper-parameters θ are optimized via the log marginal likelihood $\log p(y|X,\theta) = -\frac{1}{2}y^\top K_y^{-1}y - \frac{1}{2}\log|K_y| - \frac{n}{2}\log 2\pi$.

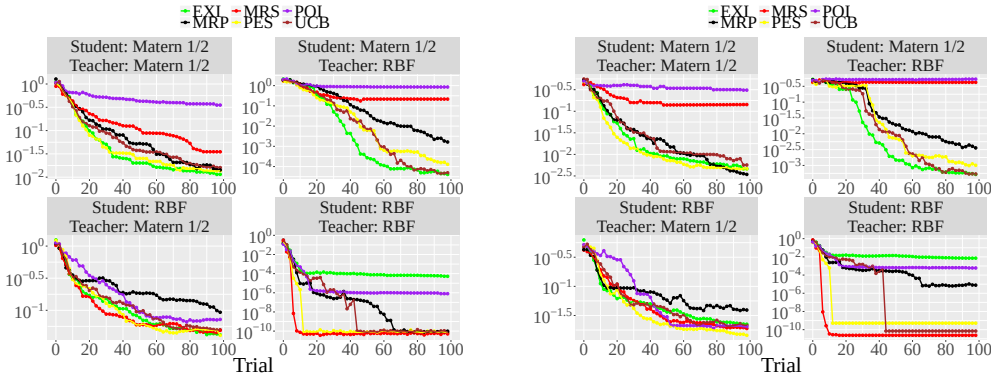

Figure 4: Median regret (left) and distance to best input space (right) with optimized hyper-parameters.

Figure 4 shows that mismatch still affected the performance of the Bayesian optimization routine. When the teacher and student both come from a Matérn kernel, than the lowest overall reached regret is in the region of 10^{-2}, whereas the best regret when trying to optimize the Matérn kernel by using a RBF kernel is $10^{-1.5}$. The effect of mismatch is even stronger in the case of a RBF-teacher, where the best overall regret in the matched case is 10^{-10} but only $10^{-4.5}$ for the mismatched case.

6 Conclusion

Bayesian optimization is a popular method to maximize black box functions. However, taking the term "black box" too literally can lead to noticeable under-performance. We have shown that encoding the wrong prior assumptions about a target function's smoothness can lead to an increase in regret, that this effect gets worse in higher dimensions, and remains even if hyper-parameters are optimized. Therefore, our results tell a cautionary tale about how thinking hard about prior assumptions can sometimes be more important than choosing one particular acquisition function over another.

References

[1] E. Brochu, V. M. Cora, and N. De Freitas. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010.

[2] Z. Ghahramani. Probabilistic machine learning and artificial intelligence. *Nature*, 521(7553):452–459, 2015.

[3] P. Hennig and C. Schuler. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13:1809–1837, June 2012.

[4] J. M. Hernández-Lobato, M. W. Hoffman, and Z. Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *Advances in neural information processing systems*, pages 918–926, 2014.

[5] H. J. Kushner. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. *Journal of Basic Engineering*, 86(1):97–106, 1964.

[6] J. H. Metzen. Minimum regret search for single- and multi-task optimization. In *Proceedings of 33rd International Conference on Machine Learning (ICML)*, page 10, 2016.

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

[8] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[9] 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.

[10] 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.

[11] P. Sollich. Can Gaussian process regression be made robust against model mismatch? In *Deterministic and Statistical Methods in Machine Learning*, pages 199–210. Springer, 2005.

[12] N. Srinivas, A. Krause, S. Kakade, and M. Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proc. International Conference on Machine Learning (ICML)*, 2010.

[13] N. Srinivas, A. Krause, S. Kakade, and M. Seeger. Information-theoretic regret bounds for Gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, May 2012.

[14] A. W. van der Vaart and J. H. van Zanten. Rates of contraction of posterior distributions based on Gaussian process priors. *The Annals of Statistics*, pages 1435–1463, 2008.