Predictive Entropy Search for Efficient Global Optimization of Black-box Functions

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

Matthew W. Hoffman
mwh30@cam.ac.uk
University of Cambridge

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

Abstract

We propose a novel information-theoretic approach for Bayesian optimization called Predictive Entropy Search (PES). At each iteration, PES queries the point maximizing the expected information gain with respect to the the global maximum. PES relies on a reformulation of the expected reduction in differential entropy that allows us to obtain approximations that are both more accurate and efficient than other alternatives such as Entropy Search (ES). Furthermore, PES can easily perform a fully Bayesian treatment of the model hyperparameters while ES cannot. We show that the increased accuracy of PES leads to significant gains in optimization performance.

Note: this is a greatly shortened version of [5]; for further details, experiments, and discussion please refer to the longer work.

1 Introduction

In this work we are interested in finding the global maximizer $\mathbf{x}_\star = \arg\max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$ of a function f over some bounded domain, typically $\mathcal{X} \subset \mathbb{R}^d$. We assume that $f(\mathbf{x})$ can only be evaluated via queries to a black-box that provides noisy outputs of the form $y_i \sim \mathcal{N}(f(\mathbf{x}_i), \sigma^2)$. We note, however, that our framework can be extended to other non-Gaussian likelihoods. In this setting, we describe a sequential search algorithm that, after n iterations, proposes to evaluate f at some location $\mathbf{x}_{n+1}$. To make this decision the algorithm conditions on all previous observations $\mathcal{D}_n = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$. After N iterations the algorithm makes a final recommendation $\widetilde{\mathbf{x}}_N$ for the global maximizer of the latent function f. See [2] for a detailed tutorial.

We take a Bayesian approach to the problem described above and use a probabilistic model for the latent function f to guide the search and to select $\widetilde{\mathbf{x}}_N$. In this work we use a zero-mean Gaussian process (GP) prior for f. This prior is specified by a positive-definite kernel function $k(\mathbf{x}, \mathbf{x}')$. Given any finite collection of points $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, the values of f at these points are jointly zero-mean Gaussian with covariance matrix $\mathbf{K}_n$, where $[\mathbf{K}_n]_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. For the Gaussian likelihood described above, the vector of concatenated observations $\mathbf{y}_n$ is also jointly Gaussian with zero-mean. Therefore, at any location $\mathbf{x}$, the latent function $f(\mathbf{x})$ conditioned on past observations $\mathcal{D}_n$ is then Gaussian with marginal mean $\mu_n(\mathbf{x})$ and variance $v_n(\mathbf{x})$. See [12] for a more detailed derivation.

We follow an information-theoretic approach for active data collection as described in [9]. In [14] the authors propose an entropy-reduction technique for the optimization problem addressed in this paper, however their approach requires the evaluation of the expected posterior information gain over a grid in the input space. The work of [4] requires no such grid, but instead relies on a difficult-to-evaluate approximation. In this paper, we derive a rearrangement of this information-based acquisition function which leads to a more straightforward approximation that we call Predictive Entropy Search (PES). We also show empirically that our approximation is more accurate than that of [4], resulting in real performance gains.

2 Predictive entropy search

Our exploration strategy relies on selecting the point $\mathbf{x}_{n+1}$ which maximizes the expected reduction in differential entropy. The corresponding acquisition function can be written as

$$\alpha_n(\mathbf{x}) = \mathrm{H}[p(\mathbf{x}_\star|\mathcal{D}_n)] - \mathbb{E}_{p(y|\mathcal{D}_n,\mathbf{x})}[\mathrm{H}[p(\mathbf{x}_\star|\mathcal{D}_n \cup \{(\mathbf{x}, y)\})]]\,, \tag{1}$$

where $\mathrm{H}[p(\mathbf{x})] = -\int p(\mathbf{x}) \log p(\mathbf{x}) d\mathbf{x}$ represents the differential entropy of its argument and the expectation above is taken with respect to the posterior predictive distribution of y given $\mathbf{x}$. The exact evaluation of (1) is infeasible in practice. The main difficulties are i) $p(\mathbf{x}_\star|\mathcal{D}_n \cup \{(\mathbf{x}, y)\})$ must be computed for many different values of $\mathbf{x}$ and y during the optimization of (1) and ii) the entropy computations themselves are not analytic. To avoid this, we follow the approach described in [6] by noting that (1) can be equivalently written as the mutual information between $\mathbf{x}_\star$ and y given $\mathcal{D}_n$. Since the mutual information is a symmetric function, $\alpha_n(\mathbf{x})$ can be rewritten as

$$\alpha_n(\mathbf{x}) = \mathrm{H}[p(y|\mathcal{D}_n, \mathbf{x})] - \mathbb{E}_{p(\mathbf{x}_\star|\mathcal{D}_n)}[\mathrm{H}[p(y|\mathcal{D}_n, \mathbf{x}, \mathbf{x}_\star)]]\,, \tag{2}$$

where $p(y|\mathcal{D}_n, \mathbf{x}, \mathbf{x}_\star)$ is the posterior predictive distribution for y given the observed data $\mathcal{D}_n$ and the location of the global maximizer of f. Note that, unlike the previous formulation, this objective is based on the entropies of predictive distributions, which are analytic or can be easily approximated, rather than on the entropies of distributions on $\mathbf{x}_\star$ whose approximation is more challenging.

The first term in (2) is the entropy of a Gaussian and as a result can be computed analytically. The second term can be approximated by first performing a Monte Carlo average over M samples of the optimizer $\mathbf{x}_\star^{(i)}$ and then approximating $p(y|\mathcal{D}_n, \mathbf{x}, \mathbf{x}_\star)$ by a Gaussian with variance $v_n(\mathbf{x}|\mathbf{x}_\star^{(i)})$. Given these approximations the resulting acquisition function can be written as

$$\alpha_n(\mathbf{x}) \approx \tfrac{1}{M} \textstyle\sum_{i=1}^{M} \left\{0.5 \log[v_n(\mathbf{x}) + \sigma^2] - 0.5 \log[v_n(\mathbf{x}|\mathbf{x}_\star^{(i)}) + \sigma^2]\right\}. \tag{3}$$

In the next two sub-sections we will detail these two approximations. Finally, although we will not describe this process here, this acquisition function can further be marginalized over kernel hyperparameters. This is a significant advantage with respect to other methods that optimize the same information-theoretic acquisition function but do not marginalize over the hyper-parameters, as we will show in our experiments.

2.1 Sampling from the posterior over global maxima

In order to implement PES we must first approximately sample from the conditional distribution of the global maximizer $\mathbf{x}_\star$ given the observed data $\mathcal{D}_n$, i.e. from

$$p(\mathbf{x}_\star|\mathcal{D}_n) = p\big(f(\mathbf{x}_\star) = \max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) \big| \mathcal{D}_n\big)\,. \tag{4}$$

In order to do so we will first sample an analytic approximation to f and optimize the resulting sample. Our approximation can be obtained via Bochner's theorem [1] which states that any shift-invariant kernel k can be written as the Fourier transform of its spectral density $s(\mathbf{w})$. Letting $p(\mathbf{w}) = s(\mathbf{w})/\alpha$ be the associated normalized density, we can write the kernel as the expectation

$$k(\mathbf{x}, \mathbf{x}') = \alpha\, \mathbb{E}_{p(\mathbf{w})}[e^{-i\mathbf{w}^\mathsf{T}(\mathbf{x}-\mathbf{x}')}] = 2\alpha\, \mathbb{E}_{p(\mathbf{w},b)}[\cos(\mathbf{w}^\mathsf{T}\mathbf{x} + b)\cos(\mathbf{w}^\mathsf{T}\mathbf{x}' + b)]\,, \tag{5}$$

where $b \sim \mathcal{U}[0, 2\pi]$. Letting $\boldsymbol{\phi}(\mathbf{x}) = \sqrt{2\alpha/m}\cos(\mathbf{W}\mathbf{x} + \mathbf{b})$ be an m-dimensional feature map consisting of m samples from $p(\mathbf{w}, b)$ we can then approximate the kernel with $k(\mathbf{x}, \mathbf{x}') \approx \boldsymbol{\phi}(\mathbf{x})^\mathsf{T}\boldsymbol{\phi}(\mathbf{x}')$ as in [11]. The feature mapping $\boldsymbol{\phi}(\mathbf{x})$ allows us to approximate the Gaussian process posterior for f as a linear model $f(\mathbf{x}) = \boldsymbol{\phi}(\mathbf{x})^\mathsf{T}\boldsymbol{\theta}$ where the posterior for $\boldsymbol{\theta}$ is a simple Gaussian which we can sample from directly.

2.2 Approximating the predictive entropy

The next step in implementing PES involves approximating $\mathrm{H}[p(y|\mathcal{D}_n, \mathbf{x}, \mathbf{x}_\star)]$ from (2). To do so we will construct a Gaussian approximation to $p(y|\mathcal{D}_n, \mathbf{x}, \mathbf{x}_\star)$ which approximately enforces the constraints

C1. $\nabla f(\mathbf{x}_\star) = 0$ and $\text{upper}[\nabla^2 f(\mathbf{x}_\star)] = 0$;

C2. $\text{diag}[\nabla^2 f(\mathbf{x}_\star)] < 0$, which together with C1 approximately ensures that $\mathbf{x}_\star$ is a local maximizer, and $f(\mathbf{x}_\star) > \max_{i \leq n} f(\mathbf{x}_i)$, i.e. that $\mathbf{x}_\star$ is greater than past observations; and

C3. $f(\mathbf{x}_\star) > f(\mathbf{x})$, i.e. that $\mathbf{x}_\star$ is greater than subsequent queries.

Consider the latent variable $\mathbf{z} = [f(\mathbf{x}_\star); \text{diag}[\nabla^2 f(\mathbf{x}_\star)]]$. We can compute the posterior distribution $p(\mathbf{z}|D_n, \text{C1})$ by noting that the covariance between inputs and gradients can be determined by differentiating the kernel function [13]. The resulting posterior is a multivariate Gaussian with mean $\mathbf{m}_0$ and covariance $\mathbf{V}_0$. Constraints C2 can then be incorporated by writing

$$p(\mathbf{z}|\mathcal{D}_n, \text{C1}, \text{C2}) \propto \Phi_{\sigma^2}(f(\mathbf{x}_\star) - y_{\text{max}}) \Big[\textstyle\prod_{i=1}^{d} \mathbb{I}\big([\nabla^2 f(\mathbf{x}_\star)]_{ii} \leq 0\big) \Big] \mathcal{N}(\mathbf{z}|\mathbf{m}_0, \mathbf{V}_0)\,, \tag{6}$$

where Φ_{σ^2} is the cdf of a zero-mean Gaussian distribution with variance σ^2. The first new factor in this expression guarantees that $f(\mathbf{x}_\star) > y_{\text{max}} + \epsilon$, where we have marginalized out the zero-mean Gaussian noise ϵ, with variance σ^2. The second set of factors guarantees that the entries in $\text{diag}[\nabla^2 f(\mathbf{x}_\star)]$ are negative. We will then approximate this density with

$$p(\mathbf{z}|\mathcal{D}_n, \text{C1}, \text{C2}) \approx q(\mathbf{z}) \propto \Big[\textstyle\prod_{i=1}^{d+1} \mathcal{N}(z_i|\widetilde{m}_i, \widetilde{v}_i) \Big] \mathcal{N}(\mathbf{z}|\mathbf{m}_0, \mathbf{V}_0) \tag{7}$$

formed by replacing the intractible factors with Gaussian factors computed using Expectation Propagation (EP) [10]. The resulting algorithm is similar to the implementation of EP for binary classification with Gaussian processes [12]. Note that these computations have so far not depended on $\mathbf{x}$, so we can compute them for a given $\mathbf{x}_\star$ and use them to evaluate any subsequent inputs.

Finally, given a query input $\mathbf{x}$ let $\mathbf{f} = [f(\mathbf{x}); f(\mathbf{x}_\star)]$ be the concatenation of the values of the latent function at the input and optimizer. We can write this joint distribution as the Gaussian

$$p(\mathbf{f}|\mathcal{D}_n, \text{C1}, \text{C2}) \approx \textstyle\int p(\mathbf{f}|\mathbf{z}, \mathcal{D}_n, \text{C1})\, q(\mathbf{z})\, d\mathbf{z} = \mathcal{N}(\mathbf{f}|\mathbf{m}_\mathbf{f}, \mathbf{V}_\mathbf{f})\,. \tag{8}$$

The required quantities are similar to the ones used by EP to make predictions in the Gaussian process binary classifier [12]. We can then incorporate C3 by multiplying $\mathcal{N}(\mathbf{f}|\mathbf{m}_\mathbf{f}, \mathbf{V}_\mathbf{f})$ with a factor that guarantees $f(\mathbf{x}) < f(\mathbf{x}_\star)$ and using a similar approximation as above. The predictive distribution for $f(\mathbf{x})$ given $\mathbf{x}_\star$ can then be read off from the resulting Gaussian approximation resulting in variance $v_n(\mathbf{x}|\mathbf{x}_\star)$. This leads to the final form of our approximation, $\text{H}[p(y|\mathcal{D}_n, \mathbf{x}, \mathbf{x}_\star)] \approx 0.5 \log[2\pi e(v_n(\mathbf{x}|\mathbf{x}_\star) + \sigma^2)]$.

3 Experiments

First, we analyze the accuracy of PES in the task of approximating the differential entropy (1). We compare the PES approximation (3), with the approximation used by the entropy search (ES) method [4]. We also compare with the ground truth for (1) obtained using a rejection sampling (RS) algorithm based on (2). For this experiment we generate the data $\mathcal{D}_n$ using an objective function f sampled from the Gaussian process prior as in [4]. We fix $M = 200$ and $m = 1000$. The plots in Figure 1 show that the PES approximation to (1) is more similar to the ground truth given by RS than the approximation produced by ES. In this figure we also see a discrepancy between RS and PES at locations near $\mathbf{x} = (0.572, 0.687)$. This difference is an artifact of the discretization used in RS. By zooming in and drawing many more samples we would see the same behavior in both plots.

We now evaluate the performance of PES in the task of finding the optimum of synthetic black-box objective functions. In this experiment we optimize objective functions defined in the 2-dimensional unit domain $\mathcal{X} = [0, 1]^2$. Each objective function is generated by first sampling 1024 function values from a known GP prior and defining the the objective function by the resulting GP posterior mean. We generated a total of 1000 objective functions by following this procedure. In these experiments we compared the performance of PES with that of ES [4] and expected improvement (EI) [8]. Predictive performance is measured in terms of the immediate regret (IR) $|f(\widetilde{\mathbf{x}}_n) - f(\mathbf{x}_\star)|$, where $\widetilde{\mathbf{x}}_n$ is the recommendation of each algorithm had we stopped at step n—for all methods this is given by the maximizer of the posterior mean. The right plot in Figure 2 shows the decimal logarithm of the median of the IR obtained by each method across the 1000 *different* objective functions. Confidence bands equal to one standard deviation are obtained using the bootstrap method.

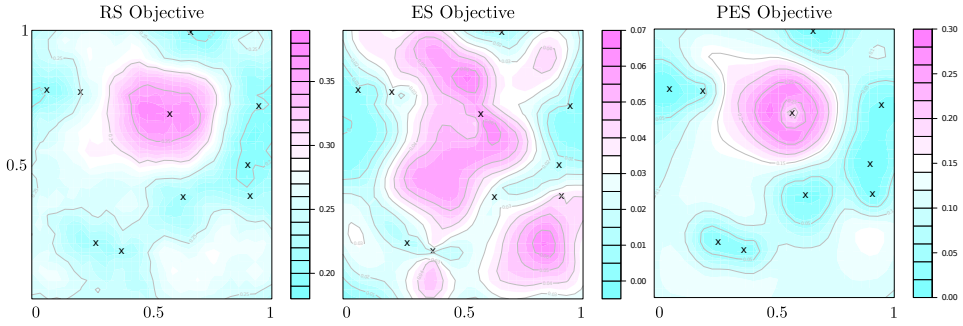

Figure 1: Comparison of different estimates of $\alpha_n(\mathbf{x})$. Left, ground truth obtained by the rejection sampling method RS. Middle, approximation produced by ES. Right, approximation produced by PES.

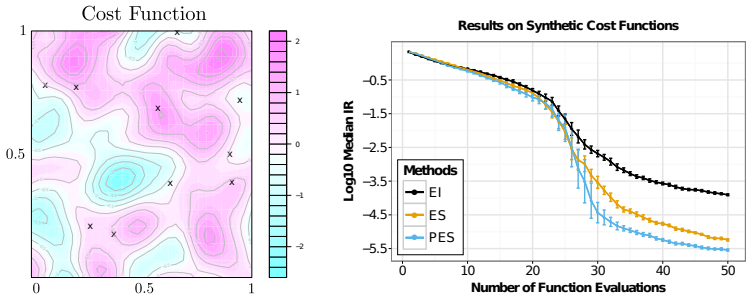

Figure 2: Left, example of objective functions f. Right, median of the immediate regret (IR) for the methods PES, ES and EI in the experiments with synthetic objective functions.

We finally optimize several real-world cost functions. These examples optimize (NNet) the hyperparameters of a neural network; (Hydrogen) the hydrogen production of a particular bacteria with respect to its growth medium [3]; (Portfolio) the Sharpe ratio of 1-year returns generated by simulation of a multivariate time-series model [7]; and the speed of a bipedal robot [15] under (Walker A) noiseless and (Walker B) noisy observations.

4 Conclusions

We have proposed a novel information-theoretic approach for Bayesian optimization, PES, which maximizes the one-step information gain over the optimizer. We show that PES produces more accurate approximations than ES and our experiments show that it can more easily marginalize over hyperparameters. Experiments with synthetic and real-world functions also show that PES often outperforms ES as well as EI, another popular Bayesian optimization technique. Overall, this work shows that PES provides an attractive, efficient formalism for Bayesian optimization.

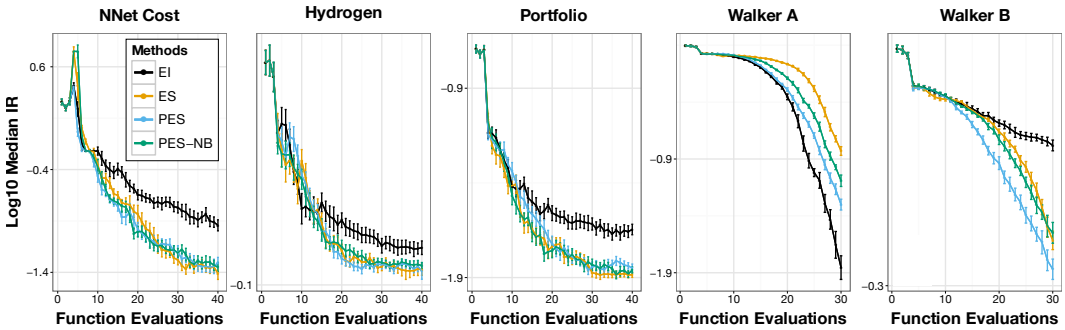

Figure 3: Median of the immediate regret (IR) for the methods PES, PES-NB, ES and EI in the experiments with non-analytic real-world cost functions.

References

[1] S. Bochner. *Lectures on Fourier integrals*. Princeton University Press, 1959.

[2] 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. Technical Report UBC TR-2009-23 and arXiv:1012.2599v1, Dept. of Computer Science, University of British Columbia, 2009.

[3] E. H. Burrows, W.-K. Wong, X. Fern, F. W. R. Chaplen, and R. L. Ely. Optimization of ph and nitrogen for enhanced hydrogen production by synechocystis sp. pcc 6803 via statistical and machine learning methods. *Biotechnology Progress*, 25(4):1009–1017, 2009.

[4] P. Hennig and C. J. Schuler. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13, 2012.

[5] J. M. Hernández-Lobato, M. W. Hoffman, and Z. Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *NIPS*, 2014.

[6] N. Houlsby, J. M. Hernández-Lobato, F. Huszar, and Z. Ghahramani. Collaborative Gaussian processes for preference learning. In *NIPS*, pages 2096–2104, 2012.

[7] E. Jondeau and M. Rockinger. The copula-GARCH model of conditional dependencies: An international stock market application. *Journal of international money and finance*, 25(5):827–853, 2006.

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

[9] D. J. MacKay. Information-based objective functions for active data selection. *Neural Computation*, 4(4):590–604, 1992.

[10] T. P. Minka. *A family of algorithms for approximate Bayesian inference*. PhD thesis, Massachusetts Institute of Technology, 2001.

[11] A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *NIPS*, pages 1177–1184, 2007.

[12] C. E. Rasmussen and C. K. Williams. *Gaussian processes for machine learning*. The MIT Press, 2006.

[13] E. Solak, R. Murray-Smith, W. E. Leithead, D. J. Leith, and C. E. Rasmussen. Derivative observations in Gaussian process models of dynamic systems. In *NIPS*, pages 1057–1064, 2003.

[14] J. Villemonteix, E. Vazquez, and E. Walter. An informational approach to the global optimization of expensive-to-evaluate functions. *Journal of Global Optimization*, 44(4):509–534, 2009.

[15] E. Westervelt and J. Grizzle. *Feedback Control of Dynamic Bipedal Robot Locomotion*. Control and Automation Series. CRC PressINC, 2007.