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 \( f : \mathcal{X} \rightarrow \mathbb{R} \) over an index set \( \mathcal{X} \subseteq \mathbb{R}^d \). 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 \( y_t = f(\mathbf{x}_t) \), or stochastic, \( y_t = f(\mathbf{x}_t) + \epsilon_t \), where \( \epsilon_t \) 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 \( \mathcal{D}_t = \{ \mathbf{x}_{1:t}, y_{1:t} \} \) up to the \( t \)-th round. Using these two models and the rules of probability, we derive a posterior distribution \( p(f|\mathcal{D}_t) \) 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 [Brochu et al. 2009, Snoek et al. 2012].

Most functions encountered in practice, specially in automatic algorithm configuration, tend to be heteroscedastic. [Snoek et al. 2013] addressed this fundamental problem using 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

In this section, we give a brief overview of Bayesian optimisation as well as a brief survey on Heteroscedastic Gaussian processes.

2.1 Bayesian optimisation with GPs

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 [Rasmussen and Williams 2006] 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 \( \mathcal{X} \times \mathcal{X} \). 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 \( \mathbf{m}(\mathbf{x}_{1:t}) = m(\mathbf{x}_i) \) 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 \( \theta \). For convenience, we assume constant prior mean functions. The choice of covariance function is important as it governs the smoothness of the function. While Gaussian kernels are popular, we opt for the Matérn(5/2) kernel with automatic relevance determination:
\[ k^\theta_{\text{Matérn}(5/2)}(\mathbf{x}, \mathbf{x}') = \exp(-\sqrt{5}r)(1 + \sqrt{5}r + \frac{5}{3}r^2) \]
Heteroscedastic Treed Bayesian Optimisation

where

\[ r = (x - x')^\top \text{diag}(\theta^2)^{-1}(x - x'). \]

The Matérn(5/2) kernel is parametrised by the amplitude hyper-parameter \( \theta_0 \), and \( d \) length-scale hyper-parameters \( \theta_d \). It makes less stringent smoothness assumptions than the Gaussian kernel, and is thus a better fit for heteroscedastic Bayesian optimisation.

Given the observations \( D_t = \{x_{1:t}, y_{1:t}\} \), where \( y_t = f(x_t) + \epsilon_t \), the joint distribution of these observations and an arbitrary evaluation point \( x \) is:

\[
\begin{bmatrix}
  y_{1:t} \\
  f(x)
\end{bmatrix} \sim \mathcal{N}
\begin{bmatrix}
  m(x_{1:t}) \\
  m(x)
\end{bmatrix}, \begin{bmatrix}
  K^\theta_t + \sigma^2 I & K^\theta_t(x) \\
  K^\theta_t(x)^\top & k^\theta(x, x)
\end{bmatrix},
\]

where \( K^\theta_t = K^\theta(x_{1:t}, x_{1:t}) \) and \( k^\theta(x) = k^\theta(x_{1:t}, x) \).

The posterior predictive distribution of any evaluation point \( x \) is marginally Gaussian \( f(x) | D_t, \theta \sim \mathcal{N}(\mu_t(x; \theta), \sigma_t(x; \theta)^2) \), where

\[
\begin{align*}
\mu_t(x; \theta) &= \mathbb{E}[f(x)|D_t] = m(x) + k^\theta_t(x) \mathbb{K}^{-1}(y_{1:t} - m(x_{1:t})), \\
k^\theta_t(x, x') &= \mathbb{E}[f(x)f(x')|D_t] = k^\theta(x, x') - k^\theta_t(x) \mathbb{K}^{-1}k^\theta_t(x'), \\
\sigma_t(x; \theta)^2 &= k^\theta_t(x, x).
\end{align*}
\]

Having specified a distribution to capture our beliefs about the behaviour of the function, as well as a mechanism to update it at each step, we define an acquisition function \( \alpha(\cdot|D_t) \) for choosing the next evaluation point

\[ x_{t+1} = \arg \max_{x \in X} \alpha(x|D_t). \]

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, Moˇckus (1982); Jones (2001); Hoffman et al. (2011); Hennig and Schuler (2012); Snoek et al. (2012); Hoffman et al. (2014); Wang et al. (2014a); Shahriari et al. (2014)), the expected improvement (EI) criterion remains a default choice in popular Bayesian optimisation packages, such as SMAC and Spearmint (Hutter et al. 2011; Snoek et al. 2012). If we let

\[ x^+_t = \arg \max_{i \leq t} f(x_i; \theta) \]

denote the current incumbent, the EI acquisition function can be written in closed form as

\[
\alpha^\text{EI}(f)(x|D_t) = \mathbb{E}[\max\{0, f(x) - f(x^+)|D_t\}
= \sigma_t(x; \theta)[a\Phi(a) + \phi(a)]
\]

with

\[ a = \frac{\mu_t(x; \theta) - f(x^+)}{\sigma_t(x; \theta)}, \]

and \( \phi \) and \( \Phi \) are the standard normal density and distribution functions respectively. In the special case of \( \sigma_t(x; \theta) = 0 \), we set \( \alpha^\text{EI}(f)(x|D_t) = 0 \). The expected improvement is best understood as a family of one-step-decision heuristics [Brochu et al. (2009)], with many members in this family.

2.2 Heteroscedastic Gaussian processes

Many approaches have been proposed to manage heteroscedasticity with Gaussian processes. Sampson and Guttorp (1992) attempted to project inputs into a latent space that is stationary. This approach was later extended by Schmidt and O’Hagan (2003). A latent space representation in higher dimensions was also proposed by Borm et al. (2012). Others such as Higdon et al. (1999) and Williams and Rasmussen (2006) have tried to model heteroscedasticity directly with the choice of covariance function. In 2005, Gramacy proposed a treed GP model to attack non-stationarity. While this work is the closest to ours, treed GPs were developed for modelling functions and not for global optimisation.

Warping is another popular approach for dealing with non-stationarity (Snelson et al. 2004; Adams and Stegle 2008). Recently, Snoek et al. (2013) proposed an input warping Bayesian optimisation strategy, using a parametrised Beta CDF as the warping function. The goal of input warping is to transform non-stationary functions to stationary ones by applying a Beta CDF mapping \( w(.) \) to each dimension \( d \). The new covariance becomes \( \kappa(w(x), w(x')) \). We have found that input warping can lead to remarkable improvements in automatic algorithm configuration. However, the Beta CDF transformation has limitations, which we hope to address in this paper by using a treed approach.

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. (1984). A Decision tree may be understood in terms of a sequence of binary tests applied to an input \( x \), which determines the path followed by \( x \) from the root of the tree to a leaf. Each node has a function of the form \( h(x) > \tau \), where \( h \) extracts a coordinate (feature) of \( x \) and compares it to a threshold \( \tau \). The tree is constructed in a recursive manner by choosing splits on features and thresholds so as to reduce uncertainty [Denil et al. 2014].
We can measure uncertainty in a node $A$ using the empirical mean squared error:

$$U(A) = \frac{1}{|A|} \sum_{y_i \in A} (\bar{y}_A - y_i)^2,$$

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 splits on features and 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:

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

In CART, the splitting threshold $\tau$ of feature $h(x)$ is the midpoint of two points $(x_i, x_j)$, which is convenient for constant predictions as $x_i$ will go to the left child and $x_j$ to the right one respectively. However, in the proposed approach this would create unwanted variance in the gap between $x_i$ and $x_j$, as one GP will have to cover the unknown space from $x_i$ to $\tau$ and another GP the unknown space from $\tau$ to $x_j$. This antagonises the goal of minimising the conditional variance in Bayesian Optimisation \cite{Brochu2010} as shown in the left plot of Figure 1.

To overcome this difficulty, we place $\tau$ exactly at one of the points $x_i$, and let $x_i$ 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 Treating GP hyper-parameters

A common approach to estimate the hyper-parameters of GPs is to maximise the log-marginal-likelihood, given by

$$\log p(y_{1:t} | x_{1:t}, \theta) = -\frac{1}{2} y_{1:t}^T (K_\theta + \sigma^2 I)^{-1} y_{1:t}$$
$$- \frac{1}{2} \log |K_\theta + \sigma^2 I| - \frac{t}{2} \log(2\pi).$$  \hspace{1cm} (2)

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.

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 $(x, y)$ that fall in node $j$, let $\gamma(i) = \{ t : (x_{t+1}, y_t) \in \mathcal{M}_j(\mathcal{D}) \}$ denote the data pairs in node $i$, let $\text{path}_{ij}(\mathcal{D})$ return the index of each node $i$ in the path from the root node to a leaf node $j$, and finally let $\text{depth}_{i}(\mathcal{D})$ 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 approach can be cast in terms of the following optimisation problem:

$$\arg\max_{\theta} \sum_{i \in \text{path}_j(D)} w_i \log p(y_{\tau(i)}|x_{\tau(i)}, \theta),$$

(3)

where $i \in \text{path}_j(D)$ and

$$w_i = \frac{|M_j(D_i)|}{|\mathcal{M}_i(D_t)|} \frac{1}{(1 + \text{depth}_j(D_t) - \text{depth}_i(D_t))}.$$  

(4)

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. So far, for the ease of presentation, we have focused our attention on maximising the marginal likelihood. It is straightforward, however, to adopt a more Bayesian approach. If we are interested in estimating the hyper-parameters of the $j$-th leaf, we could simply put a prior $p(\theta)$ on the hyper-parameters and sample from

$$\sum_{i \in \text{path}_j(D)} w_i \log p(y_{\tau(i)}|x_{\tau(i)}, \theta) + \log p(\theta)$$

(5)

using Markov chain Monte Carlo (MCMC).

### 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 as discussed in the previous section. The same approach is also used in the estimation of the mean $\mu_t$ and the kernel amplitude $\theta_0$ of the GP. Once the GPs have been fit to the data in each leaf, 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.

It is interesting to note that, HTBO does not place explicit requirements on its GP leaves so long as we can compute the log-marginal-likelihood and EI efficiently. As a result, not only could we use standard GPs but also GPs with warping (Snoek et al. 2013) as leaves. When using warping, the $\alpha$ and $\beta$ parameters of the Beta CDF are estimated using the proposed hierarchical prior. We refer to HTBO with Warping as HTBO Warp and evaluate it in our experiments.

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

In this section we evaluate the proposed methods HTBO and HTBO Warp. Comparisons are made against standard Bayesian optimisation (BO) as well as the Bayesian optimisation approach with input warping (BO Warp) of Snoek et al. (2013). For all the experiments presented in this paper, we used the Matérn(5/2) kernel. The GP hyper-parameters of all four approaches are estimated using slice sampling.

We use three different sets of experiments for evaluation: synthetic functions, algorithm configuration benchmarks, and mineral exploration datasets. The results are summarised and discussed in Section 4.4.

#### 4.1 Synthetic problems

We first introduce two heteroscedastic synthetic functions. The first synthetic function, which we refer to as RKHS, is shown in Figure 3. The function is constructed as a weighted sum of squared exponential kernel functions with 2 different length scales. The left hand side of the function is smooth whereas the right hand size jagged. For a more detailed description as well as the code of the function, please refer to Wang et al. (2014b).

The second synthetic function is a two-dimensional exponential function from Gramacy (2005). The precise mathematical expression for the function is:

$$f(x_1, x_2) = x_1 \exp(-x_1^2 - x_2^2).$$

We refer to this function as 2-D Exp, and plot it in Figure 4. This function is interesting because it is “flat” over most of its domain with a peak that can be easily missed without careful exploration.
4.2 Automatic machine learning problems

Latent Dirichlet Allocation (LDA) is a directed graphical model for documents used in topic modelling tasks, for which [Hoffman et al., 2010] proposed an online learning approach in the variational Bayes paradigm. In this experiment we use precomputed performance data for this on-line LDA algorithm on a dataset of 250,000 Wikipedia articles under many parameter settings. Tuning the online LDA algorithm involves choosing the 2 learning parameters, $\tau_0$ and $\kappa$ as well as a third parameter specifying the mini-batch size, yielding a three-dimensional problem. Following the original authors, the search space is restricted to a $6 \times 6 \times 8$ grid (Hoffman et al., 2010).

Next, we optimize a latent structured support vector machine (SVM) using a dataset available from Snoek et al. (2012). As in the LDA tuning experiment, the authors consider the latent structured SVM on a three-dimensional grid of parameter settings which include two regularisation parameters and a convergence tolerance. We used data corresponding to the precomputed performance of this algorithm on binary classification of protein DNA sequences (Yu and Joachims, 2009; Miller et al., 2012). The three-dimensional search space is discretised yielding a $25 \times 14 \times 4$ grid, once again following the original methodology.

Both datasets serve as benchmarks in the algorithm configuration community and the performance of a few different global optimization approaches on these datasets is publicly available from Eggensperger et al. (2013).

4.3 Geostatistical problems

In geostatistics, Kriging is a method of interpolation with the aim of modelling a function efficiently with a minimal number of observations. Hence, Kriging is very closely related to Bayesian Optimisation. In this subsection, we describe two datasets available at kriging.com.

The Agromet dataset was acquired by Isobel Clark and describes a square area in the Natal Highlands, South Africa, measuring Gold grade of a drill hole intersection at 400-meter spacing. Hence, this example is two-dimensional and is defined by the latitude and the longitude values of each acquired sample. There are in total 18,189 observations. As shown in Figure 6 the surface described by the dataset is clearly heteroscedastic and very rough, making it difficult for a single Gaussian Process, even with input warping, to model accurately.
Brenda is a dataset of 1,856 observations of the depths of a copper mine in British Columbia, Canada, that was closed in 1990. Each observation in the dataset measures the concentrations of Copper, Molybdenum, Silver, and Gold deposits. As in the case of Agromet, the surface described by Brenda also exhibits non-stationarity. In our experiments, we try only to identify the areas that contain the highest concentrations of Copper.

Please refer to Clark and Harper (2008) for more comprehensive descriptions of these datasets. In this paper, we use these two Kriging examples to construct optimisation problems by trying to find the highest concentrations of ores. We only query the expected improvement function at the points available from the
4.4 Evaluation

Each of the approaches were run 32 times on all of the six benchmarks. Figure 5 summarises the median performance of all the runs. In all examples, we try to minimize (instead of maximize) the objective function to follow the convention of earlier work Snoek et al. (2012, 2013). We also report the mean and standard deviation of the runs in Table 1.

In the experiment involving the RKHS function (Figure 5(a)), most of the BO and BO WARP runs failed to converge to the global optimum. In contrast, by taking advantage of the tree partitioning, the proposed HTBO and HTBO WARP approaches are capable of modelling and optimising the heteroscedastic objective function. Specifically, HTBO WARP converged to the global optimum of the function in approximately 30 evaluations.

As illustrated in Figure 5(b), BO WARP again performs poorly on the 2D Exp function and HTBO and HTBO WARP exhibit the fastest convergence. This illustrates the fact that input warping, despite being a very powerful technique, can fail in some heteroscedastic domains.

As shown in Figure 5(c), on LDA, the best performance was achieved by HTBO WARP, while BO WARP catches up after about 10 evaluations. A similar behaviour is observed on the structured SVM example (Figure 5(d)). In both cases, HTBO converges faster than standard BO. In the SVM case, HTBO also converges faster than standard BO. It appears that both problems are simple enough that all four methods converge eventually. It is important to note that despite the lack of heteroscedasticity, the proposed methods are still competitive with the state of art.

BO has the worst performance out of the four methods evaluated on the Kriging examples as it does not deal with heteroscedasticity (Figure 5(e) (f)). BO WARP strategy performs better. As in the synthetic experiments, however, BO WARP is not as efficient as the HTBO WARP and HTBO. With the tree structure, HTBO as well as HTBO WARP achieve a significantly faster rate of convergence.

Table 1: The mean and standard deviation and the end of the scheduled iterations with the best results in bold. In most experiments, the proposed method HTBO WARP achieved the best performance, only with HTBO achieving better results on Brenda. HTBO also performed competitively in all experiments.

| Method    | BO     | BO Warp | HTBO   | HTBO Warp |
|-----------|--------|---------|--------|-----------|
| RKHS      | -5.16 ± 0.33 | -5.25 ± 0.37 | -5.71 ± 0.13 | -5.73 ± 0.00 |
| 2-D Exp   | -0.34 ± 0.16 | -0.24 ± 0.20 | -0.38 ± 0.11 | -0.41 ± 0.07 |
| LDA       | 1266.65 ± 1.33 | 1266.26 ± 0.30 | 1266.42 ± 0.44 | 1266.26 ± 0.30 |
| SVM       | 0.24 ± 0.00 | 0.24 ± 0.00 | 0.24 ± 0.00 | 0.24 ± 0.00 |
| Agromet   | -0.22 ± 0.10 | -0.23 ± 0.12 | -0.28 ± 0.13 | -0.28 ± 0.13 |
| Brenda    | -963.68 ± 78.0 | -1004.81 ± 15.6 | -1061.31 ± 61.7 | -1037.13 ± 43.5 |

5 Conclusion

In this work, we introduced two models based on decision trees with GP leaves for dealing with hard heteroscedastic functions in Bayesian optimisation. We proposed hierarchical priors for estimating the hyperparameters of the GP leaves. We demonstrated empirically, that our proposed methodological improvements have robust behaviour across a wide range of heteroscedastic functions. Moreover, after evaluating the performance of four different approaches in six problems, we showed that the two proposed approaches outperform the competition and can yield both performance gains and robustness.

References

Adams, R. P. and Stegle, O. (2008). Gaussian process product models for nonparametric nonstationarity. In ICML, pages 1–8.

Bornn, L., Shaddick, G., and Zidek, J. V. (2012). Modeling nonstationary processes through dimension ex-
pansion. *Journal of the American Statistical Association*, **107**(497), 281–289.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

Brochu, E., de Freitas, N., and Ghosh, A. (2007). Active preference learning with discrete choice data. In *NIPS*, pages 409–416.

Brochu, E., Cora, V. M., and de Freitas, N. (2009). A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. Technical Report UBC TR-2009-23 and arXiv:1012.2599v1, Dept. of Computer Science, University of British Columbia.

Clark, I. and Harper, W. V. (2008). *Practical Geostatistics 2000: Case Studies*. Ecosse North America.

Denil, M., Matheson, D., and de Freitas, N. (2014). Narrowing the gap: Random forests in theory and in practice. In *ICML*.

Eggensperger, K., Feurer, M., Hutter, F., Bergstra, J., Snoek, J., Hoos, H., and Leyton-Brown, K. (2013). Towards an empirical foundation for assessing Bayesian optimization of hyperparameters. In *NIPS Workshop on Bayesian Optimization in Theory and Practice*.

Gramacy, R. B. (2005). *Bayesian treed Gaussian process models*. Ph.D. thesis, University Of California Santa Cruz.

Hennig, P. and Schuler, C. (2012). Entropy search for information-efficient global optimization. *JMLR*, **13**, 1809–1837.

Higdon, D., Swall, J., and Kern, J. (1999). Non-stationary spatial modeling. *Bayesian statistics*, **6**(1), 761–768.

Hoffman, M., Bach, F. R., and Blei, D. M. (2010). Online learning for latent Dirichlet allocation. In *NIPS*, pages 856–864.

Hoffman, M., Brochu, E., and de Freitas, N. (2011). Portfolio allocation for Bayesian optimization. In *UAI*, pages 327–336.

Hoffman, M., Shahriari, B., and de Freitas, N. (2014). On correlation and budget constraints in model-based bandit optimization with application to automatic machine learning. In *AISTats*, pages 365–374.

Hutter, F., Hoos, H. H., and Leyton-Brown, K. (2011). Sequential model-based optimization for general algorithm configuration. In *LION*, pages 507–523.

Jones, D. (2001). A taxonomy of global optimization methods based on response surfaces. *Journal of Global Optimization*, **21**(4), 345–383.

Miller, K., Kumar, M. P., Packer, B., Goodman, D., Koller, D., et al. (2012). Max-margin min-entropy models. In *AISTats*.

Močkus, J. (1982). The Bayesian approach to global optimization. *Systems Modeling and Optimization*, **38**, 473–481.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. The MIT Press.

Sampson, P. D. and Guttorp, P. (1992). Nonparametric estimation of nonstationary spatial covariance structure. *Journal of the American Statistical Association*, **87**(417), 108–119.

Schmidt, A. M. and O’Hagan, A. (2003). Bayesian inference for non-stationary spatial covariance structure via spatial deformations. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **65**(3), 743–758.

Shahriari, B., Wang, Z., Hoffman, M. W., Bouchard-Cote, A., and de Freitas, N. (2014). An Entropy Search Portfolio for Bayesian Optimization. Technical Report arXiv:1406.4625, University of Oxford.

Snelson, E., Rasmussen, C. E., and Ghahramani, Z. (2004). Warped gaussian processes. *NIPS*, **16**, 337–344.

Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. In *NIPS*, pages 2951–2959.

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

Wang, Z., Shakibi, B., Jin, L., and de Freitas, N. (2014a). Bayesian multi-scale optimistic optimization. In *AISTats*, pages 1005–1014.

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

Williams, C. K. and Rasmussen, C. E. (2006). Gaussian processes for machine learning. *the MIT Press*, **2**(3), 4.

Yu, C.-N. J. and Joachims, T. (2009). Learning structural SVMs with latent variables. In *ICML*, pages 1169–1176. ACM.