A warped kernel improving robustness in Bayesian optimization via random embeddings

Mickaël Binois, David Ginsbourger, Olivier Roustant

A warped kernel improving robustness in Bayesian optimization via random embeddings

Mickaël Binois[1,2] David Ginsbourger[3] Olivier Roustant[1]

[1] Mines Saint-Étienne, UMR CNRS 6158, LIMOS, F-42023 Saint-Étienne, France, firstname.lastname@mines-stetienne.fr
[2] Renault S.A.S., 78084 Guyancourt, France
[3] University of Bern, Department of Mathematics and Statistics, Alpeneggstrasse 22, CH-3012 Bern, Switzerland, david.ginsbourger@stat.unibe.ch

Abstract

This works extends the Random Embedding Bayesian Optimization approach by integrating a warping of the high dimensional subspace within the covariance kernel. The proposed warping, that relies on elementary geometric considerations, allows mitigating the drawbacks of the high extrinsic dimensionality while avoiding the algorithm to evaluate points giving redundant information. It also alleviates constraints on bound selection for the embedded domain, thus improving the robustness, as illustrated with a test case with 25 variables and intrinsic dimension 6.

Keywords: Black-box optimization, Expected Improvement, low-intrinsic dimensionality, Gaussian processes, REMBO

1 Introduction

The scope of Bayesian Optimization methods is usually limited to moderate-dimensional problems [2]. To overcome this restriction, [9] recently proposed to extend the applicability of these methods to up to billions of variables, when only few of them are actually influential, through the so-called Random EMbedding Bayesian Optimization (REMBO) approach. In REMBO, optimization is conducted in a low-dimensional domain $\mathcal{Y}$, randomly embedded in the high-dimensional source space $\mathcal{X}$. New points are chosen by maximizing the Expected Improvement (EI) criterion [4] with Gaussian process (GP) models incorporating the considered embeddings via two kinds of covariance kernels proposed in [9]. A first one, $k_{\mathcal{X}}$, relies on Euclidean distances in $\mathcal{X}$. It delivers good performance in moderate dimension, albeit its main drawback is to remain high-dimensional so that the benefits of the method are limited. A second one, $k_{\mathcal{Y}}$, is defined directly over $\mathcal{Y}$ and is therefore independent from the dimension of $\mathcal{X}$. However, it has been shown [9] to possess artifacts that may lead EI algorithms to spend many iterations exploring equivalent points.

Here we propose a new kernel with a warping (see e.g. [7]) inspired by simple geometrical ideas, that retains key advantages of $k_{\mathcal{X}}$ while remaining of low dimension like $k_{\mathcal{Y}}$. Its effectiveness is illustrated on a 25-dimensional test problem with 6 effective variables.

2 Background on the REMBO method and related issues

The considered minimization problem is to find $\mathbf{x}^* \in \text{argmin}_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$, with $f : \mathcal{X} \subseteq \mathbb{R}^D \to \mathbb{R}$, where $\mathcal{X}$ is a compact subset of $\mathbb{R}^D$, assumed here to be $[-1, 1]^D$ for simplicity. From [9], one main hypothesis about f is that its effective dimensionality is $d_e < D$: there exists a linear subspace $\mathcal{T} \subset \mathbb{R}^D$ of dimension d_e such that $f(\mathbf{x}) = f(\mathbf{x}_\top + \mathbf{x}_\perp) = f(\mathbf{x}_\top)$, $\mathbf{x}_\top \in \mathcal{T}$ and $\mathbf{x}_\perp \in \mathcal{T}^\perp \subset \mathbb{R}^D$ ([9], Definition 1). Given a random matrix $\mathbf{A} \in \mathbb{R}^{D\times d}$ $(d \geq d_e)$ with components sampled independently from $\mathcal{N}(0, 1)$, for any optimizer $\mathbf{x}^* \in \mathbb{R}^D$, there exists at least a point $\mathbf{y}^* \in \mathbb{R}^d$ such that $f(\mathbf{x}^*) = f(\mathbf{A}\mathbf{y}^*)$ with probability 1 ([9], Theorem 2.). To respect box constraints, f is evaluated at $p_\mathcal{X}(\mathbf{A}\mathbf{y})$, the convex projection of $\mathbf{A}\mathbf{y}$ onto $\mathcal{X}$. The low dimensional function to optimize is then $g : \mathbb{R}^d \to \mathbb{R}$, $g(\mathbf{y}) = f\left(p_\mathcal{X}(\mathbf{A}\mathbf{y})\right)$.

Optimizing g is carried out using Bayesian Optimization, e.g, with the EGO algorithm [1]. It bases on Gaussian Process Regression [5], also known as Kriging [3], to create a surrogate of g. Supposing that g is a sample from a GP with known mean (zero here to simplify notations) and covariance kernel $k(.,.)$, conditioning it on n observations $\mathbf{Z} = f(\mathbf{x}_{1:n}) = g(\mathbf{y}_{1:n})$, provides a GP $Z(.)$ with mean $m(\mathbf{x}) = \mathbf{k}(\mathbf{x})^T K^{-1}\mathbf{Z}$ and kernel $c(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}') - \mathbf{k}(\mathbf{x})^T K^{-1}\mathbf{k}(\mathbf{x}')$, where $\mathbf{k}(\mathbf{x}) = (k(\mathbf{x}, \mathbf{x}_i))_{1\leq i\leq n}$ and $K = (k(\mathbf{x}_i, \mathbf{x}_j))_{1\leq i,j\leq n}$. The choice of k is preponderant, since it reflects a number of beliefs about the function at hand. Among the most commonly used are the "squared exponential" (SE) and "Matérn" stationary kernels, with hyperparameters such as length scales or degree of smoothness [6, 8]. For REMBO, [9] proposed two versions of the SE kernel with length scales l, namely the low-dimensional $k_\mathcal{Y}(\mathbf{y}, \mathbf{y}') = \exp\left(-\|\mathbf{y} - \mathbf{y}'\|_d^2/2l_\mathcal{Y}^2\right)$ and the high-dimensional $k_\mathcal{X}(\mathbf{y}, \mathbf{y}') = \exp\left(-\left\|p_\mathcal{X}(\mathbf{A}\mathbf{y}) - p_\mathcal{X}(\mathbf{A}\mathbf{y}')\right\|_D^2/2l_\mathcal{X}^2\right)$ $(\mathbf{y}, \mathbf{y}' \in \mathcal{Y})$.

Selecting the domain $\mathcal{Y} \subset \mathbb{R}^d$ is a major difficulty of the method: if too small, the optimum may not be reachable while a too large domain renders optimizing harder, in particular since $p_\mathcal{X}$ is far from being injective. Distant points in $\mathcal{Y}$ may coincide in $\mathcal{X}$, especially far from the center, so that using $k_\mathcal{Y}$ leads to sample useless new points in $\mathcal{Y}$ corresponding to the same location in $\mathcal{X}$ after the convex projection. On the other hand, $k_\mathcal{X}$ suffers from the curse of dimensionality when $\mathcal{Y}$ is large enough so that most or all of the points of $\mathcal{X}$ belonging to the convex projection of the subspace spanned by $\mathbf{A}$ onto $\mathcal{X}$ have at least one pre-image in $\mathcal{Y}$. Indeed, whereas embedded points $p_\mathcal{X}(\mathbf{A}\mathbf{y})$ lie in a d dimensional subspace when they are inside of $\mathcal{X}$, they belong to a D-dimensional domain when they are projected onto the faces and edges of $\mathcal{X}$. To alleviate these shortcomings, after showing that with probability $1 - \epsilon$ the optimum is contained in the centered ball of radius d_e/ϵ (Theorem 3), the authors of [9] then suggest to set $\mathcal{Y} = [-\sqrt{d}, \sqrt{d}]^d$. In practice, they split the evaluation budget over several random embeddings or set $d > d_e$ to increase the probability for the optimum to actually be inside $\mathcal{Y}$, slowing down the convergence.

3 Proposed kernel and experimental results

Both $k_\mathcal{Y}$ and $k_\mathcal{X}$ suffering from limitations, it is desirable to have a kernel that retains as much as possible of the actual high dimensional distances between points while remaining of low dimension. This can be achieved by first projecting points orthogonally on the faces of the hypercube to the subspace spanned by $\mathbf{A}$: $\text{Ran}(\mathbf{A})$, with $p_\mathbf{A} : \mathcal{X} \mapsto \mathbb{R}^D$, $p_\mathbf{A}(\mathbf{x}) = \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{x}$. Note that these back-projections from the hypercube can be outside of $\mathcal{X}$. The calculation of the projection matrix is done only once, inverting a

$d \times d$ matrix. This solves the problem of adding already evaluated points: their back-projections coincide. Nevertheless, distant points on the sides of $\mathcal{X}$ from the convex projection can be back-projected close to each other, which may cause troubles with the stationary kernels classically used.

The next step is to respect as much as possible distances on the border of $\mathcal{X}$, denoted $\partial\mathcal{X}$. Unfolding and parametrizing the manifold corresponding to the convex projection of the embedding of $\mathcal{Y}$ with $\mathbf{A}$ would be best but unfortunately it seems intractable with high D. Indeed, it amounts to finding each intersection of the d-dimensional subspace spanned by $\mathbf{A}$ with the faces of the D-hypercube, before describing the parts resulting from the convex projection. Alternatively, we propose to distort the back-projections which are outside of $\mathcal{X}$, corresponding to those convex-projected parts on the sides of $\partial\mathcal{X}$. In more details, from the back-projection of the initial mapping with $p_{\mathcal{X}}$, a pivot point is selected as the intersection between $\partial\mathcal{X}$ and the line $(O; p_{\mathbf{A}}(p_{\mathcal{X}}(\mathbf{A}y))$. Then the back-projection is stretched out such that the distance between the pivot point and the initial convex projection are equal. It results in respecting the distance *on the embedding* between the center O and the initial convex projection. The resulting warping, denoted Ψ, is detailed in Algorithm 1 and illustrated in Figure 1. Based on this, any positive definite kernel k on $\mathcal{Y}$ can be used. For example, the resulting SE kernel is $k_\Psi(\mathbf{y}, \mathbf{y'}) = \exp\left(-\|\Psi(\mathbf{y}) - \Psi(\mathbf{y}')\|_D^2 / 2l_\Psi^2\right)$. Note that the function value corresponding to $\Psi(\mathbf{y})$ remains $g(\mathbf{y})$.

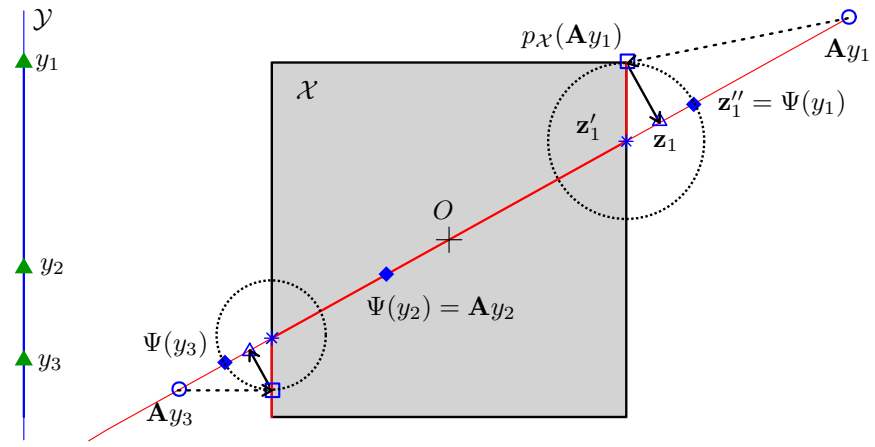

Figure 1: Illustration of the new warping Ψ , $d = 1$ and $D = 2$, from triangles in $\mathcal{Y}$ to diamonds in $\mathcal{X}$, on three points y_1, y_2, y_3. As for REMBO, the points y_i are first mapped by $\mathbf{A}$ and convexly projected onto $\mathcal{X}$ (if out of $\mathcal{X}$). If the resulting image is strictly contained in $\mathcal{X}$ – as for y_2 – nothing else is done. Otherwise, the new warping is defined in two supplementary steps: back-projection onto $\mathrm{Ran}(\mathbf{A})$ (giving $\mathbf{z}_i$) and stretching out in the resulting line $[0, \mathbf{z}_i)$ (red solid line) by reporting the distance between the intersection of $[0, \mathbf{z}_i]$ on the frontier of $\mathcal{X}$, $\mathbf{z}_i'$, and the initial convex projection $p_{\mathcal{X}}(\mathbf{A}y_i)$. The points y_1 and y_3 correspond to cases where such projections are on a corner or a face of $\mathcal{X}$.

Like $k_{\mathcal{X}}$, k_Ψ is not hindered by the non-injectivity brought by the convex projection $p_{\mathcal{X}}$. Furthermore, it can explore sides of the hypercube without spending too much budget since belonging to $\mathrm{Ran}(\mathbf{A})$ (all distances between embedded points after warping are d-dimensional instead of D-dimensional, thus smaller, hence limiting the risk of over-exploring sides of $\mathcal{X}$). It is thus possible to extend the size of $\mathcal{Y}$ to avoid the risk of

Algorithm 1 Calculation of Ψ.

Map $\mathbf{y} \in \mathcal{Y}$ to $\mathbf{Ay}$
If $\mathbf{Ay} \in \mathcal{X}$ **Then**
    Define $\Psi(\mathbf{y}) = \mathbf{Ay}$
Else
    Project onto $\mathcal{X}$ and back-project onto $\mathrm{Ran}(\mathbf{A})$: $\mathbf{z} = p_{\mathbf{A}}(p_{\mathcal{X}}(\mathbf{Ay}))$
    Compute the intersection of $[O; \mathbf{z}]$ with $\partial\mathcal{X}$: $\mathbf{z}' = (\max_{i=1,\dots,D} |z_i|)^{-1}\mathbf{z}$
    Define $\Psi(\mathbf{y}) = \mathbf{z}' + \|p_{\mathcal{X}}(\mathbf{A}y) - \mathbf{z}'\|_D \cdot \frac{\mathbf{z}'}{\|\mathbf{z}'\|_D}$
EndIf

missing the optimum. For instance, one can check that $\mathcal{Y}$ is larger than $[-\gamma, \gamma]^d$ with γ such that $\gamma^{-1} = \min_{j \in [1,\dots,D]} \sum_{i=1}^{d} |A_{j,i}|$, with $A_{j,i}$ the components of $\mathbf{A}$, ensuring to span $[-1, 1]$ for each of the D variables.

We compare the performances of the usual REMBO method with $k_{\mathcal{Y}}$, $k_{\mathcal{X}}$ and the proposed k_Ψ, with a unique embedding. Tests are conducted with the *DiceKriging* and *DiceOptim* packages [6]. We use the isotropic Matérn 5/2 kernel with hyperparameters estimated with Maximum Likelihood and we start optimization with space filling designs of size $10d$. Initial designs are modified such that no points are repeated in $\mathcal{X}$ for $k_{\mathcal{Y}}$ and $k_{\mathcal{X}}$. For k_Ψ, we apply Ψ to bigger initial designs before selecting the right number of points, as distant as possible between each other. Experiments are repeated fifty times, taking the same random embeddings for all kernels. To allow a fair comparison, $\mathcal{Y}$ is set to $[-\sqrt{d}, \sqrt{d}]^d$ for all kernels and the computational efforts on the maximization of the Expected Improvement are the same.

Results in Figure 2 show that the proposed kernel k_Ψ outperforms both $k_{\mathcal{Y}}$ and $k_{\mathcal{X}}$ when $d = 6$. In particular, $k_{\mathcal{Y}}$ loses many evaluations on the sides of $\mathcal{Y}$ for already known points in $\mathcal{X}$ and $k_{\mathcal{X}}$ has a propensity to explore sides of $\mathcal{X}$, while k_Ψ avoids both pitfalls.

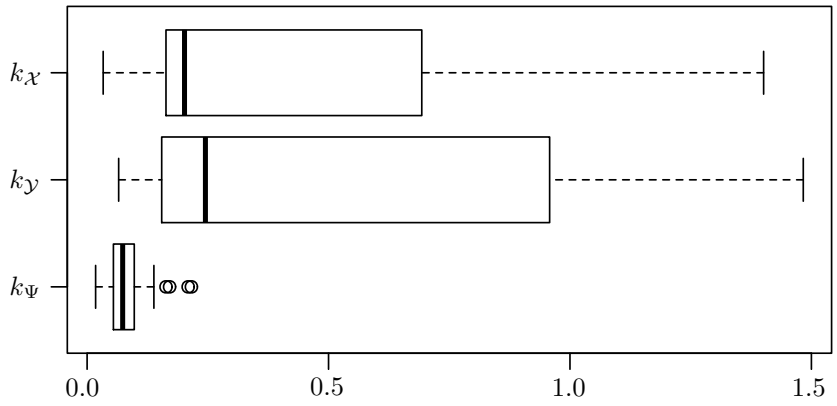

Figure 2: Boxplot of the optimality gap (best value found minus actual minimum) for kernels $k_{\mathcal{X}}$, $k_{\mathcal{Y}}$ and k_Ψ on the Hartmann6 test function (see e.g. [1]) with 250 evaluations, $d = d_e = 6$, $D = 25$.

4 Conclusion and perspectives

The composition with a warping of the covariance kernel used with REMBO wipes out some of the previous shortcomings. It thus achieved the goal of improving the results with a single embedding, as was shown on the Hartman6 example. Studying the efficiency of splitting the evaluation budget between several random embeddings, compared to relying on a single one along with k_Ψ, would be the scope of future research. Of interest is also the study of the embedding itself, such as properties ensuring fast convergence in practice.

Acknowledgments

This work has been conducted within the frame of the ReDice Consortium, gathering industrial (CEA, EDF, IFPEN, IRSN, Renault) and academic (Ecole des Mines de Saint-Etienne, INRIA, and the University of Bern) partners around advanced methods for Computer Experiments.
The authors also thanks the anonymous reviewers as well as Frank Hutter for their helpful suggestions.

References

[1] Jones, D., Schonlau, M., and Welch, W. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492.

[2] Koziel, S., Ciaurri, D. E., and Leifsson, L. (2011). Surrogate-based methods. In *Computational Optimization, Methods and Algorithms*, pages 33–59. Springer.

[3] Matheron, G. (1963). Principles of geostatistics. *Economic geology*, 58(8):1246–1266.

[4] Mockus, J., Tiesis, V., and Zilinskas, A. (1978). The application of bayesian methods for seeking the extremum. *Towards Global Optimization*, 2(117-129):2.

[5] Rasmussen, C. E. and Williams, C. (2006). *Gaussian Processes for Machine Learning*. MIT Press.

[6] Roustant, O., Ginsbourger, D., and Deville, Y. (2012). DiceKriging, DiceOptim: Two R packages for the analysis of computer experiments by kriging-based metamodeling and optimization. *Journal of Statistical Software*, 51(1):1–55.

[7] Snoek, J., Swersky, K., Zemel, R. S., and Adams, R. P. (2014). Input warping for Bayesian optimization of non-stationary functions. *In ICML*.

[8] Stein, M. L. (1999). *Interpolation of spatial data: some theory for kriging*. Springer.

[9] Wang, Z., Zoghi, M., Hutter, F., Matheson, D., and de Freitas, N. (2013). Bayesian optimization in high dimensions via random embeddings. *In IJCAI*.