Heteroscedastic Treed Bayesian Optimisation

John-Alexander M. Assael
Imperial College London
i.assael@imperial.ac.uk

Ziyu Wang
University of Oxford
ziyu.wang@cs.ox.ac.uk

Nando de Freitas
University of Oxford
CIFAR Fellow
nando@cs.ox.ac.uk

Abstract

We propose new hierarchical models and estimation techniques to solve the problem of heteroscedasticity in Bayesian optimisation. Our results demonstrate substantial gains in a wide range of applications, including automatic machine learning and mining exploration.

1 Introduction

The goal of Bayesian optimisation is to find the global optimum $\mathbf{x}^* = \arg\max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$ of a black-box function $f(\cdot) : \mathcal{X} \mapsto \mathbb{R}$ over an index set $\mathcal{X} \subset \mathbb{R}^d$. Bayesian optimisation may be understood in the setting of sequential decision making, whereby at the t-th decision round, we select an input $\mathbf{x}_t \in \mathcal{X}$ and observe the value of the black-box reward function $f(\mathbf{x}_t)$. The returned value y_t may be deterministic, $y_t = f(\mathbf{x}_t)$, or stochastic, $y_t = f(\mathbf{x}_t) + \epsilon_t$, where ϵ_t is a noise process.

Since the function is unknown, we use a Bayesian prior model to encode our beliefs about its smoothness, and an observation model to describe the data $\mathcal{D}_t = \{\mathbf{x}_{1:t}, \mathbf{y}_{1:t}\}$ up to the t-th round. Using these two models and the rules of probability, we derive a posterior distribution $p(f(\cdot)|\mathcal{D}_t)$ that can in turn be used to build an acquisition function to decide the next input query $\mathbf{x}_{t+1}$. The acquisition function trades-off exploitation and exploration in the search process. For a comprehensive introduction of Bayesian optimization, please refer to [1, 2].

Most functions encountered in practice, specially in automatic algorithm configuration, tend to be heteroscedastic. Snoek et al. [3] addressed this fundamental problem using input warped Gaussian processes. In this work, we introduce more flexible prior models for dealing with heteroscedasticity. In particular, we adopt trees with (warped) Gaussian process leaves. We explain how to construct these trees properly so as to avoid variance explosion near split points. We also introduce a hierarchical approach for estimating the hyper-parameters so as to address situations in which only a few points are observed at each leaf. All these methodological improvements, when combined, result in improved empirical performance in a wide range of applications to algorithm configuration and geophysics. The treed approach is particularly relevant to the latter application, where abrupt discontinuities arise.

2 Background and Related Work

2.1 Bayesian optimisation with Gaussian processes

Gaussian processes (GPs) are popular priors for Bayesian optimisation as they offer a simple and flexible way to capture our beliefs about the behaviour of the function; we refer the reader to [4] for details on these stochastic processes. These priors are defined by a mean function $m(\cdot)$ and a covariance kernel $k(\cdot, \cdot)$ on the index sets $\mathcal{X}$ and $\mathcal{X} \otimes \mathcal{X}$. Given any collection of inputs $\mathbf{x}_{1:t}$, the outputs are jointly Gaussian: $f(\mathbf{x}_{1:t})|\theta \sim \mathcal{N}(\mathbf{m}(\mathbf{x}_{1:t}), \mathbf{K}^\theta(\mathbf{x}_{1:t}, \mathbf{x}_{1:t}))$, where $\mathbf{m}(\mathbf{x}_{1:t})_i = m(\mathbf{x}_i)$ is the i-th entry of the mean vector and $\mathbf{K}^\theta(\mathbf{x}_{1:t}, \mathbf{x}_{1:t})_{ij} = k^\theta(\mathbf{x}_i, \mathbf{x}_j)$ is the ij-th entry of the

covariance matrix parametrised by θ. For convenience, we assume a zero-mean prior. The choice of covariance function is important as it governs the smoothness of the function. We adopt the Matérn(5/2) kernel with automatic relevance determination.

Given the observations $\mathcal{D}_t = \{\mathbf{x}_{1:t}, \mathbf{y}_{1:t}\}$, where $y_t = f(\mathbf{x}_t) + \epsilon_t$, the posterior predictive distribution of any evaluation point $\mathbf{x}$ is marginally Gaussian $f(\mathbf{x})|\mathcal{D}_t, \theta \sim \mathcal{N}(\mu_t(\mathbf{x};\theta), \sigma_t(\mathbf{x};\theta)^2)$, see [4] for the expanded expressions of these sufficient statistics.

Having specified a distribution to capture our beliefs about the behaviour of the function, we define an acquisition function $\alpha(\cdot|\mathcal{D}_t)$ for choosing the next evaluation point $\mathbf{x}_{t+1} = \arg\max_{\mathbf{x}\in\mathcal{X}} \alpha(\mathbf{x}|\mathcal{D}_t)$. The acquisition function must trade-off exploration and exploitation to ensure that the location of the global maximum (or minimum) is found in as few steps as possible.

Although many acquisition strategies have been proposed (see for example [5, 6, 7, 8, 2, 9, 10, 11]), the expected improvement (EI) criterion remains a default choice in popular Bayesian optimisation packages, such as SMAC and Spearmint [12, 2], and consequently we adopt it here.

Finally, several approaches have been proposed to overcome heteroscedasticity [13, 14, 15, 16, 17, 18, 19, 20]. Our approach is uniquely crafted for BO tasks, exploits all the information obtained and is able to work efficiently even with a few data points.

3 Treed Bayesian Optimisation

3.1 Constructing Gaussian process trees

Our proposed Heteroscedastic Treed Bayesian Optimisation (HTBO) method is based on CART, a decision tree model of Breiman et al. [21]. A Decision tree may be understood in terms of a sequence of binary tests applied to an input $\mathbf{x}$, which determines the path followed by $\mathbf{x}$ from the root of the tree to a leaf. Each node has a function of the form $h(\mathbf{x}) > \tau$, where h extracts a coordinate (feature) of $\mathbf{x}$ and compares it to a threshold τ. The tree is constructed in a recursive manner by choosing splits on features and thresholds so as to reduce uncertainty [22].

As suggested by the CART model, we can measure uncertainty in a node A using the empirical mean squared error: $\mathrm{U}(A) = \frac{1}{|A|}\sum_{y_i\in A}(\bar{y}_A - y_i)^2$, where $\bar{y}_A$ is the average of the output values in A. We could also use the entropy of the GPs in each node, but we found this alternative uncertainty measure to require much more computation without leading to better performance.

The optimal split thresholds are the ones that reduce uncertainty the most when splitting node A into $A'_{h,\tau}$ and $A''_{h,\tau}$. They are obtained by optimising the following reduction in uncertainty objective:

$$\mathrm{I}(A, A'_{h,\tau}, A''_{h,\tau}) = \mathrm{U}(A) - \frac{|A'_{h,\tau}|}{|A|}\mathrm{U}(A'_{h,\tau}) - \frac{|A''_{h,\tau}|}{|A|}\mathrm{U}(A''_{h,\tau}). \tag{1}$$

In CART, the splitting threshold τ of feature $h(\mathbf{x})$ is the midpoint of two points $(\mathbf{x}_i, \mathbf{x}_j)$, which is convenient for constant predictions as $\mathbf{x}_i$ will go to the left child and $\mathbf{x}_j$ to the right one respectively. However, in the proposed approach this would create unwanted variance in the gap between $\mathbf{x}_i$ and $\mathbf{x}_j$. This antagonises the goal of minimising the conditional variance in Bayesian Optimisation [23] as shown in the left plot of Figure 1. To overcome this difficulty, we place τ exactly at one of the points $\mathbf{x}_i$, and let $\mathbf{x}_i$ belong to both children nodes, as shown on the right hand side of Figure 1. This splitting strategy is essential for Bayesian optimisation to work well with treed GPs.

3.2 GP hyper-parameter optimisation

A common approach to estimate the hyper-parameters of GPs is to maximise the log-marginal-likelihood [4]. The simplest way to implement this strategy in our setting is to independently maximise the GP log-marginal-likelihoods in each leaf. This naive strategy is however bound to fail because some leaves have very few data points. To circumvent this difficulty, we need a way of aggregating information from different levels of the tree hierarchy.

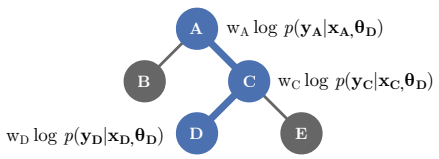
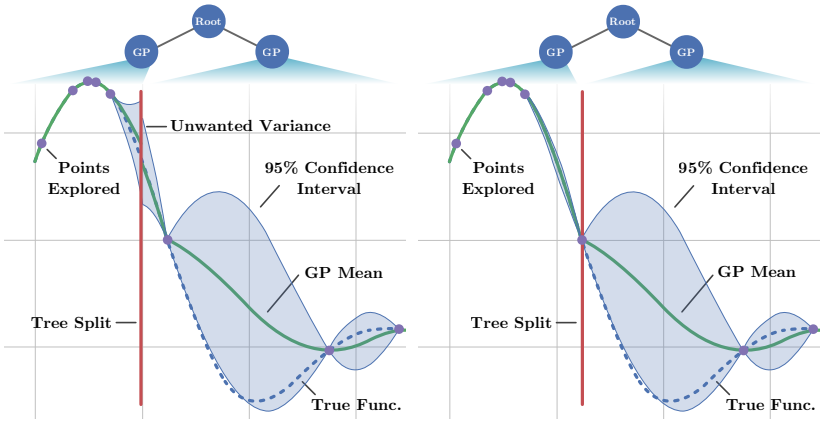


Figure 2: Tree-structured hyper-parameter estimation for the GP at leaf node D.



(a) CART Splitting (b) Proposed approach

Figure 1: Comparison between conventional CART splitting and our proposed splitting method.

To describe our solution to this problem, we need to introduce some notation. Let $\mathcal{M}_j(\mathcal{D})$ denote the set of data point pairs $(\mathbf{x}, y)$ that fall in node j, let $\tau(i) = \{t : (\mathbf{x}_t, y_t) \in \mathcal{M}_i(\mathcal{D})\}$ denote the data pairs in node i, let $path_j(\mathcal{D}_t)$ return the index of each node i in the path from the root node to a leaf node j, and finally let $depth_i(\mathcal{D}_t)$ return the depth of node i.

Suppose we are interested in estimating the hyper-parameters of the GP associated with the j-th leaf. Our solution is to maximise the sum of weighted log-marginal-likelihoods for nodes in the path from the root to the j-th leaf, as depicted in Figure 2. This hierarchical information aggregation can be cast as an optimisation problem, or by adopting a more Bayesian approach, it can be estimated by simply putting a prior $p(\theta)$ on the hyper-parameters and sampling from:

$$\sum_{i \in path_j(\mathcal{D}_t)} w_i \log p(\mathbf{y}_{\tau(i)} | \mathbf{x}_{\tau(i)}, \theta) + \log p(\theta), \tag{2}$$

using Markov chain Monte Carlo (MCMC), where

$$w_i = \frac{|\mathcal{M}_j(\mathcal{D}_t)|}{|\mathcal{M}_i(\mathcal{D}_t)|} \frac{1}{(1 + depth_j(\mathcal{D}_t) - depth_i(\mathcal{D}_t))}, \qquad i \in path_j(\mathcal{D}_t).$$

The first ratio in the weight expression is a normalisation factor ensuring that the weight at the leaf is equal to 1. The second ratio ensures that points closer to the leaf will have a higher influence in the estimate of the hyper-parameters for the GP associated with that leaf.

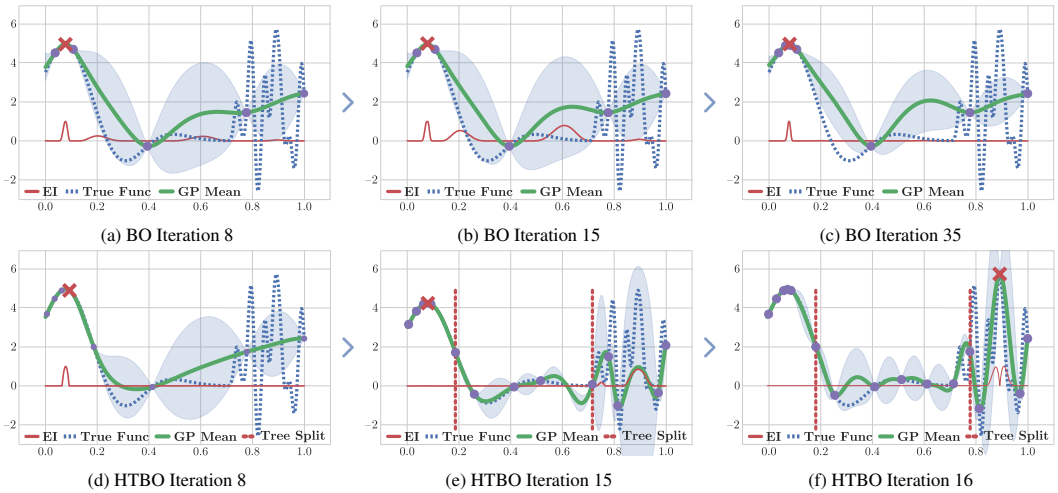
3.3 Putting it all together

At each iteration of Bayesian optimisation, the decision tree is reconstructed. In doing so, we ensure that there is a minimum number of data points per leaf (5 in our experiments). Subsequently, we estimate the hyper-parameters, the kernel amplitude θ_0, and the mean μ_t as discussed in the previous section. Once the GPs have been optimised for the data on each leaf, predictions are made employing all observations, and we use their statistics to construct the EI acquisition function. This function is then optimised using a d-dimensional adaptive Sobol Grid with $20,000$ points as in the package Spearmint.

Figure 3 compares a few iterations of the proposed treed approach against standard Bayesian optimisation on a one-dimensional heteroscedastic function. While the standard approach fails, the proposed method, hierarchical treed Bayesian optimisation (HTBO), is able to overcome non-stationarity to find the maximum of the objective function.

4 Experiments

We compare the proposed HTBO method and a hybrid approach (HTBO WARP), where the leaves of the decision tree use input warping and the parameters of the Beta CDF are estimated using the



(a) BO Iteration 8 (b) BO Iteration 15 (c) BO Iteration 35

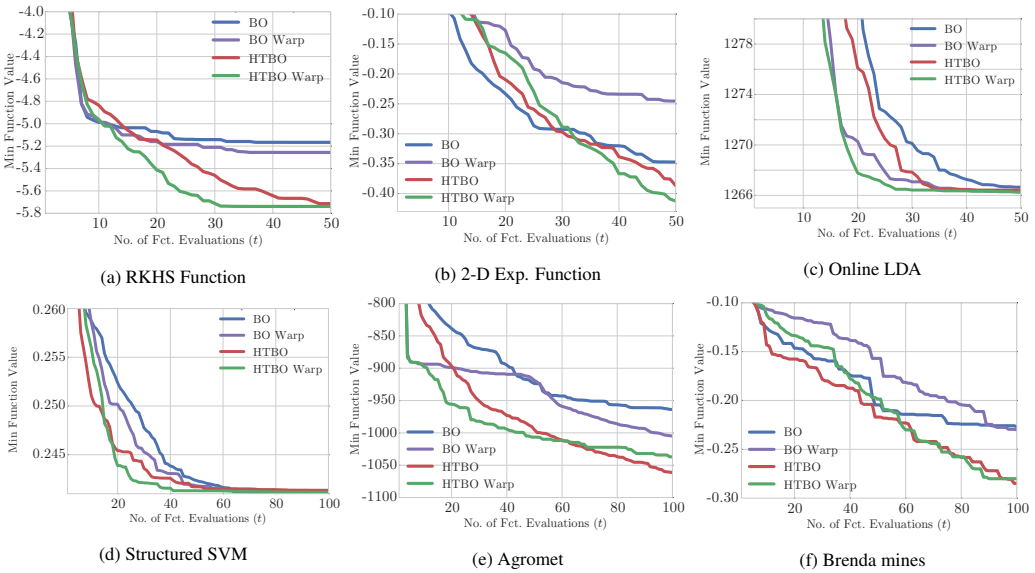
(d) HTBO Iteration 8 (e) HTBO Iteration 15 (f) HTBO Iteration 16

Figure 3: Comparison between standard BO (a-c) and the proposed HTBO method (d-f). HTBO is able to find the maximum in 16 iterations, while BO is not able to overcome heteroscedasticity and over-samples one of the local maxima.

proposed hierarchical model , against standard Bayesian optimisation (BO) and Bayesian optimisation with input warping (BO WARP) [3].

Our benchmarks include a one-dimensional synthetic example with a discontinuity (RKHS) [24], a synthetic example from R. Gramacy [25] where the objective function is flat over a large region (2-D Exp), two standard benchmarks from automatic machine learning (online LDA and structured SVM) and two mining exploration examples (Agromet and Brenda mines) from the kriging literature [26]. In the latter, we have the coordinates and depth of the explored regions as well as the function evaluation (e.g., amount of ore), and hence we only query EI at the available points.

Figure 4 summaries the results. Overwhelmingly the proposed approach leads to significant improvements over this wide range of test cases. The results also confirm that input warping is very useful, but that it is insufficient to handle some types of heteroscedasticity, where our proposed approaches outperform the competition and can yield both performance gains and robustness.



(a) RKHS Function (b) 2-D Exp. Function (c) Online LDA

(d) Structured SVM (e) Agromet (f) Brenda mines

Figure 4: Performance evaluation of BO, BO with input warping (BO Warp), and the proposed approaches (HTBO and HTBO Warp) on synthetic functions, algorithm configuration problems and mining problems.

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," Tech. Rep. UBC TR-2009-23 and arXiv:1012.2599v1, Dept. of Computer Science, University of British Columbia, 2009.

[2] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," in *NIPS*, pp. 2951–2959, 2012.

[3] J. Snoek, K. Swersky, R. Zemel, and R. R.P. Adams, "Input warping for Bayesian optimization of non-stationary functions," in *NIPS Workshop on Bayesian Optimization in Theory and Practice*, 2013.

[4] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[5] J. Močkus, "The Bayesian approach to global optimization," *Systems Modeling and Optimization*, vol. 38, pp. 473–481, 1982.

[6] D. Jones, "A taxonomy of global optimization methods based on response surfaces," *Journal of Global Optimization*, vol. 21, no. 4, pp. 345–383, 2001.

[7] M. Hoffman, E. Brochu, and N. de Freitas, "Portfolio allocation for Bayesian optimization," in *UAI*, pp. 327–336, 2011.

[8] P. Hennig and C. Schuler, "Entropy search for information-efficient global optimization," *JMLR*, vol. 13, pp. 1809–1837, 2012.

[9] M. Hoffman, B. Shahriari, and N. de Freitas, "On correlation and budget constraints in model-based bandit optimization with application to automatic machine learning," in *AIStats*, pp. 365–374, 2014.

[10] Z. Wang, B. Shakibi, L. Jin, and N. de Freitas, "Bayesian multi-scale optimistic optimization," in *AIStats*, pp. 1005–1014, 2014.

[11] B. Shahriari, Z. Wang, M. W. Hoffman, A. Bouchard-Cote, and N. de Freitas, "An Entropy Search Portfolio for Bayesian Optimization," Tech. Rep. arXiv:1406.4625, University of Oxford, 2014.

[12] F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Sequential model-based optimization for general algorithm configuration," in *LION*, pp. 507–523, 2011.

[13] P. D. Sampson and P. Guttorp, "Nonparametric estimation of nonstationary spatial covariance structure," *Journal of the American Statistical Association*, vol. 87, no. 417, pp. 108–119, 1992.

[14] A. M. Schmidt and A. O'Hagan, "Bayesian inference for non-stationary spatial covariance structure via spatial deformations," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 65, no. 3, pp. 743–758, 2003.

[15] L. Bornn, G. Shaddick, and J. V. Zidek, "Modeling nonstationary processes through dimension expansion," *Journal of the American Statistical Association*, vol. 107, no. 497, pp. 281–289, 2012.

[16] D. Higdon, J. Swall, and J. Kern, "Non-stationary spatial modeling," *Bayesian statistics*, vol. 6, no. 1, pp. 761–768, 1999.

[17] C. K. Williams and C. E. Rasmussen, "Gaussian processes for machine learning," *the MIT Press*, vol. 2, no. 3, p. 4, 2006.

[18] E. Snelson, C. E. Rasmussen, and Z. Ghahramani, "Warped gaussian processes," *NIPS*, vol. 16, pp. 337–344, 2004.

[19] R. P. Adams and O. Stegle, "Gaussian process product models for nonparametric nonstationarity," in *ICML*, pp. 1–8, 2008.

[20] D. B. Dunson and E. B. Fox, "Multiresolution gaussian processes," in *Advances in Neural Information Processing Systems*, pp. 737–745, 2012.

[21] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and regression trees*. CRC press, 1984.

[22] M. Denil, D. Matheson, and N. de Freitas, "Narrowing the gap: Random forests in theory and in practice," in *ICML*, 2014.

[23] E. Brochu, N. de Freitas, and A. Ghosh, "Active preference learning with discrete choice data," in *NIPS*, pp. 409–416, 2007.

[24] Z. Wang, J.-A. Assael, and N. de Freitas, *RKHS 1D Function for Bayesian Optimization tasks*, Oct. 2014. https://github.com/iassael/function-rkhs.

[25] R. B. Gramacy, *Bayesian treed Gaussian process models*. PhD thesis, University Of California Santa Cruz, 2005.

[26] I. Clark and W. V. Harper, *Practical Geostatistics 2000: Case Studies*. Ecosse North America, 2008.