Knowledge-Based Systems 322 (2025) 113612

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Alpha entropy search for new information-based Bayesian optimization

Daniel Fernández-Sánchez [a,*], Eduardo C. Garrido-Merchán [b], Daniel Hernández-Lobato [a]

[a] Machine Learning Group, Computer Science Department, Escuela Politécnica Superior, Universidad Autónoma de Madrid, Francisco Tomás y Valiente 11, 28049, Madrid, Spain

[b] Institute for Research in Technology (IIT), Universidad Pontificia Comillas, Alberto Aguilera 23, 28015 Madrid, Spain

ARTICLE INFO

Dataset link: github.com/fernandezdaniel/alphaES

Keywords:
Bayesian optimization
Information theory
Entropy search
Alpha-divergence

ABSTRACT

Bayesian optimization (BO) methods based on information theory have obtained state-of-the-art results in several tasks. These techniques rely on the Kullback–Leibler (KL) divergence to compute the acquisition function. We introduce a novel information-based class of acquisition functions for BO called Alpha Entropy Search (AES). AES is based on the alpha-divergence, which generalizes the KL-divergence. Iteratively, AES selects the next evaluation point as the one whose associated target value has the highest level of dependency with respect to the location and associated value of the global maximum of the optimization problem. Dependency is measured in terms of the alpha-divergence, as an alternative to the KL-divergence. Intuitively, this favors evaluating the objective function at the most informative points about the global maximum. The alpha-divergence has a free parameter α, which determines the behavior of the divergence, balancing local and global differences. Therefore, different values of α result in different acquisition functions. AES acquisition lacks a closed-form expression. However, we propose an efficient and accurate approximation using a truncated Gaussian distribution. In practice, the value of α can be chosen by the practitioner, but here we suggest using a combination of acquisition functions obtained by simultaneously considering a range of α values. We provide an implementation of AES in BOTorch and we evaluate its performance in synthetic, benchmark, and real-world experiments involving the tuning of the hyper-parameters of a deep neural network. These experiments show that AES performance is competitive with other information-based acquisition functions such as JES, MES, or PES.

1. Introduction

Bayesian optimization (BO) includes a set of methods that have been successfully used for the optimization of black-box functions, and most concretely for the problem of tuning the hyper-parameters of machine learning models [1]. In particular, a black-box function $f(\cdot)$ is characterized by having an unknown analytical expression and being costly to evaluate in computational or economic terms. Besides this, we also consider that the evaluation of $f(\cdot)$ may be corrupted by noise. Formally, the optimization scenario we consider can be defined as trying to find:

$$\mathbf{x}^\star = \arg\max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}), \tag{1}$$

where $f(\cdot)$ is the objective, and $\mathbf{x}^\star$ is the global optimum in the considered bounded input space $\mathcal{X} \subset \mathbb{R}^d$. We also denote $y^\star$ as the associated optimal objective value. Namely, $y^\star = f(\mathbf{x}^\star)$. Importantly, we also assume that the evaluation of $f(\cdot)$ may be contaminated by Gaussian random noise. That is, instead of observing directly $f(\mathbf{x})$, we observe $y = f(\mathbf{x}) + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma^2)$.

Since the objective is assumed to be very expensive to evaluate, we would like to use as few evaluations as possible to estimate $\mathbf{x}^\star$. BO methods are very successful in this task [2–4]. To tackle this scenario, given a small number of initial evaluations, they focus on modeling the black-box function $f(\cdot)$ with a probabilistic surrogate model in the input space $\mathcal{X}$. This model is typically a Gaussian Process (GP), which outputs a predictive distribution for $f(\cdot)$, capturing the potential values of the objective in regions of the input space that have not been explored so far [5]. Using this model, BO methods guide the search for the optimum and make intelligent decisions about what point should be evaluated next at each iteration. For this, they use an acquisition function $a(\mathbf{x})$ that measures the expected utility of performing an evaluation at $\mathbf{x}$ with the goal of solving the optimization problem [3]. The next evaluation is simply the maximizer of the acquisition function $a(\mathbf{x})$. Importantly, the probabilistic model and the acquisition function are very cheap to

* Corresponding author.
E-mail addresses: daniel.fernandezs@uam.es (D. Fernández-Sánchez), ecgarrido@icade.comillas.edu (E.C. Garrido-Merchán), daniel.hernandez@uam.es (D. Hernández-Lobato).

https://doi.org/10.1016/j.knosys.2025.113612
Received 27 November 2024; Received in revised form 12 April 2025; Accepted 19 April 2025
Available online 16 May 2025

evaluate and maximize, respectively, since they do not imply evaluating the actual objective. Therefore, the overhead that the process described introduces can be considered negligible. After enough evaluations of the objective are performed, the best observation is returned as the global optimizer of $f(\cdot)$, in a noiseless setting. In a noisy setting, a similar recommendation is made, but the probabilistic model is used first to remove the noise from the evaluations performed.

There is a plethora of different optimization scenarios, which are particular cases of the one described, where BO methods can be applied in practice in science or engineering. For example, BO has successfully been used in energy for robust ocean wave features prediction [6]; in chemistry, for the discovery of energy storage molecular materials [7]; in robotics, for a better design of the wing shape of an unmanned aerial vehicle [8]; in finance, for environmental, social and governance sustainable portfolio optimization [9]; or even as a way to suggest a better dessert, optimizing chocolate chip cookies [10]. Importantly, BO also gives superior results to other optimization methods that do not rely on a model to guide the search for the optimum, such as meta-heuristics or genetic algorithms [11].

A critical and important part of any BO method is the acquisition function. In this regard, in the BO literature, acquisition functions based on information theory have delivered state-of-the-art results by efficiently guiding the search for $\mathbf{x}^\star$ [12–14]. These strategies choose the next evaluation point as the one that results in the highest expected decrease in the entropy of the global optimum $\mathbf{x}^\star$. The global optimum can be regarded as a random variable because there is uncertainty about the potential values of the objective $f(\cdot)$. These potential values are modeled by the GP. Among information-based strategies, the recently proposed method Joint Entropy Search (JES) obtains state-of-the-art results [15,16]. The JES acquisition function $a(\mathbf{x})$ measures the expected reduction in the differential entropy of both $\mathbf{x}^\star$ and $y^\star$, *i.e.*, $\{\mathbf{x}^\star, y^\star\}$ after observing y at $\mathbf{x}$. Importantly, this expected reduction in the differential entropy can be shown to be equal to the mutual information between $\{\mathbf{x}^\star, y^\star\}$ and y, where the distribution of y, *i.e.*, the noisy objective value at the candidate point $\mathbf{x}$, is also given by the GP [15]. The mutual information is just the Kullback–Leibler (KL) divergence between $p(\{\mathbf{x}^\star, y^\star\}, y)$ and $p(\{\mathbf{x}^\star, y^\star\})p(y)$, *i.e.*, the same distribution, but where independence between $\{\mathbf{x}^\star, y^\star\}$ and y is assumed. The KL divergence is always non-negative and equal to zero only when the two distributions are equal. Thus, JES simply chooses the next point $\mathbf{x}$ to evaluate as the one at which there is a higher level of dependency between $\{\mathbf{x}^\star, y^\star\}$ and y, as measured by the KL divergence. Observing y at such an $\mathbf{x}$ will provide more knowledge about the potential values of $\{\mathbf{x}^\star, y^\star\}$, due to the strong dependencies, and is expected to help the most to solve the optimization problem.

As an alternative to the KL divergence, we consider in this paper other methods to estimate the level of dependency between $\{\mathbf{x}^\star, y^\star\}$ and y. More precisely, we consider a generalization of the KL divergence to measure how similar $p(\{\mathbf{x}^\star, y^\star\}, y)$ is to $p(\{\mathbf{x}^\star, y^\star\})p(y)$, the α-divergence [17]. The α-divergence includes a parameter α that influences the behavior of the divergence, trading-off evaluating differences between each distribution at a single mode, and evaluating differences globally [18]. We denote the resulting acquisition function as Alpha Entropy Search (AES). AES relies on the evaluation of the α-divergence between $p(\{\mathbf{x}^\star, y^\star\}, y)$ and $p(\{\mathbf{x}^\star, y^\star\})p(y)$. Unfortunately, such a divergence is intractable. To address this problem, we describe a simple and efficient method to approximate its value and hence the corresponding acquisition function. Of course, changing α results in different acquisition functions. Notwithstanding, empirically, we did not observe a value for α that gives overall good results. In consequence, we suggest considering simultaneously a range of values for α, resulting in a weighted combination of acquisition functions. Such a method can be seen as an ensemble that combines several acquisition functions to obtain a final acquisition function, in a similar way as machine learning ensemble methods combine several individual predictors to obtain a more robust aggregated predictor [19]. This avoids relying on a single α value which may omit relevant information, since α influences the sensitivity of the divergence to the mismatch between probability distributions in different regions of the input space. We have evaluated this combination across several experiments, including synthetic, benchmark, and real-world problems related to the tuning of the hyper-parameters of deep neural networks. These results show that, in general, the proposed method based on the α-divergence, gives similar or better results than other information-based BO acquisition functions.

The organization of the paper is as follows: first, we include a section where we give fundamental details about information-based BO. Next, another section explains the analytical details and the methodology of our proposed approach, the AES acquisition function and the associated ensemble method. We continue with a section about related work, where we emphasize the similarities and differences between our proposed method and other related methods that have been published in the BO literature. Afterward, we include a section where we give empirical evidence of the performance of AES with respect to other information-based acquisition functions in synthetic, benchmark, and real-world experiments. Finally, we end the manuscript with a section summarizing the conclusions.

2. Information-based Bayesian optimization

For clarity, we begin this section by illustrating the fundamentals of information-based BO, to further introduce the proposed acquisition function, AES, in the next section.

We begin with a short description of the vanilla BO algorithm. As we have briefly described in the introduction, BO is a class of methods that optimize black-box functions [3]. In particular, the algorithm receives as an input an initial dataset $D_0 = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_0}$, where each $\mathbf{x}_i$ represents the initial input space points in $\mathcal{X}$ and y_i denotes the associated noisy observation of the objective value at $\mathbf{x}_i$. This dataset is chosen at random from $\mathcal{X}$. Then, a probabilistic surrogate model, such as a Gaussian process (GP) fits the initial points in D_0 to model the underlying objective function $f(\cdot)$ [5]. Afterward, we enter a loop where, at each iteration t, the algorithm selects the next candidate point to evaluate $\mathbf{x}_t$ by maximizing an acquisition function $a(\mathbf{x})$. This acquisition function balances exploration and exploitation of the objective values [4]. The acquisition function uses the posterior predictive distribution of the GP for the values of $f(\cdot)$ at each $\mathbf{x}$, *i.e.*, $p(f(\mathbf{x})|\mathbf{x}, D_0)$, to estimate the expected utility of performing an evaluation of the objective at $\mathbf{x}$. Then, the objective function is evaluated at the maximizer of the acquisition. That is, $\mathbf{x}_t = \arg\max_{\mathbf{x}\in\mathcal{X}} a(\mathbf{x})$. This retrieves the noisy objective value at $\mathbf{x}_t$. Namely, $y_t = f(\mathbf{x}_t) + \epsilon_t$, where ϵ_t is assumed to be Gaussian noise. Next, this new observation is added to the dataset of observed points, $D_t = D_{t-1} \cup \{(\mathbf{x}_t, y_t)\}$, and the GP model is updated with D_t. We illustrate these steps in Fig. 1. The process iterates for a predefined number of steps or budget T, and when the execution finishes, then it returns the point $\mathbf{x}_t$ with the best associated value y_t in the noiseless setting. In the noisy setting, we may use the GP to remove the noise around each y_t first. We summarize the steps of the described BO algorithm in Algorithm 1. Importantly, the GP and the acquisition $a(\cdot)$ are very cheap to evaluate and maximize, respectively, since they do not imply evaluating the actual objective. Thus, the extra time that the process described introduces can be considered negligible compared with the cost of evaluating the objective each time, which is assumed to be extremely expensive. Because of the intelligent decisions made when choosing each evaluation point, BO methods often perform much better than a random exploration of the input space [1].

When a GP is used as the underlying probabilistic model in BO, it is assumed that the objective function $f(\cdot)$ is a sample from such a GP. As a consequence, at any location $\mathbf{x} \in \mathcal{X}$, the distribution of the potential values of the latent function $f(\cdot)$ at $\mathbf{x}$, conditioned on the

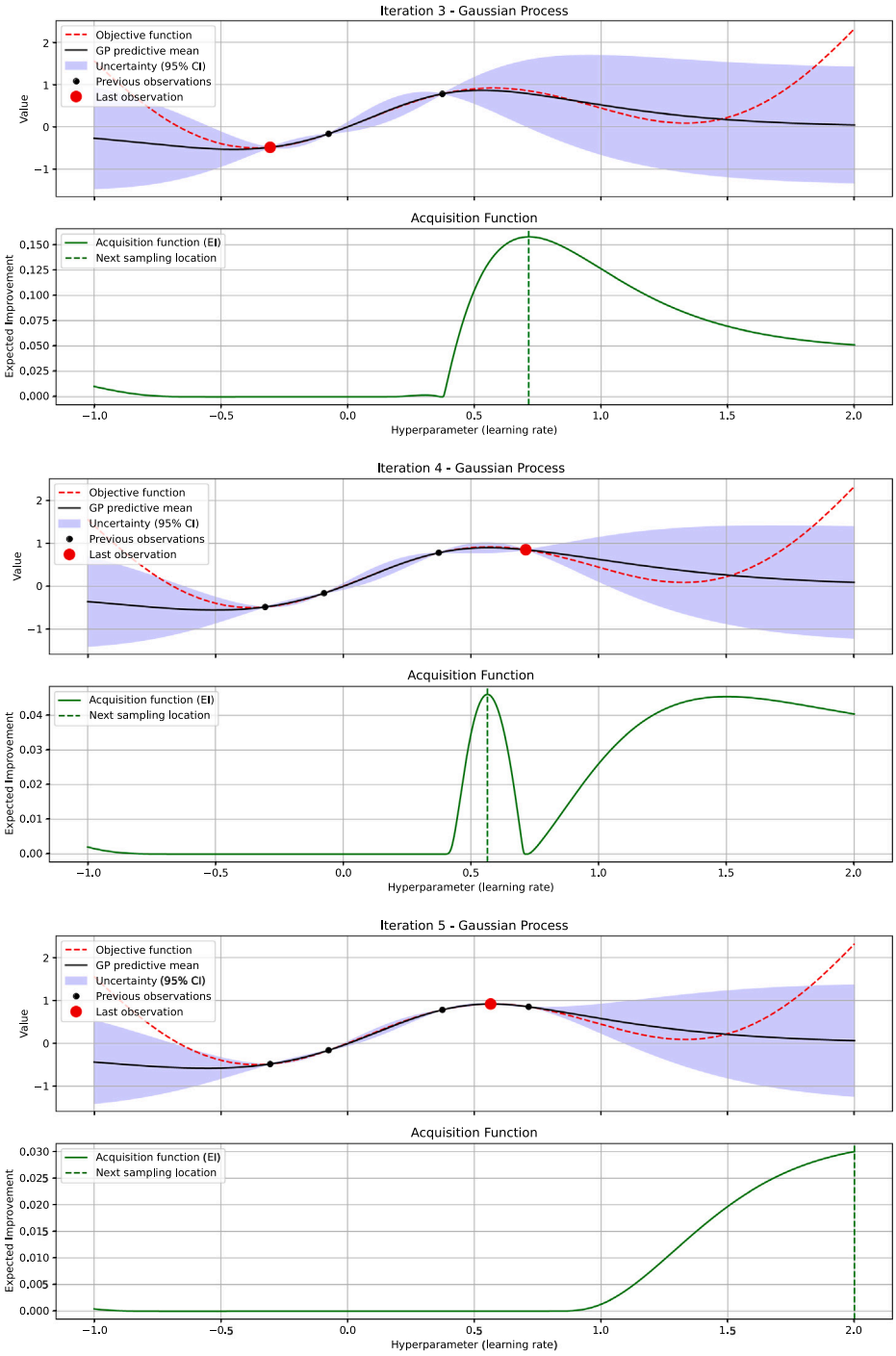

Fig. 1. GP fit of the objective function (top of images) and the associated acquisition function (Expected Improvement [4]) built using the predictive distribution of the GP (down). Black points are observations, and the red point is the last evaluation performed (*i.e.*, the maximizer of the acquisition function in the previous iteration). We can see the BO process as iterations are carried out (from$t = 3$ to t=5) and how the GP and the associated acquisition function guide the search for the optimum.

current observations $\mathcal{D}_{t-1} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, *i.e.*, $p(y|\mathbf{x}, \mathcal{D}_{t-1})$, is Gaussian, with mean $\mu_n(\mathbf{x})$ and variance $\upsilon_n(\mathbf{x})$

$$\mu_n(\mathbf{x}) = \mathbf{k}_n(\mathbf{x})^\top (\mathbf{K}_n + \sigma^2 \mathbf{I})^{-1} \mathbf{y}_n, \tag{2}$$

$$\upsilon_n(\mathbf{x}) = k(\mathbf{x}, \mathbf{x}) - \mathbf{k}_n(\mathbf{x})^\top (\mathbf{K}_n + \sigma^2 \mathbf{I})^{-1} \mathbf{k}_n(\mathbf{x}), \tag{3}$$

where $\mathbf{k}_n(\mathbf{x})$ is a vector of cross-covariance terms between $f(\mathbf{x})$ and $\{f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)\}$; $\mathbf{K}_n$ is the $n \times n$ covariance matrix between each $\{f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)\}$; $k(\mathbf{x}, \mathbf{x})$ is the prior variance of $f(\mathbf{x})$; and σ^2 is the variance of the noise. See [5] for further details. All these covariance values are computed in terms of a covariance function $k(\cdot, \cdot)$, which in the case of BO is often set to be the Matérn covariance function [1]. All GP hyper-parameters such as length-scales, the variance of the noise, etc. are tuned, *e.g.*, by maximizing the marginal likelihood [5].

Information-based BO methods use concepts from information theory to estimate the acquisition function $a(\mathbf{x})$ [12–16]. Concretely, they use the notion of information gain to guide the selection of the next query point $\mathbf{x}_t$ to reduce the most the uncertainty about the objective global maximum. Uncertainty can be measured in terms of the entropy. In other words, one seeks the point whose evaluation maximizes the expected reduction of the entropy about some random variable that is related to the extremum, which, *e.g.*, can be the location of the optimum

Algorithm 1 Bayesian Optimization Vanilla Algorithm

Input: Initial dataset $\mathcal{D}_0 = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_0}$, GP prior $p(f)$, acquisition function $a(\cdot)$, number of iterations T
Fit Gaussian process model to $\mathcal{D}_0$
for $t = 1$ to T **do**
  Select next query point $\mathbf{x}_t = \arg\max_{\mathbf{x}\in\mathcal{X}} a(\mathbf{x})$
  Evaluate the objective function $y_t = f(\mathbf{x}_t) + \epsilon_t$, where $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$
  Augment the dataset $\mathcal{D}_t = \mathcal{D}_{t-1} \cup \{(\mathbf{x}_t, y_t)\}$
  Update Gaussian process model with $\mathcal{D}_t$
end for
Output: The best point found $\mathbf{x}_i$, where $i = \arg\max y_i$, in the noiseless setting.

in the input space $\mathbf{x}^\star$ or its associated value $y^\star$. These quantities can be regarded as random variables since the GP imposes a probability distribution on the objective values, and hence on the global optimum of the objective. The logic behind the process is to iteratively minimize the entropy of these quantities through evaluations that maximize the expected reduction of their entropy. Low entropy means a better knowledge of the values that the random variable can take. The current information about $\{\mathbf{x}^\star, y^\star\}$ can be estimated in terms of the negative differential entropy of the distribution $p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})$, where $\mathcal{D}_{t-1}$ is the dataset of observations and evaluations performed so far. Such a distribution is fully specified by the GP model. Specifically, in Joint Entropy Search (JES) one selects $\mathbf{x}_t$ by maximizing the following expression [15]:

$$a(\mathbf{x}) = H[p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})] - \mathbb{E}_{p(y|\mathcal{D}_{t-1},\mathbf{x})}[H[p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1} \cup \{(\mathbf{x}, y)\})]], \tag{4}$$

where $H[\cdot]$ denotes the differential entropy of the associated probability distribution; the first term in (4) is just the entropy of the solution of the optimization problem at the current iteration; and the second term in (4) is the expected entropy after observing y at $\mathbf{x}$. Of course, we do not know the actual value of y. However, as described previously, the GP gives a predictive distribution for its values given the observed data. Namely, $p(y|\mathcal{D}_{t-1}, \mathbf{x})$, which is Gaussian with mean given by (2) and variance given by (3) plus the noise variance σ^2, to account for the fact that y is a potential noisy version of $f(\mathbf{x})$.

A limitation of the approach described is that (4) is too complicated to evaluate in closed-form, which makes difficult its practical use. To overcome this difficulty, in [15] it is suggested to use the fact that (4) is the mutual information between $\{\mathbf{x}^\star, y^\star\}$ and y, $I(\{\mathbf{x}^\star, y^\star\}; y)$. The mutual information is symmetric and one can swap the roles between $\{\mathbf{x}^\star, y^\star\}$ and y to obtain an alternative but equivalent expression to (4) [12]:

$$a(\mathbf{x}) = H[p(y|\mathcal{D}_{t-1}, \mathbf{x})] - \mathbb{E}_{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})}[H[p(y|\{\mathbf{x}^\star, y^\star\}, \mathcal{D}_{t-1}, \mathbf{x})]], \tag{5}$$

where now the first term in (5) is just the entropy of the predictive distribution at $\mathbf{x}$ given the observed data; the expectation in (5) can be simply approximated by Monte Carlo by sampling $\{\mathbf{x}^\star, y^\star\}$ given the current observations; and $H[p(y|\{\mathbf{x}^\star, y^\star\}, \mathcal{D}_{t-1}, \mathbf{x})]$ is the entropy of the predictive distribution at $\mathbf{x}$ given that $\{\mathbf{x}^\star, y^\star\}$ is the solution to the optimization problem.

Approximately sampling $\{\mathbf{x}^\star, y^\star\}$ from the GP predictive distribution is tractable. This can be done using a random Fourier feature approximation of the GP to sample functions from the GP posterior [12]. These functional samples can then be optimized to obtain an approximate sample of $\{\mathbf{x}^\star, y^\star\}$. The distribution $p(y|\{\mathbf{x}^\star, y^\star\}, \mathcal{D}_{t-1}, \mathbf{x})$ is intractable. However, it can be approximated by a Gaussian distribution by making use of a truncated Gaussian to estimate the distribution of $f(\mathbf{x})$ given that $f(\mathbf{x}) < y^\star$ [15]. This is precisely the approach followed in the BO framework BOTorch to evaluate the JES acquisition function [20].

Finally, besides JES, there are other information-based strategies suggested in the literature: entropy search [13,14], the first information-based BO method; predictive entropy search [12], which only consider the entropy of $\mathbf{x}^\star$; and max value entropy search [21], which only considers the entropy of $y^\star$. JES has shown similar or better results than these strategies [15].

3. Alpha entropy search

As described in the previous section, the JES acquisition in (5) is the mutual information between $\{\mathbf{x}^\star, y^\star\}$ and y, denoted $I(\{\mathbf{x}^\star, y^\star\}; y)$, given the current observations collected so far $\mathcal{D}_{t-1}$. It is well known that the mutual information can be expressed in terms of the Kullback–Leibler (KL) divergence between probability distributions. To illustrate this, consider two distributions $p(\mathbf{x})$ and $q(\mathbf{x})$. The KL-divergence between them is:

$$\mathrm{KL}(p(\mathbf{x}) \parallel q(\mathbf{x})) = \int p(\mathbf{x}) \log \frac{p(\mathbf{x})}{q(\mathbf{x})}\, d\mathbf{x}, \tag{6}$$

which is non-symmetric, non-negative, and equal to zero only when $p(\mathbf{x}) = q(\mathbf{x})$. Thus, by taking a look at (6) we can see that we can write (5) as follows:

$$\begin{aligned} a(\mathbf{x}) &= H[p(y|\mathcal{D}_{t-1}, \mathbf{x})] - \mathbb{E}_{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})}[H[p(y|\{\mathbf{x}^\star, y^\star\}, \mathcal{D}_{t-1}, \mathbf{x})]] \\ &= I(\{\mathbf{x}^\star, y^\star\}; y) = I(y; \{\mathbf{x}^\star, y^\star\}) \\ &= \mathrm{KL}(p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x}) \parallel p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1}) p(y|\mathcal{D}_{t-1}, \mathbf{x})). \end{aligned} \tag{7}$$

Appendix A shows the details of this identity. Note that (7) is just the KL-divergence between a joint probability distribution $p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x})$ and the corresponding factorizing distribution $p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1}) p(y|\mathcal{D}_{t-1}, \mathbf{x})$ that assumes independence between y and $\{\mathbf{x}^\star, y^\star\}$. This allows to interpret the JES acquisition in the following way. Specifically, (7) measures the level of dependency between y and $\{\mathbf{x}^\star, y^\star\}$ at $\mathbf{x}$ in terms of the KL-divergence. The next point to evaluate is thus the one maximizing this level of dependency. The idea is that because of the strong dependencies between y and $\{\mathbf{x}^\star, y^\star\}$ at $\mathbf{x}$, observing y will give a lot of information about the potential values of $\{\mathbf{x}^\star, y^\star\}$, *i.e.*, the solution of the optimization problem, and will help the most to solve it. In this work, we conjecture that other methods to estimate the level of dependency between $\{\mathbf{x}^\star, y^\star\}$ and y may provide better optimization results in some scenarios. With this idea in mind, we propose to consider a different divergence between probability distributions. Namely, the α-divergence, which is described next.

3.1. Amari's α-divergence

The KL-divergence (6) can be generalized by replacing the natural logarithm with the α-logarithm and multiplying it by α^{-1} [22]. The α-logarithm (also known as the Tsallis logarithm or the q-logarithm) is defined as:

$$\log_\alpha(x) = \frac{x^{1-\alpha} - 1}{1 - \alpha}, \tag{8}$$

with $\alpha \in \mathbb{R} \setminus \{1\}$ and such that $\log_\alpha(x) \to \log(x)$ when $\alpha \to 1$. This substitution leads to Amari's α-divergence between probability distributions $p(\mathbf{x})$ and $q(\mathbf{x})$ [17], which is defined as:

$$D_\alpha(p(\mathbf{x}) \parallel q(\mathbf{x})) = \frac{1}{(1-\alpha)\alpha} \left(1 - \int q(\mathbf{x})^{1-\alpha} p(\mathbf{x})^\alpha d\mathbf{x}\right), \tag{9}$$

for $\alpha \in \mathbb{R} \setminus \{0, 1\}$. This divergence is also non-negative and only equal to zero when the two distributions coincide. Amari's divergence is parameterized by α, which adjusts the sensitivity to different regions of the probability distributions, allowing us to control the emphasis of the divergence on specific differences between $p(\mathbf{x})$ and $q(\mathbf{x})$.

In Fig. 2 (top-row), we illustrate how varying α affects the behavior of Amari's divergence. The figure shows the results obtained when

minimizing $D_\alpha(p(x) \parallel q(x))$ when $p(x)$ is a multi-modal distribution and $q(x)$ is a Gaussian distribution. For better visualization purposes, we have considered a version of the α-divergence that applies to unnormalized distributions [18]. When using this version, the optimal parameters for the mean and the variance of the Gaussian are the same as those obtained when optimizing (9). The only difference is found in the constant that multiplies $q(x)$. The behavior of the α-divergence with respect to α can be summarized as:

- $\alpha \to -\infty$: The divergence emphasizes capturing one of the modes of $p(x)$.
- $\alpha \to 0$: The divergence converges to the reversed Kullback–Leibler divergence $\mathrm{KL}(q(x) \parallel p(x))$:

  $$\lim_{\alpha\to 0} D_\alpha(p(x) \,\|\, q(x)) = \mathrm{KL}(q(x) \,\|\, p(x))$$

  At this point, $q(x)$ starts to capture more of the narrow mode of $p(x)$.
- $\alpha = 0.5$: $q(x)$ captures slightly more of the narrow mode of $p(x)$ compared to when $\alpha \to 0$. In this case the α-divergence is equal to the Hellinger distance:

  $$D_{0.5}(p(x) \parallel q(x)) = 2\int \left(\sqrt{p(x)} - \sqrt{q(x)}\right)^2 dx$$

- $\alpha \to 1$: $q(x)$ captures even more of the narrow mode of $p(x)$ compared to $\alpha = 0.5$. The divergence converges to the direct Kullback–Leibler divergence $D_{\mathrm{KL}}(p(x) \parallel q(x))$:

  $$\lim_{\alpha\to 1} D_\alpha(p(x) \,\|\, q(x)) = D_{\mathrm{KL}}(p(x) \,\|\, q(x))$$

- $\alpha \to \infty$: In this limit, the divergence encourages $q(x)$ to cover the full distribution $p(x)$.

Thus, by adjusting α, we can control how the divergence trades-off evaluating differences between the distributions at a single mode, and evaluating differences between them globally, as illustrated by Fig. 2 (top-row).

Fig. 2 (bottom-row) shows the results obtained when minimizing $D_\alpha(p(\mathbf{x}) \parallel q(\mathbf{x}))$ when $p(\mathbf{x})$ is a bi-variate Gaussian distribution with strong dependencies and $q(\mathbf{x})$ is a factorizing bi-variate Gaussian. For better visualization, here, we guarantee that each distribution integrates to 1 and we directly optimize (9). Again, we observe that when $\alpha \to 0$, $q(\mathbf{x})$ captures just the mode of $p(\mathbf{x})$. By contrast, when $\alpha \to 1$, the divergence encourages $q(\mathbf{x})$ to cover more of the full distribution $p(\mathbf{x})$. Summing up, α has a strong influence on the behavior of the divergence and determines in which regions of the input space the differences between the two distributions are measured.

In most practical applications of α-divergences for approximate inference, only values of α in the interval $(0, 1)$ are considered, allowing to interpolate between the two KL-divergences described above [23,24]. Therefore, in this work, we only consider such a range of values for α. Note that considering different values of α will result in different acquisition functions for BO.

We expect that by replacing the KL-divergence of JES with a more general divergence, such as Amari's α-divergence, we will be able to adjust α and modulate the weight given to the discrepancies between distributions across different regions. Specifically, different values of α will amplify or down-weight differences across different areas of the input space, leading to a hopefully better way, in some scenarios, of exploring the objective function towards its optimum. This substitution gives us the following acquisition function:

$$\begin{aligned} a_{\mathrm{AES}}(\mathbf{x}) &= D_\alpha(p(y, \{\mathbf{x}^\star, y^\star\}|D_{t-1}, \mathbf{x}) \parallel p(\{\mathbf{x}^\star, y^\star\}|D_{t-1}, \mathbf{x})p(y|D_{t-1}, \mathbf{x})) \\ &= \frac{1}{(1-\alpha)\alpha}\left(1 - \mathbb{E}_{p(\{\mathbf{x}^\star, y^\star\}|D_{t-1})}\left[\int p(y|D_{t-1}, \mathbf{x})\left(\frac{p(y|D_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})}{p(y|D_{t-1}, \mathbf{x})}\right)^\alpha dy\right]\right). \end{aligned} \tag{10}$$

Appendix B shows the detailed derivation. As is common with other information-based BO methods, this expression is analytically intractable and requires approximation. Specifically, neither the expectation in (10) nor the conditional distribution $p(y|D_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})$ can be computed in closed-form. Therefore, in the next section, we outline the specific approximations employed to evaluate and optimize the resulting acquisition function.

3.2. Approximating the conditional distribution and the expectation

As described previously, the conditional predictive distribution $p(y|D_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})$ is intractable. This distribution describes the potential values that y may take at $\mathbf{x}$ given the data observed so far D_{t-1} and that $\{\mathbf{x}^\star, y^\star\}$ is the solution of the optimization problem. Here, we adopt a similar approach as that of Joint Entropy Search (JES) [15] to approximate such a conditional predictive distribution. For simplicity, first, consider a noiseless evaluation setting, *i.e.*, $y = f(\mathbf{x})$. To approximate $p(f(\mathbf{x})|D_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})$, we incorporate $\{\mathbf{x}^\star, y^\star\}$ as extra data and use a truncated Gaussian distribution. Specifically, if $\{\mathbf{x}^\star, y^\star\}$ is the solution of the optimization problem, $f(\mathbf{x})$ cannot take larger values than $y^\star$, and we know that $f(\mathbf{x}^\star) = y^\star$ must be fulfilled. This last condition can be satisfied by incorporating the pair $(\mathbf{x}^\star, y^\star)$ into the observed data D_{t-1}, without observational noise. The truncated Gaussian distribution guarantees $f(\mathbf{x}) < y^\star$. More precisely, we consider the Gaussian unconditional predictive distribution after incorporating $(\mathbf{x}^\star, y^\star)$ as training data. That is, $p(f(\mathbf{x})|D_{t-1} \bigcup \{\mathbf{x}^\star, y^\star\}, \mathbf{x}) = \mathcal{N}(f(\mathbf{x})|m(\mathbf{x}), v(\mathbf{x}))$, with the mean and the variance respectively given by (2) and (3), and truncate the density above $y^\star$ to zero. The mean $m_{\mathrm{tr}}(\mathbf{x})$ and variance $v_{\mathrm{tr}}(\mathbf{x})$ of such a truncated Gaussian distribution are given by:

$$m_{\mathrm{tr}}(\mathbf{x}) = m(\mathbf{x}) - \sqrt{v(\mathbf{x})}\frac{\phi(\beta)}{\Phi(\beta)}, \tag{11}$$

$$v_{\mathrm{tr}}(\mathbf{x}) = v(\mathbf{x})\left[1 - \beta\frac{\phi(\beta)}{\Phi(\beta)} - \left(\frac{\phi(\beta)}{\Phi(\beta)}\right)^2\right], \tag{12}$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ are respectively the p.d.f. and c.d.f of a standard Gaussian, and $\beta = (y^\star - m(\mathbf{x}))/\sqrt{v(\mathbf{x})}$. Again, this is only an approximation to the intractable distribution $p(f(\mathbf{x})|D_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})$.

Consider now the noisy setting, *i.e.*, $y = f(\mathbf{x})+\epsilon$. As indicated in [25], taking into account the noise yields an extended skew distribution that does not have a tractable density. Therefore, to account for the noise and compute the conditional predictive distribution of y, we approximate the truncated Gaussian with a Gaussian distribution, as in [15]. Namely, $p(f(\mathbf{x})|D_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\}) \approx \mathcal{N}(f(\mathbf{x})|m_{\mathrm{tr}}(\mathbf{x}), v_{\mathrm{tr}}(\mathbf{x}))$. Now, we only have to add the Gaussian noise to $f(\mathbf{x})$. This results in a Gaussian approximation of the conditional distribution of y. That is, $p(y|D_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\}) \approx \mathcal{N}(y|m_{\mathrm{tr}}(\mathbf{x}), v_{\mathrm{tr}}(\mathbf{x}) + \sigma^2)$, where σ^2 is the variance of the noise.

After approximating the conditional distribution, we need to evaluate the integral in (10) with respect to y, so that we can estimate the AES acquisition at $\mathbf{x}$. This integral involves a product and a ratio between the predictive distribution conditioned to the problem's solution and the unconditioned distribution, to the power of α. Given that both distributions are Gaussian (after the approximations described above), we can evaluate the integral in closed form using the exponential form of the Gaussian distribution:

$$\begin{aligned} \int p(y|D_{t-1}, \mathbf{x})\left(\frac{p(y|D_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})}{p(y|D_{t-1}, \mathbf{x})}\right)^\alpha dy = \exp\big\{&(\alpha-1)g(\boldsymbol{\eta}) - \alpha g(\boldsymbol{\eta}^\star) \\ &+ g((1-\alpha)\boldsymbol{\eta} + \alpha\boldsymbol{\eta}^\star)\big\}, \end{aligned} \tag{13}$$

where $g(\boldsymbol{\eta})$ is the log-normalizer of a Gaussian with natural parameters $\boldsymbol{\eta}$, $\boldsymbol{\eta}$ are the natural parameters of $p(y|D_{t-1}, \mathbf{x})$, and $\boldsymbol{\eta}^\star$ are the natural parameters of the Gaussian approximation of $p(y|D_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})$. Specifically,

$$g(\boldsymbol{\eta}) = 0.5\log(2\pi) - 0.5\log\eta_2 + 0.5\frac{\eta_1^2}{\eta_2}, \quad \boldsymbol{\eta} = \left(\frac{m(\mathbf{x})}{v(\mathbf{x})+\sigma^2}, \frac{1}{v(\mathbf{x})+\sigma^2}\right)^{\mathrm{T}},$$

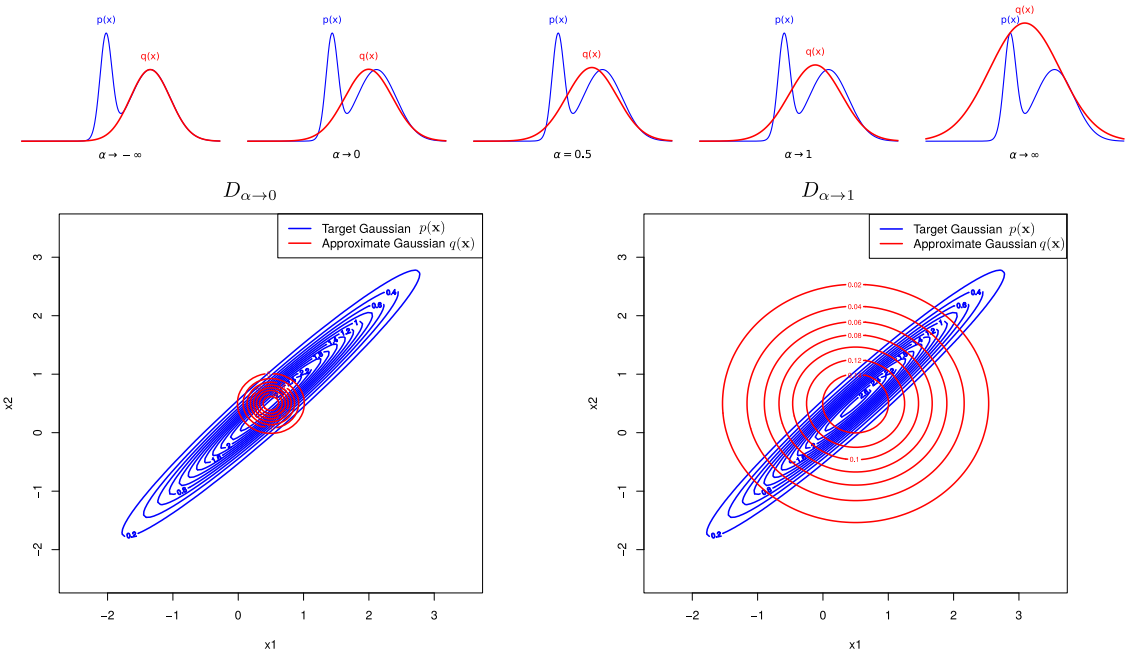

Fig. 2. (top-row) The Gaussian distribution $q(x)$ is fitted to $p(x)$ by minimizing Amari's divergence with different values of α. When $\alpha \to -\infty$, α tries to match one mode of $p(x)$, and as α increases, $q(x)$ starts covering more of the entire distribution. Finally, when $\alpha \to \infty$, $q(x)$ covers $p(x)$ entirely. Reproduced from [18]. (bottom-row) A factorizing Gaussian distribution $q(\mathbf{x})$ is fitted to a multi-variate Gaussian distribution $p(\mathbf{x})$ by minimizing Amari's divergence with different values of α. When $\alpha \to 1$, $q(\mathbf{x})$ has high density where $p(\mathbf{x})$ has high density, but not the other way around. Thus, $q(\mathbf{x})$ tends to cover more of the entire distribution $p(\mathbf{x})$. By contrast, when $\alpha \to 0$, $p(\mathbf{x})$ has high density where $q(\mathbf{x})$ has high density, but not the other way around. Thus, $q(\mathbf{x})$ tends to cover less of the entire distribution $p(\mathbf{x})$.

$$\boldsymbol{\eta}^\star = \left(\frac{m_{\text{tr}}(\mathbf{x})}{v_{\text{tr}}(\mathbf{x}) + \sigma^2}, \frac{1}{v_{\text{tr}}(\mathbf{x}) + \sigma^2} \right)^{\text{T}}, \tag{14}$$

where $m(\mathbf{x})$ and $v(\mathbf{x})$ are given by (2) and (3), respectively, and where $m_{\text{tr}}(\mathbf{x})$ and $v_{\text{tr}}(\mathbf{x})$ are given by (11) and (12), respectively. See Appendix C for further details.

Furthermore, to approximate the expectation in (10), we generate samples of pairs of optimal locations and optimal values $\{\mathbf{x}^\star, y^\star\}$ from $p(\{\mathbf{x}^\star, y^\star\}|D_{t-1})$ using a random features approximation of the GP [26], as in [12,15]. Such a random features approximation allows to sample functions from the GP posterior. These functions can be optimized to obtain an approximate sample of $\{\mathbf{x}^\star, y^\star\}$. This is a common and efficient method often used in BO methods to sample from the posterior distribution and obtain optimal values. Given the samples, the expectation with respect to $p(\{\mathbf{x}^\star, y^\star\}|D_{t-1})$ in (10) is evaluated employing a Monte Carlo approach.

Using the approximations described in this section, the approximate acquisition function of AES is given by:

$$\tilde{a}_{\text{AES}}(\mathbf{x}; \alpha) = \frac{1}{(1-\alpha)\alpha} \left(1 - \frac{1}{S} \sum_{s=1}^{S} \exp\left\{ (\alpha - 1) g(\boldsymbol{\eta}) - \alpha g(\boldsymbol{\eta}_s^\star) + g((1-\alpha)\boldsymbol{\eta} + \alpha \boldsymbol{\eta}_s^\star) \right\} \right), \tag{15}$$

where we have considered S samples of $\{\mathbf{x}^\star, y^\star\}$ to approximate the expectation in (10) and $\boldsymbol{\eta}_s^\star$ are the natural parameters of the approximate conditional distribution $p(y|D_{t-1}, \mathbf{x}, \{y_s^\star, \mathbf{x}_s^\star\})$, for the sth sample of $\{\mathbf{x}^\star, y^\star\}$, denoted $\{y_s^\star, \mathbf{x}_s^\star\}$. In (15), α is a free parameter that will result in different acquisition functions. The accuracy of the proposed approximation is validated by the results of the experiments carried out in Section 5.1, where we compare the described approximation with the exact acquisition computed using a Monte Carlo method.

When $S \to \infty$ and $\alpha \to 1$ one should expect that $\tilde{a}_{\text{AES}}(\mathbf{x}) \to \tilde{a}_{\text{JES}}(\mathbf{x})$, where $\tilde{a}_{\text{JES}}(\mathbf{x})$ is given by (7), when a similar approximation is employed to that of AES. That is, $p(y|D_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})$ is also approximated using a truncated Gaussian distribution and the expectation is approximated by Monte Carlo. However, this need not be the case. The reason for this is that when $S \to \infty$, $\tilde{a}_{\text{AES}}(\mathbf{x}) \to \text{KL}(\tilde{p}(y, \{\mathbf{x}^\star, y^\star\}|D_{t-1}, \mathbf{x}) \parallel p(\{\mathbf{x}^\star, y^\star\}|D_{t-1}, \mathbf{x}) p(y|D_{t-1}, \mathbf{x}))$, where $\tilde{p}(y, \{\mathbf{x}^\star, y^\star\}|D_{t-1}, \mathbf{x})$ is an approximate joint distribution that results from the ratio between the approximate conditional predictive distribution $\mathcal{N}(y|m_{\text{tr}}(\mathbf{x}), v_{\text{tr}}(\mathbf{x}) + \sigma^2) \approx p(y|D_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})$ and the unconditioned predictive distribution $p(y|D_{t-1}, \mathbf{x})$.

By contrast, in general, when $S \to \infty$, we have that $\tilde{a}_{\text{JES}}(\mathbf{x}) \nrightarrow \text{KL}(\tilde{p}(\{\mathbf{x}^\star, y^\star\}, y|D_{t-1}, \mathbf{x}) \parallel p(\{\mathbf{x}^\star, y^\star\}|D_{t-1}) p(y|D_{t-1}, \mathbf{x}))$. The reason is that it is possible to show that in the JES approximate acquisition function, sometimes the exact joint distribution is used $p(\{\mathbf{x}^\star, y^\star\}, y|D_{t-1}, \mathbf{x})$, while other times the approximate joint distribution $\tilde{p}(\{\mathbf{x}^\star, y^\star\}, y|D_{t-1}, \mathbf{x})$ is used instead. Appendix D provides the details of JES that produce this behavior. Summing up, (15) results in a different approximation to that of JES when $\alpha \to 1$. However, the differences between both approximations are small according to our experiments.

3.3. An ensemble of acquisition functions

The acquisition function of AES, given in (15), depends on the parameter α. The selection of a value for such a parameter is non-trivial, as it affects the divergence, and hence the acquisition function. Specifically, different values of α may yield better or worse optimization performance, depending on the particular optimization problem. Moreover, in our experiments, we did not observe a particular value for α that generally performs better than the JES acquisition function.

Motivated by the aforementioned result and by the fact that we would like to avoid choosing specific values for α, we investigate here the possibility of considering simultaneously a range of values for $\alpha \in (0, 1)$. The result is an ensemble of acquisition functions, whose optimization results are expected to be better than those of the JES acquisition function. In particular, we consider eleven values for α equally spaced in the interval $(0, 1)$, where the first value is equal to 0.001 and the last value is equal to 0.999. This range of values has been employed before in the literature of approximate inference and provides good coverage of different α values [23,24,27,28]. Moreover, the value $\alpha = 0.999$ results in the direct KL-divergence, which should provide similar,

but not exactly the same results (for the reasons given in the previous section) to those of the JES acquisition function. The value $\alpha = 0.001$ is expected to result in the reversed KL-divergence. The range of values of α considered interpolates between these two divergences. We restrict to $\alpha \in (0, 1)$ since, empirically, values outside $(0, 1)$ often lead to numerical instabilities in the computations. This is particularly the case as α approaches $-\infty$ or exceeds 2. Furthermore, according to preliminary experiments, considering other values for α did not result in improved results. Last, in our ensemble of acquisition functions, we normalized each acquisition function by its maximum value, to give equal weight in the ensemble to each value of α. Specifically, the ensemble acquisition function is:

$$\tilde{a}_{\text{ens}}(\mathbf{x}) = \sum_{\alpha \in \Gamma} \frac{1}{w_\alpha} \tilde{a}_{\text{AES}}(\mathbf{x}; \alpha), \tag{16}$$

where $\tilde{a}_{\text{AES}}(\mathbf{x}; \alpha)$ is given by (15), Γ is a set with the 11 different values of α considered, and $w_\alpha = \tilde{a}_{\text{AES}}(\mathbf{x}^\alpha_{\text{max}}; \alpha)$ with $\mathbf{x}^\alpha_{\text{max}} = \arg\max_{\mathbf{x}} \tilde{a}_{\text{AES}}(\mathbf{x}; \alpha)$. Importantly, in each of the 11 different acquisition functions, we use the same S samples of $\{\mathbf{x}^\star, y^\star\}$, which are generated only one time instead of 11 times. Thus, the overhead of sampling $\{\mathbf{x}^\star, y^\star\}$ in this acquisition function is the same as that of JES.

Note that in (16) we normalize each different acquisition function by its maximum value. This is expected to make all acquisition functions have a comparable scale even if we are not guaranteed to find each α-divergence's global maximum. Without this normalization, divergences for small α can produce extremely large values, overshadowing acquisition functions with other values of α. By rescaling each criterion into a similar range (between 0 and 1), each α-based acquisition function has the same influence on the average. Furthermore, preliminary experiments in which we consider an adaptive weight for each α value following a similar approach to the one described in [29] did not result in improved results over (16).

The acquisition in (16) is a combination of several acquisition functions, in a similar way as machine learning ensemble methods combine several individual predictors to obtain a more robust aggregated predictor [19]. Notwithstanding, the principles of variance reduction that are typically used to motivate the good performance of traditional ensembles may not be directly applicable here. Nevertheless, (16) avoids relying on a single α value which may omit relevant information, since α influences the sensitivity of the divergence to the mismatch between probability distributions in different regions of the input space. By considering multiple α values, we expect to mitigate that sensitivity. This is similar to how an ensemble of models can yield predictions that are less sensitive to the peculiarities of any single model's fit. Namely, each acquisition function, for each α value, captures partially overlapping but different information, so the final average is less dependent on the choice of one particular α and is expected to perform better. This improvement in optimization performance is observed in our experiments, at least, with noiseless observations. This agrees with previous results from the Bayesian optimization literature that show that a portfolio allocation of acquisition functions is a robust approach to address optimization problems [30]. Finally, while it is true that we need not strictly optimize any single α-divergence by maximizing (16), the maximum of the average is still expected to be a high-informative point. The benefits of this approach are supported by the results obtained in our experiments which show that the ensemble method results in better solutions than those obtained by considering any single value of alpha in the noiseless evaluation setting.

Fig. 3 (bottom) shows a comparison of the AES acquisition function for different values of α, in a one-dimensional problem. The acquisition functions of JES and the ensemble method described in the previous paragraph are also displayed. For each acquisition function we also show its maximum. For the sake of visualization, each acquisition has been normalized so that its maximum is equal to 1. Fig. 3 (top) shows the predictive distribution of the GP and the location in the input space of the generated samples of $\{\mathbf{x}^\star, y^\star\}$. Here, we considered 32 samples of $\{\mathbf{x}^\star, y^\star\}$ and a noiseless setting. The figure shows that AES for $\alpha = 0.999$ gives similar values to JES, as expected, but not exactly the same ones. The reason for this is given in the previous section. We also observe that the peaks, *i.e.*, local maxima of JES and AES, for large values of α, often occur at the locations where the samples of $\{\mathbf{x}^\star, y^\star\}$ are found. This makes sense for JES since the entropy reduction is maximum there. Specifically, observing $\{\mathbf{x}^\star_s, y^\star_s\}$ reduces the predictive variance to almost zero at $\mathbf{x}^\star_s$ in the noiseless setting. Since AES is approximating JES for larger values of α, a similar behavior is expected for AES in such a setting. Thus, both acquisition functions generate peaks at the sampled points, effectively making both methods equivalent to Thompson Sampling when we only consider one sample of $\{\mathbf{x}^\star_s, y^\star_s\}$ [3]. We observe that the number of local maxima is significantly reduced for AES, for smaller values of α, and also for the ensemble acquisition function.

Table 1
Average number of local maxima for each method over 100 repetitions.

Method	# of local maxima
JES	17.60±0.56
Ensemble	12.53±0.49

The difference between the number of local maxima in the ensemble acquisition function and the number of local maxima in the acquisition function of JES is statistically significant, as shown in Table 1. This table displays the average number of local maxima for each acquisition function across 100 repetitions of the previous experiment. In each repetition, S different sets of samples of $\mathbf{x}^\star, y^\star$ are generated. The average number of local maxima for AES, with smaller values of α (data not shown), is also significantly lower than that of JES. However, in our experiments, these smaller α values did not significantly outperform JES in real-world problems; they only did so in synthetic problems. In practice, finding the global maximum is challenging, since optimization algorithms often converge to local minima. By averaging over α, the ensemble acquisition function becomes less rugged than when using α values close to 1.0, although not as smooth as when using α values near 0.1, as illustrated by Fig. 3. We believe that the averaging approach of the ensemble method yields a more robust acquisition function with fewer local maxima (a beneficial side effect) in the noiseless setting. Appendix E presents a similar analysis in the noisy setting, where the effect of local maxima is less pronounced because the presence of noise prevents the predictive variances at each $\mathbf{x}^\star_s$ from approaching zero.

In the experiments described in this section we have considered the global maxima of each individual acquisition, for each value of α, for normalization in Eq. (16). This is feasible since the optimization problem only has one dimension. In practice, however, only local maxima are guaranteed to be found using, *e.g.*, gradient ascent. Appendix F includes extra experiments where we analyze the impact of considering local maxima for the normalization weights in Eq. (16). The results obtained indicate that the impact of local maxima is small leading only to small changes in the weights of each individual acquisition function. The final aggregated ensemble acquisition function, however, remains similar and also has a significantly smaller number of local maxima than the acquisition function of JES, in the noiseless setting.

Finally, we note that the computational cost of AES for a particular α is the same as the one of JES because both methods use the same approximation to calculate the conditional predictive distribution $p(y|D_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})$. Therefore, the cost of evaluating the ensemble acquisition function is $O(|\Gamma|S)$, where $|\Gamma|$ is the number of α values considered (eleven in our setting) and S is the number of optimal samples of $\{\mathbf{x}^\star, y^\star\}$ generated. The ensemble method has to add to this cost the overhead of having to optimize each individual AES acquisition function, for each value of α, to obtain the individual

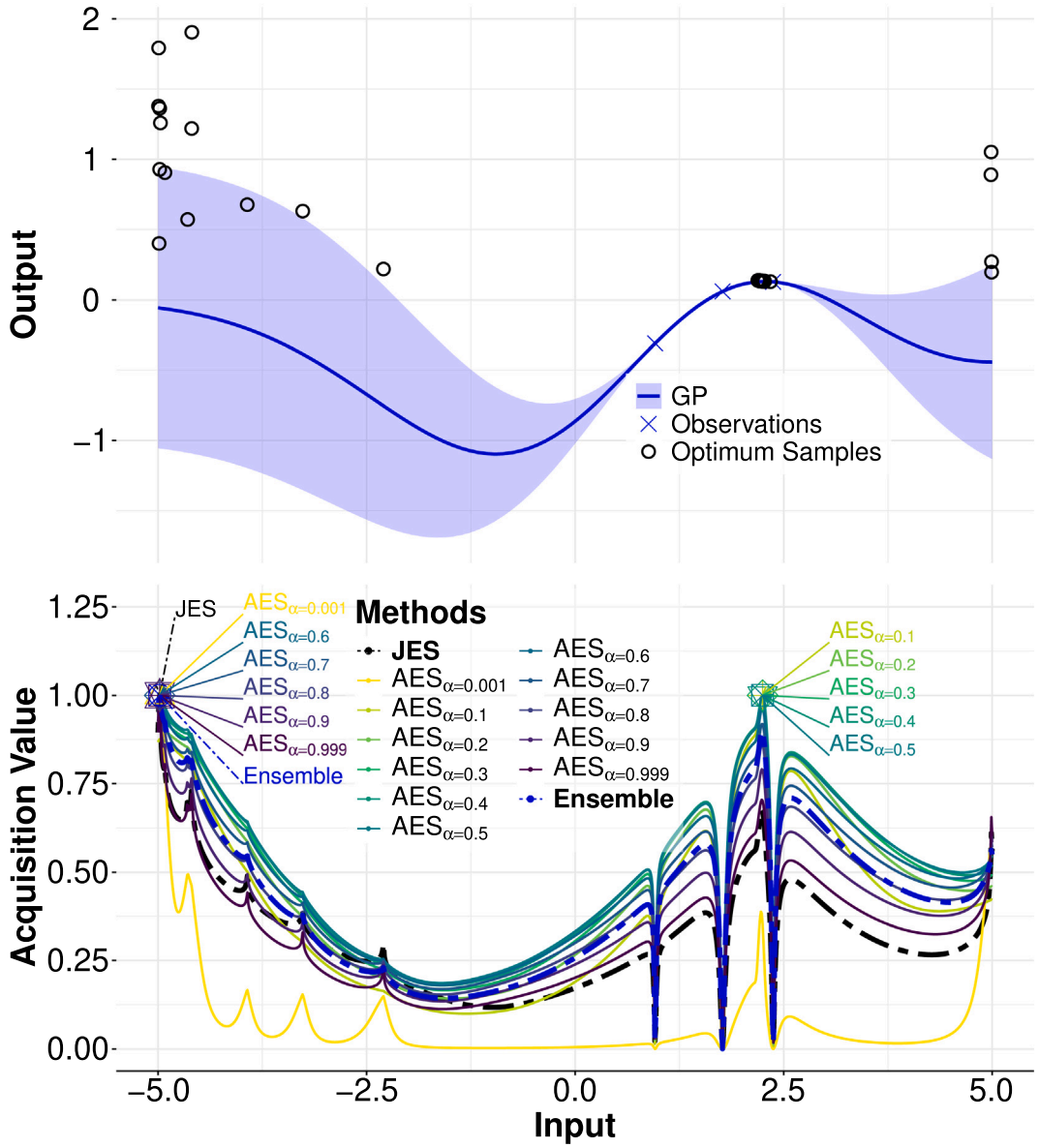

Fig. 3. (bottom) Comparison of AES for different α values, JES and the ensemble acquisition function. We also display the maximum of each acquisition function. (top) Predictive distribution of the GP and generated samples of $\{\mathbf{x}^\star, y^\star\}$. The acquisition functions have been normalized so that the maximum is equal to one for better visualization. Best viewed in color.

maxima described in (16). However, our results show that, in practice, the ensemble approach is only 3–6 times more expensive than JES. The total optimization cost is not 11 times higher than that of JES because the optimization of the acquisition function is only one step of the BO algorithm (see Algorithm 1). Specifically, the ensemble method only fits the GP model once, not 11 times, and it only generates samples of the optimum $\{\mathbf{x}^\star, y^\star\}$ once, not 11 times. Thus, the cost of sampling $\{\mathbf{x}^\star, y^\star\}$ is the same for both JES and the ensemble method. One may argue that since the ensemble method is more expensive than JES, one could increase the number of samples S in JES, at the same computational cost, to obtain better results. However, our experiments show that, in a noiseless evaluation setting, increasing S in JES does not lead to better results (see Fig. 7).

4. Related work

After introducing our proposed method, AES, and the associated ensemble method, we describe in this section related techniques from the literature and highlight the main differences of AES with respect to them. We particularly focus on information-based BO acquisition functions since our method falls in that category. Moreover, there is empirical evidence that information-based BO acquisition functions provide, in several problems, better results than other classical acquisition functions such as Expected Improvement or Upper Confidence Bound [12,14,15].

The Entropy Search (ES) strategy was considered first in [13], where the direct reduction of the current entropy of $\mathbf{x}^\star$ given the observed data was targeted. Since the estimation of the entropy of $\mathbf{x}^\star$ is intractable, an expensive sampling-based strategy based on discretizing the input space using a grid was considered. This method is practical, but makes difficult the direct optimization of the acquisition function. A set of candidate points has to be used. Furthermore, it is based on considering only the entropy of $\mathbf{x}^\star$, completely ignoring $y^\star$. Additionally, it does not consider the fact that the acquisition function is the mutual information between y and $\mathbf{x}^\star$ and does not swap these two variables to simplify its expression, resulting in a complicated acquisition function.

In [14] an alternative approximation of the ES acquisition function was considered. Instead of relying on a discretizing and sampling approach, as in [13], the expectation propagation (EP) algorithm was proposed to estimate the conditional entropy of $\mathbf{x}^\star$ after performing an evaluation at a new candidate point. This resulted in an approximation of the acquisition that is complicated and difficult to evaluate, although it provided gradients, unlike the approximation in [13]. This enables the use of gradient-based optimization algorithms to maximize the acquisition. In any case, the method still considered only $\mathbf{x}^\star$ and ignored $y^\star$. Furthermore, it did not simplify the acquisition using the mutual information trick we consider here, which allows to swap $\mathbf{x}^\star$ and y.

In [12] it is proposed to simplify the expression for the acquisition function of ES by swapping $\mathbf{x}^\star$ and y, as a consequence of the symmetry of the mutual information. The resulting acquisition function is known as Predictive Entropy Search (PES). PES results in a much simpler expression for the acquisition function than the one in [14]. The difficulty is still found in approximating the conditional predictive distribution given a sample of $\mathbf{x}^\star$. The samples of $\mathbf{x}^\star$ are obtained by optimizing functions generated from the GP posterior. For this, a random feature approximation of the GP is used. To approximate the conditional predictive distribution, again, the EP algorithm is employed. EP is expensive and consists in approximating with Gaussians several non-Gaussian factors introduced to guarantee compatibility with the sample of $\mathbf{x}^\star$. PES ignored the value of $y^\star$. Empirical evidence shows that PES performs much better than ES.

In [21,31], it is considered the entropy of the optimum in the output space, $y^\star$, instead of the optimum in the input space, $\mathbf{x}^\star$. Specifically, it is suggested to select the next evaluation as the one that minimizes the expected entropy of $y^\star$ the most. This method is known as Max-value Entropy Search (MES). As in PES, the acquisition function is simplified using the mutual information trick. The main advantage of MES is that the conditional predictive distribution has to be computed with respect to $y^\star$, instead of $\mathbf{x}^\star$, which significantly simplifies the evaluation of the acquisition. Specifically, given a sample of $y^\star$, such a conditional distribution can be approximated using a truncated Gaussian distribution. MES gives similar to or better results than those of PES and performs better with respect to the number of samples of $y^\star$ [21].

MES is computationally expensive due to the fact that it requires sampling the optimum of the optimization problem at each step using a random features approximation of the GP. Furthermore, such an approximation is only valid for particular GP covariance functions. The Fast Information-theoretic Bayesian Optimization (FITBO) [32] is an alternative approach that avoids sampling the global optimum. For this, extra approximations are introduced, based on expressing the unknown objective function in a parabolic form. This introduces an extra hyper-parameter, but circumvents the process of sampling the global minimum. After more approximations, the entropy reduction of FITBO is purely analytical, being computationally fast. FITBO also considers the expected reduction in the entropy of $y^\star$, as MES.

There are some methods proposed to alleviate some limitations of PES and MES. For example, the Trusted Maximizers Entropy Search (TES) method [33] reduces the overall cost of PES. TES selects the next point to evaluate from a finite set of trusted maximizers. These trusted maximizers are inputs optimizing functions that are sampled from the GP posterior. Evaluating TES requires either only a stochastic approximation with sampling, or a deterministic approximation with EP. TES provides similar or better results than those of PES at a smaller computational cost.

We believe that the described improvements of MES and PES are orthogonal to our work and they could in principle be also used in AES.

Another improvement is the rectified version of MES (RMES) [25], which solves known issues of MES in the noisy setting, where the conditional predictive distribution is a truncated Gaussian random variable that, when contaminated by Gaussian noise, lacks a closed-form density. RMES addresses this issue by obtaining a closed-form density for the conditional distribution using the reparametrization trick. RMES gives similar or better results than MES. As described in Section 3.2, in our experiments, in the noisy evaluation setting, we use the approximation of the conditional predictive distribution introduced in [15]. Such a correction is also used in the implementation of MES that we use in our experiments.

Recently, joint entropy search (JES) has been proposed as an improvement over MES and PES, presenting state-of-the-art optimization performance [15,16]. Concretely, this approach considers the entropy over the joint distribution of both the global optimum in the input space and the output space. Namely, $\{\mathbf{x}^\star, y^\star\}$. The acquisition function simply chooses the next point to reduce the most the expected entropy of that random variable. The expression for such an acquisition is simplified using the mutual information trick, by swapping $\{\mathbf{x}^\star, y^\star\}$ and y, and the conditional predictive distribution is approximated by a Gaussian with the same moments of a truncated Gaussian distribution contaminated by Gaussian noise (in the noisy setting). JES gives similar or better results than those of MES and PES [15].

All the strategies described so far in this section result in an acquisition function $a(\mathbf{x})$ that estimates the mutual information between the solution of the optimization problem (*i.e.*, $\mathbf{x}^\star$, $y^\star$ or $\{\mathbf{x}^\star, y^\star\}$) and y, the observation at $\mathbf{x}$. As described in Section 3, this can be shown to be equivalent to the Kullback–Leibler (KL) divergence between two probability distributions. By contrast, the proposed strategy AES, and the associated ensemble method, replace this divergence with the more general α-divergence [17], which includes a parameter α. The α-divergence generalizes the KL-divergence. Specifically, the α parameter can be used to give higher importance to particular differences among the probability distributions, and when $\alpha \to 1$, it results in the regular KL-divergence used in ES methods.

We are not aware of the use of α-divergences in the context of BO. However, the use of Shannon's entropy in ES has been generalized in [34] to a broader class of loss functions that result in other acquisition functions from the literature such as Knowledge Gradient or Expected Improvement. For this, it is suggested the use of decision-theoretic entropies parameterized by a problem-specific action set $\mathcal{A}$ and a loss function ℓ. In the case of ES, the loss is the negative logarithm, and the action set contains the GP posterior distribution. In spite of this, there is not a clear and direct way of using the framework of [34] to obtain the acquisition function of AES or the ensemble method we propose here.

In the literature, there are several works that have used α -divergences for approximate Bayesian inference. These works also consider values of α in the interval $(0, 1)$. In particular, in [23] the approximate minimization of α-divergences is used to approximate the posterior distribution of Bayesian neural networks (NN) with a Gaussian distribution. The method is known as Black-box alpha. The results obtained show that intermediate values of α, *e.g.*, $\alpha = 0.5$, lead to better results than other values of α that result in the approximate minimization of the KL-divergence. The minimization of α-divergences in the context of dropout for approximate inference in Bayesian NN is explored in [35]. A generalization of Black-box alpha is considered in [24], where the approximate distribution is a flexible implicit distribution obtained by letting some noise go through a deep NN. The results obtained show that, in general, larger values of α may lead to better predictive distributions in the case of regression problems, and smaller values of α may lead to better test mean squared error. The use of α-divergences to approximate the posterior distribution has also been studied in the case of GPs for regression and binary classification [36], GPs for multi-class classification [27] and deep GPs [28]. In such a setting, one often observes that $\alpha \approx 1$ tends to give better predictive distributions, while $\alpha \approx 0$ tends to minimize the test mean squared error. The use of α-divergences for approximate inference in generative

models has also been explored in [37,38], with superior results to those obtained by the KL-divergence.

The use of several acquisition functions to solve an optimization problem has been considered before. In [29] the authors propose a strategy to sample from a pool of acquisition functions the one to use at each iteration of the BO algorithm. The idea is to favor those acquisition functions that lead to better results at each iteration and to penalize those that do not. The resulting method is called GP-hedge. A limitation is, however, that GP-hedge assumes there is an optimal acquisition function in the pool of acquisition functions considered. Specifically, GP-hedge does not combine the acquisition functions. We evaluated GP-hedge via preliminary experiments and compared its performance with respect to our ensemble method based on AES. The ensemble method performed better, probably as a consequence of combining the acquisition functions instead of simply choosing one among them at each iteration according to some weights. Finally, in [30], entropy search is employed to choose, at each iteration, the strategy from the pool of strategies. This avoids the problem of choosing strategies based on their past performance that was inherent in [29].

5. Experiments

In this section, we compare, across several optimization problems, AES and the ensemble method with other strategies for BO. Namely, we compare results with random search, and the acquisition functions based on the KL-divergence described in Section 4, *i.e.*, JES, MES, and PES. Random search simply chooses randomly the next point to evaluate. We also compare results with Expected Improvement (EI), which is used as a base-line method [39]. We do not compare results with other BO methods such as Upper Confidence Bound since several works from the literature already compare them with PES, MES and JES, showing better results in several optimization problems [12,15, 21]. In the ensemble method, we consider eleven values for α, *i.e.*, $\{0.001, 0.1, 0.2, \ldots, 0.9, 0.999\}$. Our implementation of AES and the ensemble method are available at https://github.com/fernandezdaniel/alphaES. For the other acquisition functions, EI, PES, MES, and JES, we simply used the implementation provided in BOTorch [20]. In all problems, the goal is to maximize the objective. Minimization can be simply achieved by optimizing $-f(\mathbf{x})$.

In our experiments, we use $S = 32$ samples of the problem's solution to estimate the acquisition of AES, PES and MES. These samples are generated using a random feature approximation of the GP, as described in [12,26]. We use a Matérn 5/2 covariance function with ARD and fit the GP via maximum marginal likelihood. These are standard choices in BOTorch. In each optimization problem, unless indicated differently, we use 10 randomly chosen initial observations of the objective. This number of initial observations is expected to guarantee that the maximum marginal likelihood approach used to fit the GP does not result in overfitting. To maximize each acquisition, we use L-BFGS-B with 1 restart and 200 points to generate the initial conditions from which the starting point of the optimization is randomly chosen. See [20] for further details. Preliminary experiments in which we increase the number of restarts and the number of points used to generate the initial conditions give similar results when comparing the different methods. See Appendix H. We report average results across 100 repetitions of the experiments with different random seeds and show the corresponding error bars. At each iteration, the BO method recommends the best observation, in the noiseless setting. In the noisy setting, we recommend the observation with the best predictive mean. This is done to remove the observational noise. In general, this gives better results than optimizing the GP mean to make a recommendation.

5.1. Quality of the approximation of the acquisition function

We investigate in this section the accuracy of the approximation of the AES acquisition function suggested in (15). For this, we compare in a 1-dimensional toy problem the AES acquisition function with the exact acquisition function it targets, estimated using a more accurate Monte Carlo method. The α values considered for AES and the exact method are the same for comparison. Since the problem is one-dimensional, the calculation of the exact acquisition is feasible. Specifically, for each sample of $\{\mathbf{x}^\star, y^\star\}$, $\{\mathbf{x}_s^\star, y_s^\star\}$, the conditional density $p(y|\mathcal{D}_{t-1}, \mathbf{x}, \{y_s^\star, \mathbf{x}_s^\star\})$ is estimated using a kernel density estimator on samples from the GP posterior at $\mathbf{x}$ that are compatible with $\{\mathbf{x}_s^\star, y_s^\star\}$. The integral with respect to y in (10) is estimated using quadrature, since it is one-dimensional. We consider a large number of samples $S = 6000$ of $\{\mathbf{x}^\star, y^\star\}$ to approximate the expectation in (10). No significant changes are observed above that number of samples. Note that these operations are significantly more costly than the evaluation of AES via (15) for any value of α, and become intractable in general. However, they are expected to give a good estimate of the AES acquisition function in this simple problem. We also compare the ensemble acquisition function described in (16) with the exact ensemble acquisition function, estimated by a weighted average of the exact AES acquisition for the 11 values of α described above. For simplicity, we assume a noiseless evaluation setting.

Fig. 4 (top-left) shows the GP predictive distribution for the objective and the observed data considered. From (top-right) to (bottom-left) we compare, for a representative set of values for α, our proposed approximation of the AES acquisition function with the exact acquisition estimated as described above. We observe that the exact method and the proposed approximation are very similar and have the local maxima near the same locations. However, it is possible to observe that the proposed approximation tends to underestimate the exact acquisition function and that the approximation is worse for values of α closer to zero. The under-estimation of the exact acquisition function has also been observed in other information-based strategies for BO such as PES [40,41]. We also observe that changing the value of α has an impact on the shape of the acquisition function in particular regions of the input space (*i.e.*, when $x > 4$). Fig. 4 (bottom-right) compares the proposed approximation for the ensemble acquisition function with the exact acquisition function. We observe that the exact method and the proposed approximation are almost identical in this case, sharing the same local maxima and minima, with only small differences at particular points of the input space.

Appendix G includes extra experiments comparing the quality of the proposed approximation with the exact acquisition function when using only $S = 32$ samples in the approximation. This is the number of samples employed in our experiments. The results obtained indicate that with such a small number of samples the proposed approximation is still similar to the exact acquisition, for most values of α, and also for the ensemble acquisition function.

5.2. Synthetic experiments

We carried out several synthetic experiments in which the objective is sampled from a GP and hence there is no model bias. We consider 4 experiments with a different number of input dimensions. Namely, 4, 6, 8, and 12 dimensions. In each experiment, we consider two scenarios: one with noiseless evaluations and another with evaluations contaminated by standard Gaussian noise with a variance of 0.1. We consider 100 repetitions of the experiments and report average results with the associated error bars. We assess the performance of each method by measuring the relative difference (on a logarithmic scale) between the recommendation's value in the noiseless objective function and the optimal value, relative to the number of evaluations performed. The optimal value for each problem is found via gradient optimization using a grid of size $D \times 10,000$ to choose the starting point, where D is the dimension of the problem.

Fig. 5 shows the results obtained by AES, for each value of α considered, and the ensemble method on the noiseless synthetic problems, for each number of inputs dimensions in 4, 6, 8, and 12. We also report

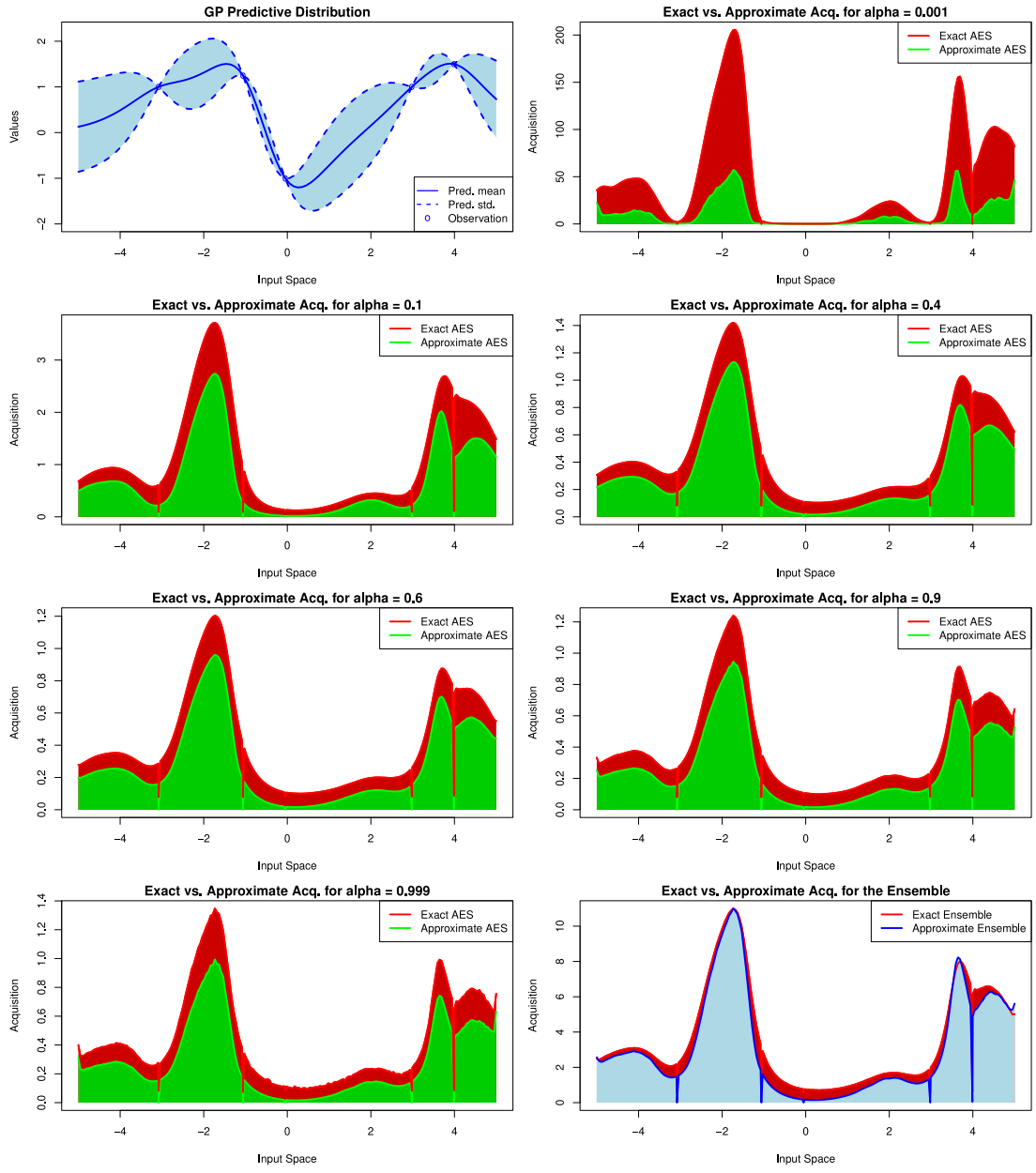

Fig. 4. (top-left) GP predictive distribution for the objective. From (top-right) to (bottom-left) acquisition function AES, when using the proposed approximation, and using a method that is expected to give the exact acquisition. Both computed using $S = 6000$ samples of $\{\mathbf{x}^\star, y^\star\}$. We report results for a representative set of α values. (bottom-right) Acquisition function for the ensemble method using the proposed approximation and a method that is expected to give the exact acquisition. Best viewed in color.

the results of EI, JES and the random search strategy. Among the BO methods, the ensemble method consistently gives the best performance with respect to the number of evaluations performed, followed by AES for different values of α. By contrast, JES gives slightly worse results followed by EI and the random search strategy, which is the worst overall method. The fact that the ensemble method outperforms AES for all values of α shows the benefits of the ensemble strategy. Specifically, combining the different acquisition functions for a range of values of α is better than using a single α value. We observe that in the 4-dimensional experiment, the difference in performance between the ensemble method and the other AES variants is smaller. However, when the number of dimensions grows, these differences increase. We believe that the differences are small in the first problem because all methods reach the optimal solution. In the experiment with 6 input dimensions, the differences with respect to the ensemble method become more significant. Here, AES variants also outperform JES, but they converge after approximately 300 evaluations with minimal further improvement. By contrast, the ensemble method still keeps giving better solutions, on average, with extra evaluations. In the experiment with 8 dimensions, a similar behavior is observed, but the differences with respect to the ensemble method become larger. Additionally, here, intermediate values of α within the range $0.2 \leq \alpha \leq 0.5$ perform slightly better at the beginning than other AES variants. Finally, in the 12-dimensional experiment, similar results are observed. The performance of AES with intermediate values of α gives better results at the beginning of the optimization process and outperforms JES. However, the ensemble method obtains overall better results. In this problem, all methods need more evaluations to reach closer solutions to the global maximum.

Fig. 6 shows the results of each method on the noisy evaluation setting, for each input dimensionality, *i.e.*, 4, 6, 8, and 12 dimensions. Again, random search exhibits the worst performance, followed by

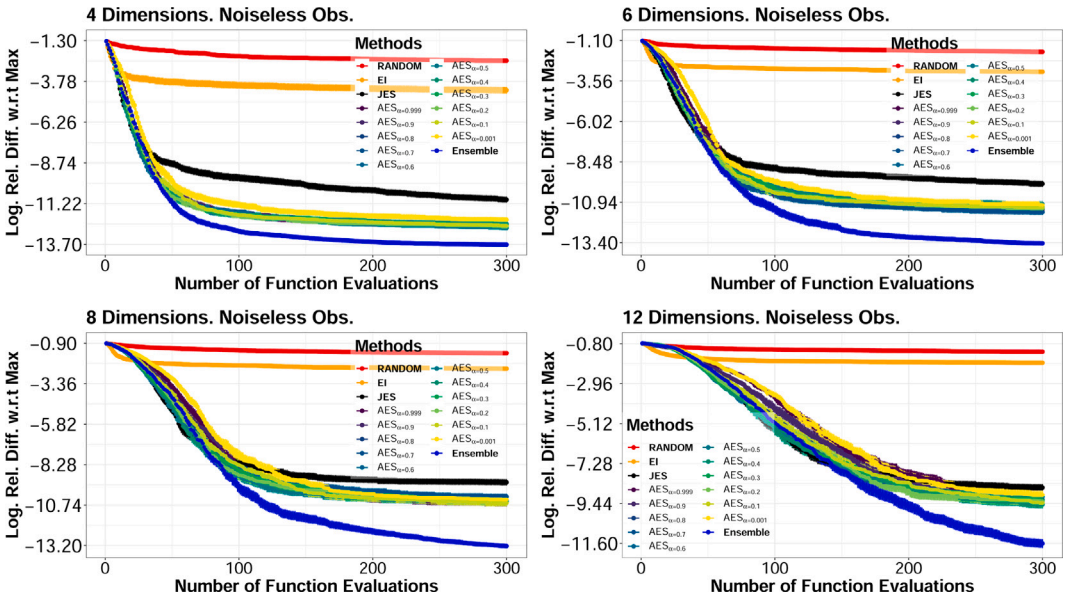

Fig. 5. Average logarithm relative difference between the objective at each method's recommendation and the objective at the global maximum, with respect to the number of evaluations. Results are shown for the 4, 6, 8, and 12 dimensional problems. Observations are noiseless. Best viewed in color.

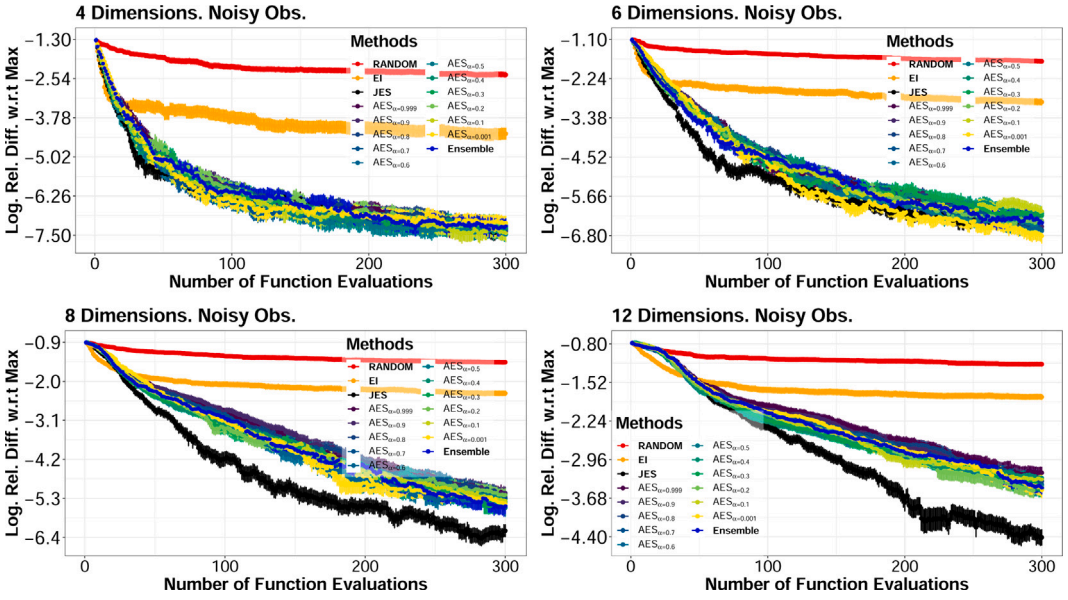

Fig. 6. Average logarithm relative difference between the objective at each method's recommendation and the objective at the global maximum, with respect to the number of evaluations. Results are shown for the 4, 6, 8, and 12 dimensional problems. Observations are noisy. Best viewed in color.

EI. Among the BO methods, there are no significant differences in the problems with 4 and 6 dimensions. However, as the number of dimensions increases, JES performs better than the other strategies. Additionally, no single α value in AES is consistently better, and the ensemble method performs similarly to the other α-divergence based methods. We believe that these results could be explained because in the noisy setting, the beneficial properties of the ensemble method are smaller and the performance of JES is not impaired by local maxima in the acquisition function. See Appendix E for further details.

These synthetic experiments illustrate the beneficial properties of the ensemble method, which considers a range of values for α, when compared to the AES method that exclusively considers a particular value of α. Specifically, the ensemble method always performs similarly or better than AES. Therefore, in the remaining experiments, we will consider exclusively the ensemble method.

Table 2 displays the average execution time in seconds per BO iteration for each method in the synthetic experiments in the noiseless setting. Appendix I shows similar results for the noisy setting. Note that we do not include here the time of the RANDOM method since it simply chooses at random the next evaluation. The fastest method is EI, whose cost is very low as it does not need to generate samples of the optimum $\{\mathbf{x}^\star, y^\star\}$, unlike the other strategies. On the other hand, the methods that sample from the optimum (PES, and $\text{AES}_{\alpha=0.999}$) exhibit comparable times. Notably, the ensemble method is the slowest, as expected. The extra cost of the ensemble method is due to optimizing the AES acquisition function for 11 different values of α. Remarkably, this extra cost only results in the ensemble method being approximately $3-4$ times more expensive than JES per iteration. The reason for this is that computing the acquisition function is only one of the steps required at each iteration. More precisely, there are other steps that are common

Table 2
Average execution time and standard error per iteration (in s) in the noiseless synthetic experiments.

	4D	6D	8D	12D
EI	**3,432±0,058**	**4,279±0,201**	**5,866±0,206**	**7,139±0,319**
JES	7,898±0,419	9,703±0,392	12,741±0,506	17,148±0,637
$AES_{\alpha=0.001}$	7,385±0,308	9,483±0,452	12,650±0,558	17,512±0,594
$AES_{\alpha=0.1}$	7,579±0,256	9,731±0,455	12,740±0,564	17,564±0,649
$AES_{\alpha=0.2}$	7,682±0,242	9,052±0,396	12,682±0,489	17,859±0,818
$AES_{\alpha=0.3}$	7,409±0,264	9,587±0,419	12,586±0,507	18,085±0,789
$AES_{\alpha=0.5}$	7,042±0,291	9,627±0,349	12,899±0,419	17,817±0,596
$AES_{\alpha=0.5}$	7,262±0,329	9,580±0,414	13,036±0,554	17,455±0,634
$AES_{\alpha=0.6}$	7,167±0,350	9,216±0,452	13,016±0,439	18,258±0,733
$AES_{\alpha=0.7}$	6,751±0,322	9,125±0,386	13,127±0,484	17,441±0,540
$AES_{\alpha=0.8}$	6,833±0,309	9,225±0,421	13,225±0,513	16,972±0,838
$AES_{\alpha=0.9}$	7,131±0,328	9,585±0,383	12,603±0,617	17,646±0,561
$AES_{\alpha=0.999}$	7,062±0,307	9,868±0,439	12,523±0,498	18,701±0,712
Ensemble	30,097±1,233	36,985±1,463	42,338±1,266	51,836±1,994

to all methods, such as fitting the GP model and sampling $\{\mathbf{x}^\star, y^\star\}$ (recall that the ensemble method uses the same samples of $\{\mathbf{x}^\star, y^\star\}$ for each α value). Across all methods, the low standard errors indicate that the execution times are quite stable, with particularly minimal variability for most methods.

5.3. Impact of the number of samples

We investigate the impact of the number of samples S considered in the performance of our ensemble method when approximating the expectation in (15). For this, we consider the 4-dimensional and the 8-dimensional synthetic problems described before. We report the performance of the ensemble method for increasing values of S, from 1 to 64. Recall that in such a method the generated samples are re-used for each value of α considered. That is, we only generate S samples once, instead of S samples for each value of α. We compare results with JES for the same number of generated samples S. We consider both a noiseless and a noisy evaluation scenario.

Fig. 7 shows the results obtained for each method and problem considered. Remarkably, in the 4-dimensional problem and the noiseless setting, the performance of JES deteriorates as the number of samples increases, whereas the performance of the ensemble method remains very similar, independently of the number of samples considered. This behavior of JES is mitigated in the 8-dimensional problem, where varying the number of samples yields similar results for all values of S. In the noisy evaluation setting, the performance of the ensemble method is also little dependent on the number of samples considered. However, the behavior of JES is a bit different and increasing S improves the results.

We believe that the phenomenon in which the performance of JES deteriorates as the number of samples S increases, is related to the large number of local minima in the JES acquisition, as described in Section 3.3, and illustrated in Fig. 3. Specifically, as S increases, JES generates more local maxima in the acquisition function in the noiseless setting. A large number of local maxima makes more it likely that the optimization of the acquisition function is trapped in a sub-optimal solution. This will force JES to explore more extensively the input space. This behavior is good in high-dimensional problems, where extensive exploration is necessary. However, in low-dimensional problems, this excess exploration prevents JES from adequately exploiting solutions in promising regions. By contrast, increasing the number of samples does not have this detrimental effect on the ensemble method. Since the number of local minima in the acquisition of the ensemble method is smaller in the noiseless setting, its trade-off between exploration and exploitation is expected to be less affected by S. In noisy problems, the conditional distribution $p(y|\mathcal{D}_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})$ does not have zero variance at the sampled solutions. Therefore, JES does not have that many local optima and the aforementioned behavior does not happen.

Summing up, in the ensemble method S has little effect on the final performance. By contrast, in JES, increasing S slightly deteriorates or gives similar results in the noiseless evaluation setting. In the noisy evaluation setting, increasing S slightly improves the results of JES in high-dimensional problems.

Appendix I shows the average cost per iteration of JES and the ensemble method as the number of samples S changes. The results obtained are very similar to those reported in the previous section.

5.4. Benchmark experiments

In the previous sections, we considered that the objective function is generated from a GP. This means that there is no model bias and the probabilistic model used in the BO loop can perfectly fit the objective. In general, the objective need not be generated from a prior GP and model bias can have an effect on the performance. Therefore, in this section, we consider several benchmark functions that are often used in optimization problems and that are not generated from a GP prior. Namely, Hartmann-3D, Hartmann-6D, Styblinski-Tang-4D, and Cosine-8D [42,43]. Note that each objective function is followed by the number of dimensions it depends on.

Using the aforementioned objectives, we carry out experiments and compare the performance of the ensemble method with that of JES, PES, MES, and a random search strategy. We exclude AES for specific values of α because the ensemble method consistently achieved equal to or better results in the synthetic experiments. As before, we consider both noiseless and noisy scenarios, contaminating observations in the noisy setting with additive Gaussian noise with variance 0.1. Again, we measure the performance of each method in terms of the relative difference (in a logarithmic scale) between the objective at the recommendation and the global maximum. We report average results over 100 repetitions of the experiments in which the initial observations differ.

The results obtained are displayed in Fig. 8. The figure shows that the ensemble method achieves the best performance in 3 of the 4 objectives considered. In the noiseless setting, the ensemble method is always equal or better than JES. In the noisy setting, JES performs better than the ensemble method in Cosine-8D. In this problem the GP model does not perform very well since the obtained solutions are the furthest away from the optimum among the 4 problems considered. Here, MES is the best performing method both in the noiseless and the noisy setting. Random is the overall worst performing method in each problem and setting, followed by EI and PES. The summary of these experiments is that the ensemble method performs very well in the noiseless evaluation setting and is competitive with state-of-the-art methods for BO based on information theory. In the noisy evaluation setting the benefits of using the ensemble method are smaller. These results are compatible with the ones observed in the previous sections.

In these experiments, we also measured the average time required per BO iteration by each method. The results obtained for the noiseless setting are displayed in Table 3. Appendix I shows the results for the noisy setting which are very similar to the ones reported here. We observe that, again, the fastest method is EI. Among information-based acquisition functions, MES is the fastest method followed by JES. PES is slower than MES and JES probably as a consequence of having to run the expectation propagation algorithm to approximate the conditional predictive distribution at each iteration. Finally, the slowest method is the proposed ensemble method, as expected. However, the ensemble method is only approximately 4–6 times slower than JES, even though it maximizes 11 different acquisition functions, one for each value of α. Again, the reason for this is that computing the acquisition function is only one of the steps required at each iteration. There are other steps that are common to all methods, such as fitting the GP model and sampling $\{\mathbf{x}^\star, y^\star\}$ (recall that the ensemble method uses the same samples of $\{\mathbf{x}^\star, y^\star\}$ for each α value).

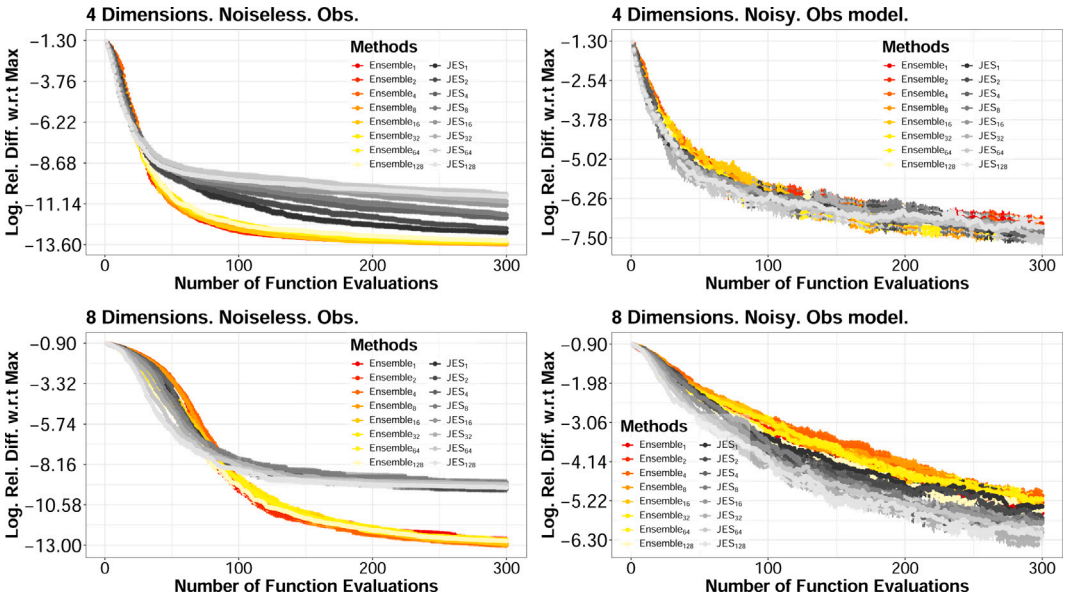

Fig. 7. Average logarithm relative difference between the objective at each method's recommendation and the objective at the global maximum, with respect to the number of evaluations. Results are shown for the 4, and 8-dimensional problems. The number of samples S ranges from 1 to 64. Best viewed in color.

Table 3
Average execution time and standard error per iteration (in s) in the noiseless benchmark experiments.

	Hartman3D	Styblinskitang4D	Hartman6D	Cosine8D
EI	**2,090±0,076**	**3,067±0,095**	**4,485±0,179**	**4,742±0,198**
PES	6,218±0,244	10,958±0,635	9,837±0,140	16,385±0,571
MES	4,125±0,166	5,820±0,251	7,581±0,244	8,538±0,326
JES	6,418±0,249	8,273±0,305	9,075±0,316	9,407±0,335
Ensemble	34,320±0,924	32,335±1,172	39,377±0,785	46,416±1,386

Table 4
Characteristics of the UCI datasets employed in the experiments.

Dataset	# Instances	# Features	# Classes
Pima	768	8	2
Image	2310	19	7
Defects	1109	21	2
Liver	583	10	2
Australian	690	14	2
Ionosphere	351	34	2

5.5. Real world experiments

We evaluate the performance of the ensemble method, JES, PES, MES, and random search in real-world experiments. We do not compare results with EI given that it did not perform very well in the previous experiments. Here, we consider tuning five hyper-parameters of a neural network for classification with two hidden layers. The hyper-parameters considered are the number of hidden units in each layer, the batch size, the amount of ℓ_2 regularization, the learning rate, and the number of training epochs. Again, we do compare results with AES for specific values of α since the ensemble method performs similarly or better in synthetic experiments. The network's accuracy is estimated using 5-fold cross-validation, and the optimization process is run for 200 iterations. As in the previous experiments, we report the performance of each method as the relative difference (in a logarithmic scale) between the objective at the recommendation, and the best objective value observed (across each method). Here, we use 25 randomly chosen initial observations. This number of initial observations is expected to guarantee that the maximum marginal likelihood approach used to fit the GP does not result in overfitting. We report average results across 100 repetitions of the experiments. We consider 6 different classification datasets extracted from the UCI repository [44]. Namely, Pima, Image, Defects, Liver, Australian, and Ionosphere. Table 4 shows the characteristics of these datasets.

Fig. 9 shows the results obtained. Note that in these experiments the variability from one repetition to another is very large, which is translated in high error bars and smaller differences among methods. Furthermore, the performance of random search is very close to that of the compared methods in some datasets. This indicates that the GP could not be a very good model in these problems and that model bias may play an important role. In spite of this, we observe that the ensemble method generally achieves good performance results. Specifically, it performs better than JES in Pima, Australian and Ionosphere, although the differences are small. In the other datasets it gives similar results to JES. The ensemble method is also, in general, comparable to or slightly better than the other methods. The only exception is the Ionosphere dataset, where MES performs better. However, MES performs poorly on the Defects dataset, where it is outperformed by random search. JES performs well on the Image dataset, but encounters difficulties on Pima and Australian. PES consistently lags behind, being the worst-performing information-based BO method. Finally, random search performs the worst overall, especially in the Pima and Ionosphere datasets. Summing up, despite the noise in these experiments, the ensemble method attains good results in these problems, achieving results that are similar and sometimes better than those of the state-of-the-art.

6. Conclusions

This paper has introduced Alpha Entropy Search (AES), a method for Bayesian optimization (BO) whose acquisition function formulation is based on information theory. Specifically, AES generalizes previous methods that aim at choosing the next evaluation point as the one that is expected to minimize the most the entropy of the solution of the problem. AES measures the level of dependency between the objective at the candidate point to evaluate, y, and the problem's solution $\{\mathbf{x}^\star, y^\star\}$ both in the input and the output space. For this, the α-divergence is used instead of the typical KL-divergence of information-based methods. The α-divergence has a parameter α that trades-off evaluating differences between each distribution at a single mode and evaluating differences

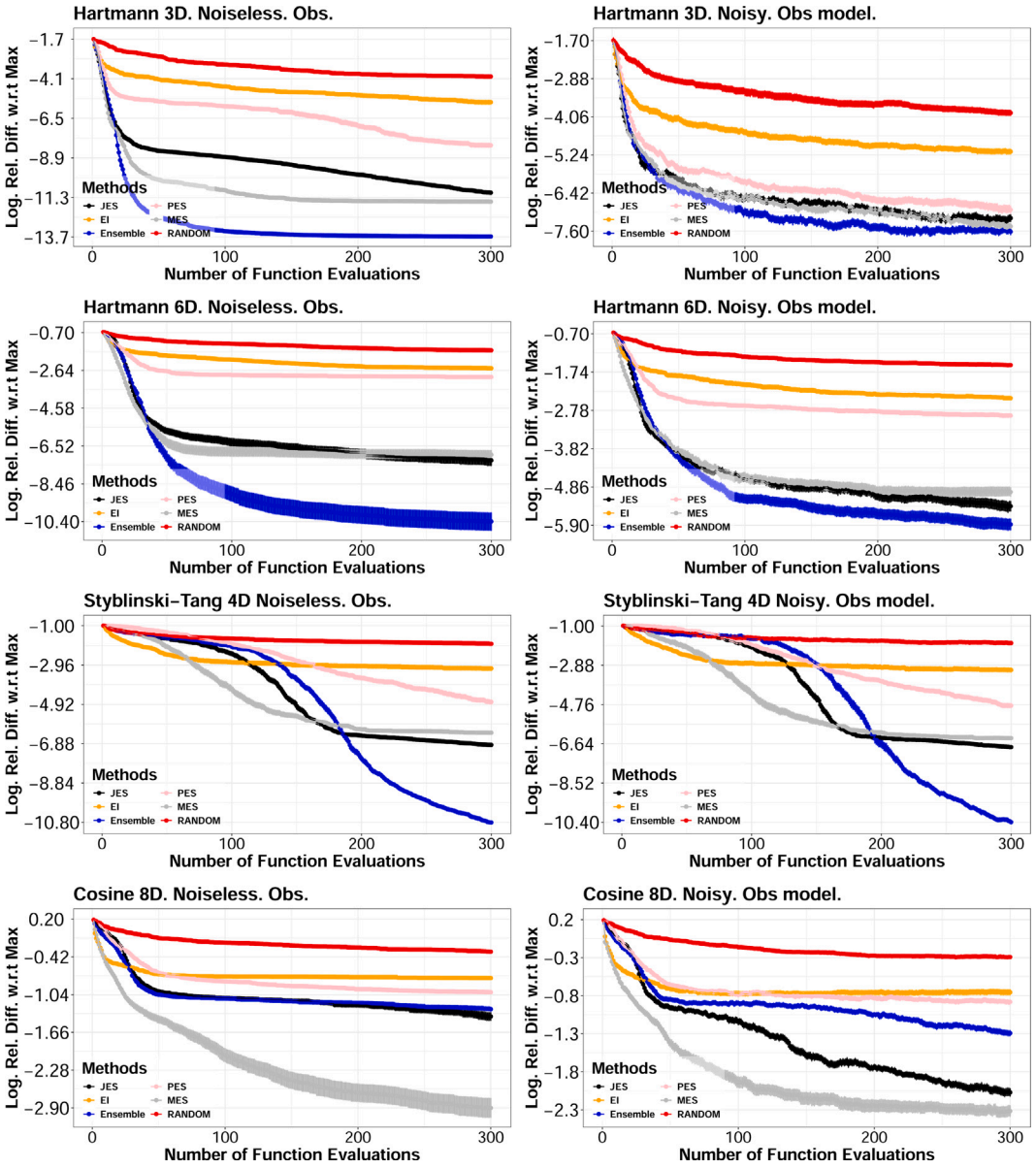

Fig. 8. Average logarithm relative difference between the objective at each method's recommendation and the objective at the global maximum, with respect to the number of evaluations. We show the results for the 3-dimensional Hartman problem, the 6-dimensional Hartman problem, the 4-dimensional Styblinski-Tang problem and the 8-dimensional Cosine problem. We consider noiseless (left-column) and noisy observations (right-column). Best seen in color.

globally. We did not find a particular value of α that generally provided the best overall results. Therefore, we considered an ensemble method that simultaneously considers a range of values for α.

Our experiments in synthetic, benchmark and real-world problems show that the ensemble method performs better than considering a single value of α and that it provides competitive results with the state-of-the-art methods for information-based BO. Namely, JES, MES and PES. This is particularly the case in a noiseless evaluation setting. In a noisy evaluation setting, however, the differences among methods are smaller and our proposed method gives similar results to those of the state-of-the-art. More precisely, one would decide to choose the ensemble method or other alternatives in these situations:

- Noiseless problems: the ensemble method, most of the time, outperforms JES, PES and MES. Only in Cosine8D we observe MES to give better results.
- High-dimensional or high-noise problems: in problems with dimensions higher than 8 or with substantial noise, the performance of the ensemble method can be inferior to that of JES.
- When the GP fails to model the objective: for example, in the Cosine 8D experiment, MES performed particularly well while the ensemble underperformed. This is most likely due to the GP model failing to accurately capture properties of the true underlying function, as evidenced by MES achieving only −2.9 after 300 iterations in the noiseless setting. This indicates that the solution is still far from the optimum suggesting that the GP is a poor model.

In the 5-dimensional real-world experiments, the ensemble method consistently ranked as either the best or the second-best method, whereas JES and MES exhibited greater variability. Summing up, these observations suggest that when the GP is a reliable model and noise is minimal

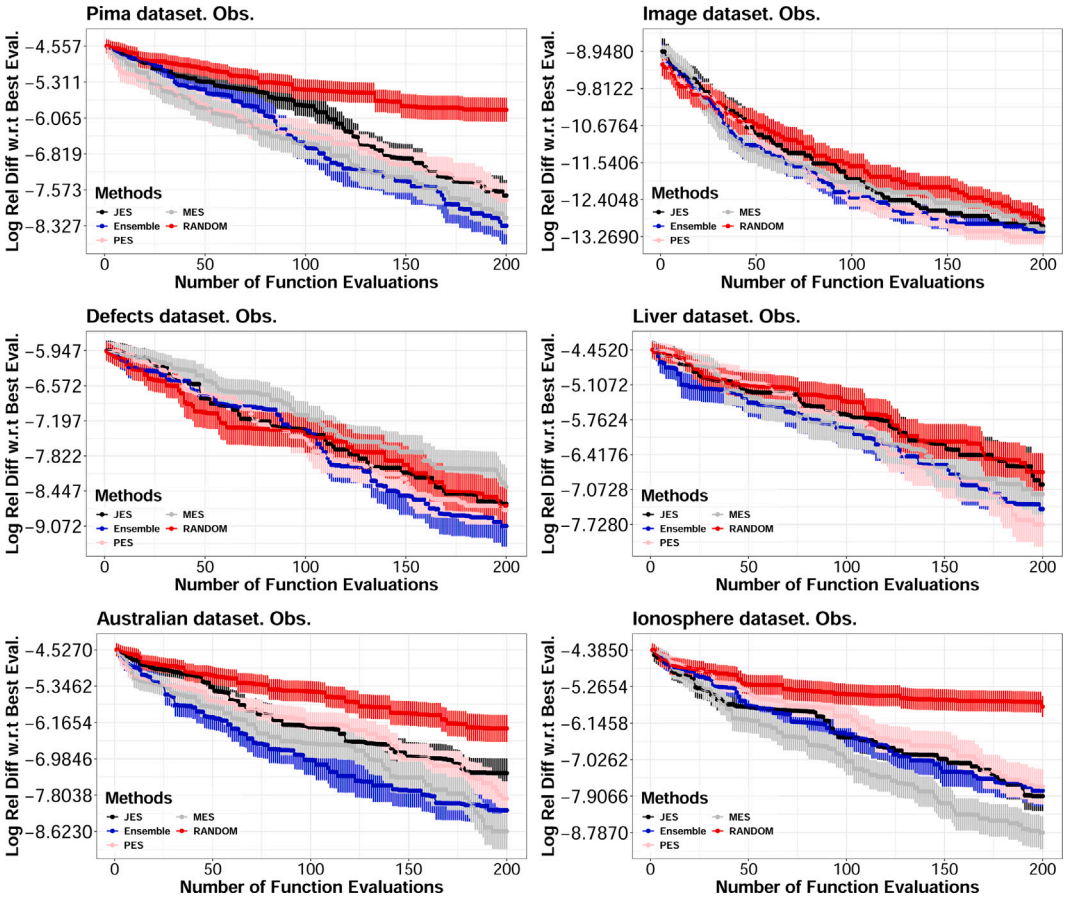

Fig. 9. Average log relative difference between the objective at each method's recommendation and the best objective value observed, with respect to the number of evaluations performed. We present results for optimizing the hyper-parameters of a neural network on the datasets Pima, Liver, Image, Defects, Australian and Ionosphere. Best seen in color.

or moderate, our ensemble approach is the most robust option. By contrast, in settings with high noise or higher dimensionality, JES or MES might be preferable.

The computational cost of the ensemble method is larger than that of the other information-based strategies since it involves 11 different acquisition functions. This may be seen as a disadvantage. However, our results show that the ensemble approach is only 3–6 times more expensive than JES per iteration. The total optimization cost is not 11 times higher because the optimization of the acquisition function is only one step of the BO algorithm. Specifically, the ensemble method only fits the GP model 1 time, not 11 times, and it only generates samples of the optimum $\{\mathbf{x}^\star, y^\star\}$ 1 time, not 11 times. Thus, the cost of sampling $\{\mathbf{x}^\star, y^\star\}$ in JES and in the ensemble method is the same. Moreover, in BO the bottleneck is always the evaluation of the objective function, which is assumed to be significantly more expensive. Therefore, under this assumption, the larger computational cost of the ensemble method with respect to the other strategies is negligible.

JES has a reduced computational cost than the ensemble method. Thus, one may generate more samples S to obtain a better approximation of the acquisition. However, our results indicate that this is not a useful strategy, at least in the noiseless evaluation setting. In particular, the performance of JES may deteriorate with an increased number of samples S. We conjecture that the reason for this is the larger number of local minima in JES with increasing S.

We believe that our method is very general and could be extended to other settings. Specifically, in our work, we have exclusively considered $\{\mathbf{x}^\star, y^\star\}$ as the solution of the optimization problem. However, one may also consider $\mathbf{x}^\star$ or $y^\star$, leading to AES generalizations of PES or MES, respectively.

Finally, we would like to point out that our work allows us to better understand information-based BO methods. Specifically, our analysis shows that they simply measure similarities between probability distributions in terms of a divergence, which is set to be the Kullback–Leibler divergence. This interpretation allows us to change the divergence employed by a more general one, i.e., the α-divergence. Moreover, this may also enable the design of new acquisition functions and new ways to approximate already existing acquisition functions based on information theory. We believe this could be very relevant for the community working in BO and that our work may act as a catalyst for the exploration of new acquisition functions and new approximations to already known acquisition functions.

CRediT authorship contribution statement

Daniel Fernández-Sánchez: Writing – review & editing, Writing – original draft, Software, Methodology, Investigation. **Eduardo C. Garrido-Merchán:** Writing – review & editing, Writing – original draft, Software, Methodology. **Daniel Hernández-Lobato:** Writing – review & editing, Validation, Supervision, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors acknowledge financial support from project PID2022-139856NB-I00 funded by MCIN/AEI/10.13039/501100011033/FEDER, UE, from project IDEA-CM (TEC-2024/COM-89) funded by the Autonomous Community of Madrid, Spain, and from the ELLIS Unit Madrid, Spain. The authors acknowledge computational support from Centro de Computación Científica-Universidad Autónoma de Madrid (CCC-UAM). This publication is also part of the R&D&i project Semi Automatic Meta Analysis helps with Reference PP2024_32, funded by the Universidad Pontificia Comillas, Spain.

Appendix A. KL-divergence and joint entropy search

Here, we show that the acquisition function of Joint Entropy Search (JES) is given by the Kullback–Leibler divergence between the conditional distribution $p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x})$ and the product of the marginal distributions $p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})$ and $p(y|\mathcal{D}_{t-1}, \mathbf{x})$:

$$
\begin{aligned}
&a_{\text{JES}}(\mathbf{x}) = \text{KL}(p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x})||p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1}, \mathbf{x}))\\
&= \int p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x}) \log \frac{p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x})}{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1}, \mathbf{x})} d\{\mathbf{x}^\star, y^\star\} dy\\
&= \int p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x}) \log \frac{p(y|\{\mathbf{x}^\star, y^\star\}, \mathcal{D}_{t-1}, \mathbf{x}) \cancel{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})}}{\cancel{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})} p(y|\mathcal{D}_{t-1}, \mathbf{x})} d\{\mathbf{x}^\star, y^\star\} dy\\
&= \int p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x}) \log p(y|\{\mathbf{x}^\star, y^\star\}, \mathcal{D}_{t-1}, \mathbf{x}) d\{\mathbf{x}^\star, y^\star\} dy\\
&- \int p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x}) \log p(y|\mathcal{D}_{t-1}, \mathbf{x}) d\{\mathbf{x}^\star, y^\star\} dy\\
&= \int p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x}) \log p(y|\{\mathbf{x}^\star, y^\star\}, \mathcal{D}_{t-1}, \mathbf{x}) d\{\mathbf{x}^\star, y^\star\} dy + H\left[p(y|\mathcal{D}_{t-1}, \mathbf{x})\right]\\
&= -\int p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1}) H\left[p(y|\{\mathbf{x}^\star, y^\star\}, \mathcal{D}_{t-1}, \mathbf{x})\right] d\{\mathbf{x}^\star, y^\star\} dy + H\left[p(y|\mathcal{D}_{t-1}, \mathbf{x})\right]\\
&= H\left[p(y|\mathcal{D}_{t-1}, \mathbf{x})\right] - \mathbb{E}_{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})}\left[H\left[p(y|\{\mathbf{x}^\star, y^\star\}, \mathcal{D}_{t-1}, \mathbf{x})\right]\right], \qquad \text{(A.1)}
\end{aligned}
$$

where we have used the product rule of probability and the fact that $p(\{\mathbf{x}^\star, \mathbf{y}^\star\}|\mathcal{D}_{t-1})$ does not depend on $\mathbf{x}$. As in the main document, $H\left[p(y|\mathcal{D}_{t-1}, \mathbf{x})\right]$ is the entropy of the predictive distribution of the GP at $\mathbf{x}$, given the data already observed $\mathcal{D}_{t-1}$, and $H\left[p(y|\{\mathbf{x}^\star, y^\star\}, \mathcal{D}_{t-1}, \mathbf{x})\right]$ is the entropy of the conditional predictive distribution, at $\mathbf{x}$, given the data already observed $\mathcal{D}_{t-1}$ and that the solution of the optimization problem is $\{\mathbf{x}^\star, y^\star\}$.

Appendix B. Derivation of alpha entropy search

In the main document, we propose replacing the KL-divergence in JES with a more general divergence, Amari's α-divergence. This divergence includes the parameter α, which allows us to vary the weight given to discrepancies between distributions across different regions. Specifically, by adjusting α, we can amplify or down-weight differences across various areas of the input space. This substitution results in the following acquisition function:

$$
\begin{aligned}
&a_{\text{AES}}(\mathbf{x}) = D_\alpha(p(y, \{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1}, \mathbf{x})||p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1}, \mathbf{x})p(y|\mathcal{D}_{t-1}, \mathbf{x}))\\
&= \tfrac{1}{(1-\alpha)\alpha}\left(1 - \int \left(p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1}, \mathbf{x})\right)^{1-\alpha} p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x})^\alpha d\{\mathbf{x}^\star, y^\star\} dy\right)\\
&= \tfrac{1}{(1-\alpha)\alpha}\left(1 - \int p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1}, \mathbf{x})\left(\tfrac{p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x})}{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1}, \mathbf{x})}\right)^\alpha d\{\mathbf{x}^\star, y^\star\} dy\right)\\
&= \tfrac{1}{(1-\alpha)\alpha}\left(1 - \mathbb{E}_{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})}\left[\int p(y|\mathcal{D}_{t-1}, \mathbf{x})\left(\tfrac{p(\{\mathbf{x}^\star, y^\star\}, y|\mathcal{D}_{t-1}, \mathbf{x})}{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1}, \mathbf{x})}\right)^\alpha dy\right]\right)\\
&= \tfrac{1}{(1-\alpha)\alpha}\left(1 - \mathbb{E}_{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})}\left[\int p(y|\mathcal{D}_{t-1}, \mathbf{x})\left(\tfrac{p(y|\{\mathbf{x}^\star, y^\star\}, \mathcal{D}_{t-1}, \mathbf{x}) \cancel{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})} p(y|\mathcal{D}_{t-1}, \mathbf{x})}{\cancel{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})} p(y|\mathcal{D}_{t-1}, \mathbf{x})}\right)^\alpha dy\right]\right)\\
&= \tfrac{1}{(1-\alpha)\alpha}\left(1 - \mathbb{E}_{p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})}\left[\int p(y|\mathcal{D}_{t-1}, \mathbf{x})\left(\tfrac{p(y|\mathcal{D}_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})}{p(y|\mathcal{D}_{t-1}, \mathbf{x})}\right)^\alpha dy\right]\right). \qquad \text{(B.1)}
\end{aligned}
$$

As with other information-based BO methods, this expression is analytically intractable and requires approximation. In particular, neither the expectation in B.1 nor the conditional distribution $p(y|\mathcal{D}_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})$ can be computed in closed form. The approximation of the conditional distribution is discussed in Section 3.2 of the main document, and in Appendix C, we detail how to evaluate the integral in B.1. Again, we have used the product rule of probability and that $p(\{\mathbf{x}^\star, y^\star\}|\mathcal{D}_{t-1})$ does not depend on $\mathbf{x}$.

Appendix C. Evaluating the integral in AES with respect to y

In this appendix, we show how to evaluate the integral:

$$
\int p(y|\mathcal{D}_{t-1}, \mathbf{x})\left(\frac{p(y|\mathcal{D}_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})}{p(y|\mathcal{D}_{t-1}, \mathbf{x})}\right)^\alpha dy, \qquad \text{(C.1)}
$$

where $p(y|\mathcal{D}_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})$ is approximated using a truncated Gaussian distribution, as described in the main manuscript. This integral involves a product and a ratio between the predictive distribution conditioned to the problem's solution and the unconditioned distribution to the power of α. Given that both distributions are Gaussian (after the approximations described in the main manuscript) and that the Gaussian belongs to the exponential family of distributions, we can evaluate the integral in closed form using the exponential form of the Gaussian distribution.

First, recall that the density of a Gaussian distribution with mean μ and variance σ^2 can be expressed using a natural parameters representation in terms of natural parameters $\boldsymbol{\eta}$, the vector of sufficient statistics $\mathbf{u}(x)$, and a log-partition function $g(\boldsymbol{\eta})$. Namely,

$$
f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) = \exp\left(\boldsymbol{\eta}^\top \mathbf{u}(x) - g(\boldsymbol{\eta})\right), \qquad \text{(C.2)}
$$

where $\boldsymbol{\eta} = \left(\eta_1 = \frac{\mu}{\sigma^2}, \eta_2 = \frac{1}{\sigma^2}\right)^\top$, sufficient statistics are $\mathbf{u}(x) = \left(x, -0.5x^2\right)^\top$ and the log-partition function is $g(\boldsymbol{\eta}) = \frac{\eta_1^2}{2\eta_2} + \frac{1}{2}\log(2\pi) - \frac{1}{2}\log(\eta_2)$. This notation greatly simplifies the evaluation of the aforementioned integral.

Using the previous representation of the Gaussian distribution, we can now focus on Eq. (C.1), which involves two different Gaussians, where each of them can be converted into its natural parameter representation in the following way:

$$
\begin{aligned}
&p(y|\mathcal{D}_{t-1}, \mathbf{x})\\
&= \exp\left(\boldsymbol{\eta}^\top \mathbf{u}(x) - g(\boldsymbol{\eta})\right), p(y|\mathcal{D}_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\}) = \exp\left(\boldsymbol{\eta}^{\star\top} \mathbf{u}(x) - g(\boldsymbol{\eta}^\star)\right), \qquad \text{(C.3)}
\end{aligned}
$$

where $\boldsymbol{\eta}$ are the natural parameter of the unconditioned predictive distribution of the GP at $\mathbf{x}$, and where $\boldsymbol{\eta}^\star$ are the natural parameters of the approximate conditional predictive distribution at $\mathbf{x}$, given that $\{\mathbf{x}^\star, y^\star\}$ is the solution of the optimization problem. Now, we can compute the ratio inside the integral as:

$$
\begin{aligned}
\left(\frac{p(y|\mathcal{D}_{t-1}, \mathbf{x}, \{\mathbf{x}^\star, y^\star\})}{p(y|\mathcal{D}_{t-1}, \mathbf{x})}\right)^\alpha &= \left(\frac{\exp\left(\boldsymbol{\eta}^{\star\top} \mathbf{u}(y) - g(\boldsymbol{\eta}^\star)\right)}{\exp\left(\boldsymbol{\eta}^\top \mathbf{u}(y) - g(\boldsymbol{\eta})\right)}\right)^\alpha \qquad \text{(C.4)}\\
&= \exp\left(\alpha\left[(\boldsymbol{\eta}^\star - \boldsymbol{\eta})^\top \mathbf{u}(y) - (g(\boldsymbol{\eta}^\star) - g(\boldsymbol{\eta}))\right]\right), \qquad \text{(C.5)}
\end{aligned}
$$

and substitute inside the integral also substituting the first factor $p(y|\mathcal{D}_{t-1}, \mathbf{x})$ by $\exp\left(\boldsymbol{\eta}^\top \mathbf{u}(y) - g(\boldsymbol{\eta})\right)$, obtaining:

$$
\begin{aligned}
&\int \exp\left(\boldsymbol{\eta}^\top \mathbf{u}(y) - g(\boldsymbol{\eta})\right) \exp\left(\alpha\left[(\boldsymbol{\eta}^\star - \boldsymbol{\eta})^\top \mathbf{u}(y) - \left(g(\boldsymbol{\eta}^\star) - g(\boldsymbol{\eta})\right)\right]\right) dy,\\
&= \int \exp\Big(\boldsymbol{\eta}^\top \mathbf{u}(y) - g(\boldsymbol{\eta}) + \alpha(\boldsymbol{\eta}^\star - \boldsymbol{\eta})^\top \mathbf{u}(y) - \alpha\left(g(\boldsymbol{\eta}^\star) - g(\boldsymbol{\eta})\right)\Big) dy\\
&= \int \exp\Big(\boldsymbol{\eta}^\top \mathbf{u}(y) - g(\boldsymbol{\eta}) + \alpha\boldsymbol{\eta}^{\star\top} \mathbf{u}(y) - \alpha\boldsymbol{\eta}^\top \mathbf{u}(y) - \alpha g(\boldsymbol{\eta}^\star) + \alpha g(\boldsymbol{\eta})\Big) dy\\
&= \int \exp\Big(\left(\boldsymbol{\eta}^\top \mathbf{u}(y) - \alpha\boldsymbol{\eta}^\top \mathbf{u}(y)\right) + \alpha\boldsymbol{\eta}^{\star\top} \mathbf{u}(y) + (-g(\boldsymbol{\eta}) + \alpha g(\boldsymbol{\eta})) - \alpha g(\boldsymbol{\eta}^\star)\Big) dy\\
&= \int \exp\Big(\left((1-\alpha)\boldsymbol{\eta}^\top \mathbf{u}(y) + \alpha\boldsymbol{\eta}^{\star\top} \mathbf{u}(y)\right) + (\alpha - 1)g(\boldsymbol{\eta}) - \alpha g(\boldsymbol{\eta}^\star)\Big) dy\\
&= \exp\left((\alpha - 1)g(\boldsymbol{\eta}) - \alpha g(\boldsymbol{\eta}^\star)\right) \int \exp\left(\left((1-\alpha)\boldsymbol{\eta} + \alpha\boldsymbol{\eta}^\star\right)^\top \mathbf{u}(y)\right) dy\\
&= \exp\left((\alpha - 1)g(\boldsymbol{\eta}) - \alpha g(\boldsymbol{\eta}^\star)\right) \exp\left(g\left((1-\alpha)\boldsymbol{\eta} + \alpha\boldsymbol{\eta}^\star\right)\right)
\end{aligned}
$$

$$= \exp\left((\alpha-1)g(\boldsymbol{\eta}) - \alpha g(\boldsymbol{\eta}^\star) + g\left((1-\alpha)\boldsymbol{\eta} + \alpha\boldsymbol{\eta}^\star\right)\right), \tag{C.6}$$

where the last integral is simply given by the exponential of the log-normalizer of a Gaussian, $g(\cdot)$ with natural parameters $(1-\alpha)\boldsymbol{\eta} + \alpha\boldsymbol{\eta}^\star$. Summing up, we have obtained that:

$$\int p(y|\mathcal{D}_{t-1},\mathbf{x})\left(\frac{p(y|\mathcal{D}_{t-1},\mathbf{x},\{\mathbf{x}^\star,y^\star\})}{p(y|\mathcal{D}_{t-1},\mathbf{x})}\right)^\alpha dy = \exp\left\{(\alpha-1)g(\boldsymbol{\eta}) - \alpha g(\boldsymbol{\eta}^\star) + g((1-\alpha)\boldsymbol{\eta} + \alpha\boldsymbol{\eta}^\star)\right\}, \tag{C.7}$$

where $g(\boldsymbol{\eta})$ is the log-normalizer of a Gaussian with natural parameters $\boldsymbol{\eta}$, $\boldsymbol{\eta}$ are the natural parameters of $p(y|\mathcal{D}_{t-1},\mathbf{x})$, and $\boldsymbol{\eta}^\star$ are the natural parameters of the Gaussian approximation of $p(y|\mathcal{D}_{t-1},\mathbf{x},\{\mathbf{x}^\star,y^\star\})$. Specifically,

$$g(\boldsymbol{\eta}) = 0.5\log(2\pi) - 0.5\log\eta_2 + 0.5\frac{\eta_1^2}{\eta_2}, \quad \boldsymbol{\eta} = \left(\frac{m(\mathbf{x})}{v(\mathbf{x})+\sigma^2}, \frac{1}{v(\mathbf{x})+\sigma^2}\right)^\top,$$
$$\boldsymbol{\eta}^\star = \left(\frac{m_{\text{tr}}(\mathbf{x})}{v_{\text{tr}}(\mathbf{x})+\sigma^2}, \frac{1}{v_{\text{tr}}(\mathbf{x})+\sigma^2}\right)^\top, \tag{C.8}$$

where $m(\mathbf{x})$, $v(\mathbf{x})$, $m_{\text{tr}}(\mathbf{x})$ and $v_{\text{tr}}(\mathbf{x})$ are respectively the mean and variances of the unconditional and conditional predictive distribution for $f(\mathbf{x})$, and σ^2 is the variance of the noise.

Appendix D. AES and JES approximations when $\alpha \to 1$

As explained in the main document, even when $S \to \infty$ and $\alpha \to 1$, we have that $\tilde{a}_{\text{AES}}(\mathbf{x}) \nrightarrow \tilde{a}_{\text{JES}}(\mathbf{x})$, although this might not be obvious. In this appendix, we provide the details for this result.

As shown in Appendix A, we can express the JES acquisition as the KL-divergence between $p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})$ and the product of the marginals $p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})$ and $p(y|\mathcal{D}_{t-1},\mathbf{x})$:

$$\begin{aligned}
a_{\text{JES}}(\mathbf{x}) &= \text{KL}(p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})||p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1},\mathbf{x})) \\
&= \int p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})\log\frac{p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})}{p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1},\mathbf{x})}d\{\mathbf{x}^\star,y^\star\}dy \\
&= \int p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})\log\frac{p(y|\{\mathbf{x}^\star,y^\star\},\mathcal{D}_{t-1},\mathbf{x})\cancel{p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})}}{\cancel{p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})}p(y|\mathcal{D}_{t-1},\mathbf{x})}d\{\mathbf{x}^\star,y^\star\}dy \\
&= \int p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})\log p(y|\{\mathbf{x}^\star,y^\star\},\mathcal{D}_{t-1},\mathbf{x})d\{\mathbf{x}^\star,y^\star\}dy \\
&\quad - \int p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})\log p(y|\mathcal{D}_{t-1},\mathbf{x})d\{\mathbf{x}^\star,y^\star\}dy \\
&= H\left[p(y|\mathcal{D}_{t-1},\mathbf{x})\right] - \mathbb{E}_{p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})}\left[H\left[p(y|\{\mathbf{x}^\star,y^\star\},\mathcal{D}_{t-1},\mathbf{x})\right]\right],
\end{aligned} \tag{D.1}$$

where the expectation has to be approximated by Monte Carlo using S samples of $\{\mathbf{x}^\star,y^\star\}$ and the conditional distribution $p(y|\{\mathbf{x}^\star,y^\star\},\mathcal{D}_{t-1},\mathbf{x})$ is approximated using a truncated Gaussian, as described in the main document.

On the other hand, in Appendix B we describe the AES acquisition as Amari's α-divergence between $p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})$ and the product of the marginals $p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})$ and $p(y|\mathcal{D}_{t-1},\mathbf{x})$. That is,

$$\begin{aligned}
&a_{\text{AES}}(\mathbf{x}) \\
&= D_\alpha(p(y,\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1},\mathbf{x}) \parallel p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1},\mathbf{x})p(y|\mathcal{D}_{t-1},\mathbf{x})) \\
&= \frac{1}{(1-\alpha)\alpha}\left(1 - \mathbb{E}_{p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})}\left[\int p(y|\mathcal{D}_{t-1},\mathbf{x})\left(\frac{p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})}{p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1},\mathbf{x})}\right)^\alpha dy\right]\right).
\end{aligned} \tag{D.2}$$

Again, we cannot compute this expression analytically. The expectation is approximated via Monte Carlo using S samples of $\{\mathbf{x}^\star,y^\star\}$, and the conditional distribution $p(y|\{\mathbf{x}^\star,y^\star\},\mathcal{D}_{t-1},\mathbf{x})$ is approximated using a truncated Gaussian, as described in the main document.

If the exact conditional distribution $p(y|\{\mathbf{x}^\star,y^\star\},\mathcal{D}_{t-1},\mathbf{x})$ is used, then AES and JES give the same result when $S \to \infty$ and $\alpha \to 1$. However, consider now the approximation of the conditional distribution $p(y|\{\mathbf{x}^\star,y^\star\},\mathcal{D}_{t-1},\mathbf{x})$ using a truncated Gaussian. Let that approximate distribution be $\tilde{p}(y|\{\mathbf{x}^\star,y^\star\},\mathcal{D}_{t-1},\mathbf{x})$. This step is common in both AES and JES. In the case of AES, the corresponding approximate acquisition is:

$$\begin{aligned}
\tilde{a}_{\text{AES}}(\mathbf{x}) &= \frac{1}{(1-\alpha)\alpha}\left(1 - \mathbb{E}_{p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})}\left[\int p(y|\mathcal{D}_{t-1},\mathbf{x})\left(\frac{\tilde{p}(y|\{\mathbf{x}^\star,y^\star\},\mathcal{D}_{t-1},\mathbf{x})}{p(y|\mathcal{D}_{t-1},\mathbf{x})}\right)^\alpha dy\right]\right) \\
&= \frac{1}{(1-\alpha)\alpha}\left(1 - \mathbb{E}_{p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})}\left[\int p(y|\mathcal{D}_{t-1},\mathbf{x})\left(\frac{\tilde{p}(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})}{p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1},\mathbf{x})}\right)^\alpha dy\right]\right) \\
&= D_\alpha(\tilde{p}(y,\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1},\mathbf{x}) \parallel p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1},\mathbf{x})p(y|\mathcal{D}_{t-1},\mathbf{x})),
\end{aligned} \tag{D.3}$$

where

$$\tilde{p}(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x}) = \tilde{p}(y|\{\mathbf{x}^\star,y^\star\},\mathcal{D}_{t-1},\mathbf{x})p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1}), \tag{D.4}$$

is an approximate joint distribution. Thus, when $\alpha \to 1$ we have that:

$$\tilde{a}_{\text{AES}}(\mathbf{x}) \to \text{KL}(\tilde{p}(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x}) \parallel p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1},\mathbf{x})). \tag{D.5}$$

By contrast, in the case of JES, the truncated Gaussian approximation gives the approximate acquisition:

$$\begin{aligned}
\tilde{a}_{\text{JES}}(\mathbf{x}) &= H\left[p(y|\mathcal{D}_{t-1},\mathbf{x})\right] - \mathbb{E}_{p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})}\left[H\left[\tilde{p}(y|\{\mathbf{x}^\star,y^\star\},\mathcal{D}_{t-1},\mathbf{x})\right]\right] \\
&= -\int p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})\log p(y|\mathcal{D}_{t-1},\mathbf{x})d\{\mathbf{x}^\star,y^\star\}dy \\
&\quad + \int p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})\log\tilde{p}(y|\{\mathbf{x}^\star,y^\star\},\mathcal{D}_{t-1},\mathbf{x})d\{\mathbf{x}^\star,y^\star\}dy \\
&= \int p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})\log\frac{\tilde{p}(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})}{p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1},\mathbf{x})}d\{\mathbf{x}^\star,y^\star\}dy \\
&\neq \text{KL}(\tilde{p}(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x}) \parallel p(\{\mathbf{x}^\star,y^\star\}|\mathcal{D}_{t-1})p(y|\mathcal{D}_{t-1},\mathbf{x})),
\end{aligned} \tag{D.6}$$

since the approximate joint distribution $\tilde{p}(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})$ appears only inside the log function. Outside the log function appears the exact joint distribution $p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})$. Note that in (D.6) we have also used (D.4). Therefore, JES uses the exact joint distribution $p(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})$ in one factor, but the approximate joint distribution $\tilde{p}(\{\mathbf{x}^\star,y^\star\},y|\mathcal{D}_{t-1},\mathbf{x})$ in the other. This explains why the final approximated expressions for JES and AES are different when $\alpha \to 1$. Summing up, JES and AES need not give the same results when $\alpha \to 1$.

Appendix E. Comparison of AES, JES, and the ensemble method in a noisy evaluation setting

In the main document, we compared JES, AES for different α values, and the ensemble method on a 1D noiseless problem. We observed that averaging over α, the ensemble acquisition function is less rugged than when using α values close to 1. In a noisy evaluation scenario, however, the number of peaks, *i.e.*, local maxima of JES and AES, that appear at the sampled locations of $\{\mathbf{x}^\star,y^\star\}$ is lower than in a noiseless setting. This reduction occurs because the probabilistic model accounts for the presence of noise, so the variance of the conditional model does not decrease to zero at the sampled locations. This reduction in the number of peaks is shown in Table E.5 where we display the average number of local maxima of JES and the ensemble method across 100 repetitions on a 1D synthetic experiment under evaluation noise. In this setting, there is no statistically significant difference between the methods with respect to the number of local maxima in the acquisition. Furthermore, Fig. E.10 shows a plot of the different acquisition functions, for a particular repetition of the experiment, where this reduction of the local maxima can be visually observed. In that figure, all acquisition functions indicate that the input around −4.8 is expected to have the highest utility. Although it may appear that all the maxima are at the same input, they are actually very close but different. Specifically, they are in the range −4.85 to −4.74. Without the adverse effect of local maxima, the approach of averaging over different α values is expected to be less advantageous in a noisy setting.

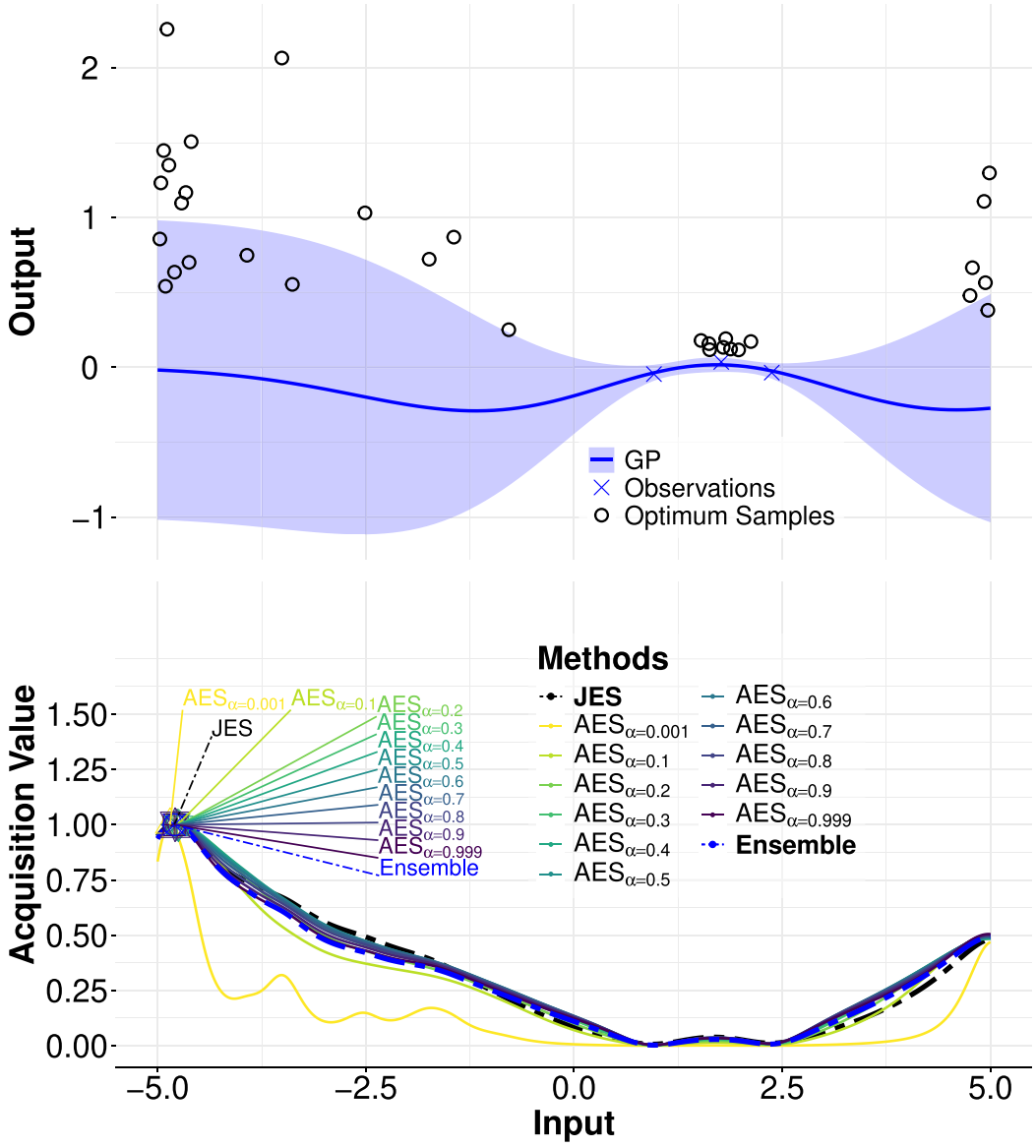

Fig. E.10. (bottom) Comparison of AES for different α values, JES and the ensemble acquisition function in a one-dimensional noisy synthetic problem. We also display the maximum of each acquisition function. (top) Predictive distribution of the GP and generated samples of $\{\mathbf{x}^\star, y^\star\}$. The acquisition functions have been normalized so that the maximum is equal to one for better visualization. Best viewed in color.

Table E.5
Average number of local maxima for each method over 100 repetitions in a 1D noisy synthetic problem.

Method	# of local maxima
JES	4.22±0.073
Ensemble	4.21±0.071

Appendix F. Impact of scaling with local maxima in the ensemble method

In this section, we analyze the impact of scaling each individual acquisition function with a value that need not be the global maximizer of each acquisition in the ensemble method described in Eq. (16). This is a more realistic scenario and the one actually used in our experiments in Section 5. Specifically, finding the global maximizer of each acquisition is generally infeasible and typically any optimization algorithm is expected to find only a local maximum.

With the aforementioned goal, we compare the ensemble method using the exact normalization values given by the global maximum of each individual acquisition and the ensemble method using for normalization a local maximum of each acquisition obtained by gradient ascent starting at a random location of the input domain. Thus, this value need not be equal to the global maximum. We refer to the first method as $\text{Ensemble}_{\text{global}}$. The second method is referred to as $\text{Ensemble}_{\text{local}}$ since it uses a local maximum for the normalization of each individual acquisition function. We remark that by starting from a random location and executing gradient ascent, we are actually following the same procedure as the one used in the experiments reported in Section 5 for the ensemble method.

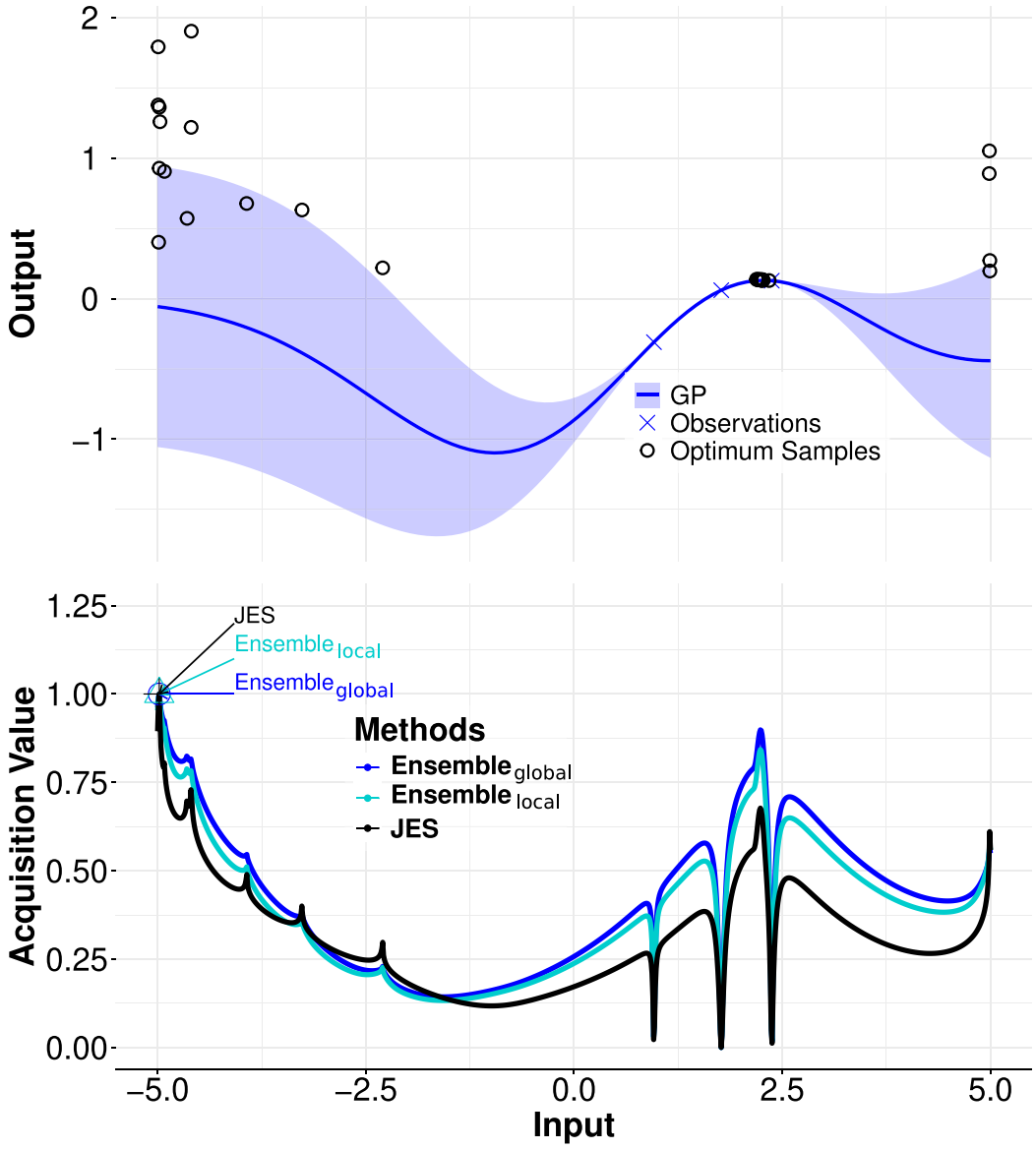

Fig. F.11. (bottom) Comparison of JES, the ensemble method using the global maxima for the weighted scaling ($\text{Ensemble}_{\text{global}}$), and the ensemble method using local maxima for the weighted scaling ($\text{Ensemble}_{\text{local}}$). We also display the maximum of each acquisition function. (top) Predictive distribution of the GP and generated samples of $\{\mathbf{x}^\star, y^\star\}$. The acquisition functions have been normalized so that the maximum is equal to one for better visualization. Best viewed in color.

Fig. F.11 shows a comparison of the two methods described for a representative scenario. We also report the JES acquisition for reference and show at the top the predictive distribution of the GP and the generated samples of $\{\mathbf{x}^\star, y^\star\}$. We have considered the same predictive distribution and samples as those displayed in Fig. 3 in the main document. Moreover, as in Fig. 3, to facilitate the comparison, the acquisition of each ensemble method and the acquisition of JES are scaled by the actual global maximum so that the global maximum is equal to 1. Note that this does not affect the location of the global maximum of the acquisition and facilitates the comparison. We can observe, in this example, that $\text{Ensemble}_{\text{local}}$ and $\text{Ensemble}_{\text{global}}$ generate very similar acquisition functions. Specifically, their local maxima are nearly identical and the global maximum coincides (at -4.85). The small differences between the two acquisitions are simply a consequence of small changes in the normalization constants of each individual acquisition. While $\text{Ensemble}_{\text{global}}$ uses the global maxima, $\text{Ensemble}_{\text{local}}$ uses local maxima found by gradient ascent. This results in slight variations in the normalization constant of each individual acquisition function and therefore, in the corresponding weights used in Eq. (16). However, these differences lead only to small changes in the combined acquisition as shown by Fig. F.11. Table F.6 shows the global maxima of each acquisition function, for each value of α, used in the method $\text{Ensemble}_{\text{global}}$. The local maxima of each acquisition function, for each value of α, used in the method $\text{Ensemble}_{\text{local}}$ are also shown for comparison. We observe that the local maxima are always smaller than the global maxima, as expected, but the orders of magnitude remain similar. Table F.6 also shows the corresponding normalized weight associated with each acquisition function, for each value of α. The sum of the normalized weights equals 1 and is proportional to $1/w_\alpha$. We observe only small differences in the normalized weights of $\text{Ensemble}_{\text{global}}$ and $\text{Ensemble}_{\text{local}}$, which explains the similarities in the final aggregated acquisition function displayed in Fig. F.11.

Table F.6
Global and local maxima of each individual acquisition function for each α value and associated normalized weights, whose sum is equal to 1, and proportional to $1/w_\alpha$.

α-value	w_α values for each method		Normalized weights for each method	
	$\text{Ensemble}_{\text{global}}$	$\text{Ensemble}_{\text{local}}$	$\text{Ensemble}_{\text{global}}$	$\text{Ensemble}_{\text{local}}$
0.001	211.7046	78.82188	0.0008	0.0011
0.1	4.267942	2.631097	0.0406	0.0350
0.2	2.488881	2.059664	0.0697	0.0448
0.3	1.876402	1.871234	0.0924	0.0493
0.4	1.579288	1.409696	0.1098	0.0654
0.5	1.416667	1.178934	0.1224	0.0783
0.6	1.376707	1.091412	0.1260	0.0845
0.7	1.397364	0.345451	0.1241	0.2672
0.8	1.480361	0.816017	0.1172	0.1131
0.9	1.641707	0.911297	0.1057	0.1013
0.999	1.911981	0.579223	0.0907	0.1594

Table F.7
Average number of local maxima for each method over 100 repetitions.

Method	# of local maxima
JES	17.60±0.56
$\text{Ensemble}_{\text{global}}$	12.53±0.49
$\text{Ensemble}_{\text{local}}$	11.65±0.37

Next, we investigate the impact on the number of local optima in the final acquisition when using the exact or the approximate normalization. For this, we repeat the experiments whose results are displayed in Table 1. Here, we report the performance of $\text{Ensemble}_{\text{local}}$ too.

Table F.7 shows the average number of local maxima, along with the standard error, for the two acquisition functions. We observe that $\text{Ensemble}_{\text{local}}$ has a slightly lower number of local maxima than $\text{Ensemble}_{\text{global}}$, although the differences are not statistically significant. Importantly, the number of local optima in the acquisition of $\text{Ensemble}_{\text{local}}$ is much smaller than that of JES. Therefore, the beneficial properties of the ensemble method, regarding the reduction of local optima in the final acquisition function, are also observed when using an approximate normalization based on local optima.

Summing up, the results of the experiments carried out in this section indicate that the impact of using local maxima to compute the corresponding weights of each individual acquisition function in Eq. (16) is small. Local maxima only tend to slightly underestimate the actual global maxima, as shown in Table F.6. This introduces a small amount of noise in the actual weights of each acquisition function considered in Eq. (16). Notwithstanding, the final acquisition remains very similar, as shown in Fig. F.11. Furthermore, when using local maxima for normalization, the aggregated acquisition function of the ensemble method also has a significantly smaller number of local maxima than the acquisition function of JES, as shown in Table F.7.

Appendix G. Quality of the approximation of the acquisition function with a small number of samples

In this section, we investigate the accuracy of the AES acquisition function approximation proposed in (15) when using a number of samples $S = 32$. We consider the same 1-dimensional toy problem as in Section 5.1, but here the conditional density $p(y|D_{t-1}, \mathbf{x}, \{y_s^\star, \mathbf{x}_s^\star\})$ is approximated using only $S = 32$ samples of $\{\mathbf{x}^\star, y^\star\}$. The exact acquisition function targeted by AES is estimated using a more accurate Monte Carlo method with 6000 samples. For a fair comparison, both methods use identical values for α. As in Section 5.1, for each sample $\{\mathbf{x}^\star, y^\star\}$, the conditional density is estimated via a kernel density estimator on samples drawn from the GP posterior at $\mathbf{x}$ that are compatible with $\{y_s^\star, \mathbf{x}_s^\star\}$, and the one-dimensional integral in (10) is computed using quadrature. For simplicity, a noiseless evaluation setting is assumed.

Fig. G.12 (top-left) displays the GP predictive distribution for the objective and the observed data. From (top-right) to (bottom-left), we compare, for a representative set of α values, the proposed AES approximation with the corresponding exact acquisition function estimate. Overall, both methods have a similar shape, with local maxima and minima at similar locations. However, in contrast with the behavior observed using $S = 6000$ samples (see Section 5.1), now the shape of the approximation is a bit shifted, due to the randomness of the samples used, and it has undesirable spikes. As before, the AES methods tend to underestimate their corresponding exact estimators. We also observe that the differences are more pronounced for α values close to zero. For intermediate values of α, the number of spikes decreases, and varying α continues to impact the shape of the acquisition function in specific regions of the input space (e.g., when $x > 4$). Finally, Fig. G.12 (bottom-right) shows that the ensemble acquisition function computed with the proposed approximation is nearly identical to its exact counterpart.

Appendix H. Impact of number of restarts and candidate points

As described in the main document, to maximize each acquisition function we use L-BFGS-B with 1 restart and 200 candidate points, from which the starting point of the optimization is chosen. In this setting, BOTorch chooses randomly one point to start the optimization, from a random set of 200 points, favoring the selection of points with high acquisition [20]. Here, we consider a different number of restarts and points to choose the starting point of the optimization process. Specifically, we consider increasing the number of restarts to 5 while keeping the number of points equal to 200. In this setting, the acquisition function is optimized 5 times, from different starting points, chosen from the initial set of 200 points. BOTorch favors the selection of 5 different points with high acquisition from the initial set of 200 points. After optimization, the final point with the best acquisition is selected as the next evaluation point. Here, we use 25 points initial points chosen at random. The results obtained, in this setting, for the 4-dimensional synthetic problem, are displayed in Fig. H.13, for the noiseless and the noisy evaluation scenario. We observe that the results obtained are very similar to those reported in the main manuscript. Here, we also consider increasing the number of points to 500 while keeping the number of restarts equal to 1. In this setting, the acquisition function is optimized 1 time, and the starting point is chosen from an initial set of 500 random points. The results obtained, in this setting, for the 4-dimensional synthetic problem, are displayed in Fig. H.14, for the noiseless and the noisy evaluation scenario. Again, we observe that the results obtained are very similar to those reported in the main manuscript.

Appendix I. Extra experiments execution time

This section gives extra results about the average time required by each method per iteration. Specifically, Table I.8 shows the results for the noisy synthetic experiments. Table I.9 shows the results for JES and the ensemble method in the synthetic experiments as a function of the number of samples S of $\{\mathbf{x}^\star, y^\star\}$ considered. Finally, Table I.10 shows the results for the noisy benchmark experiments. The reported results are very similar to those found in the experiments section. Best results are highlighted in boldface.

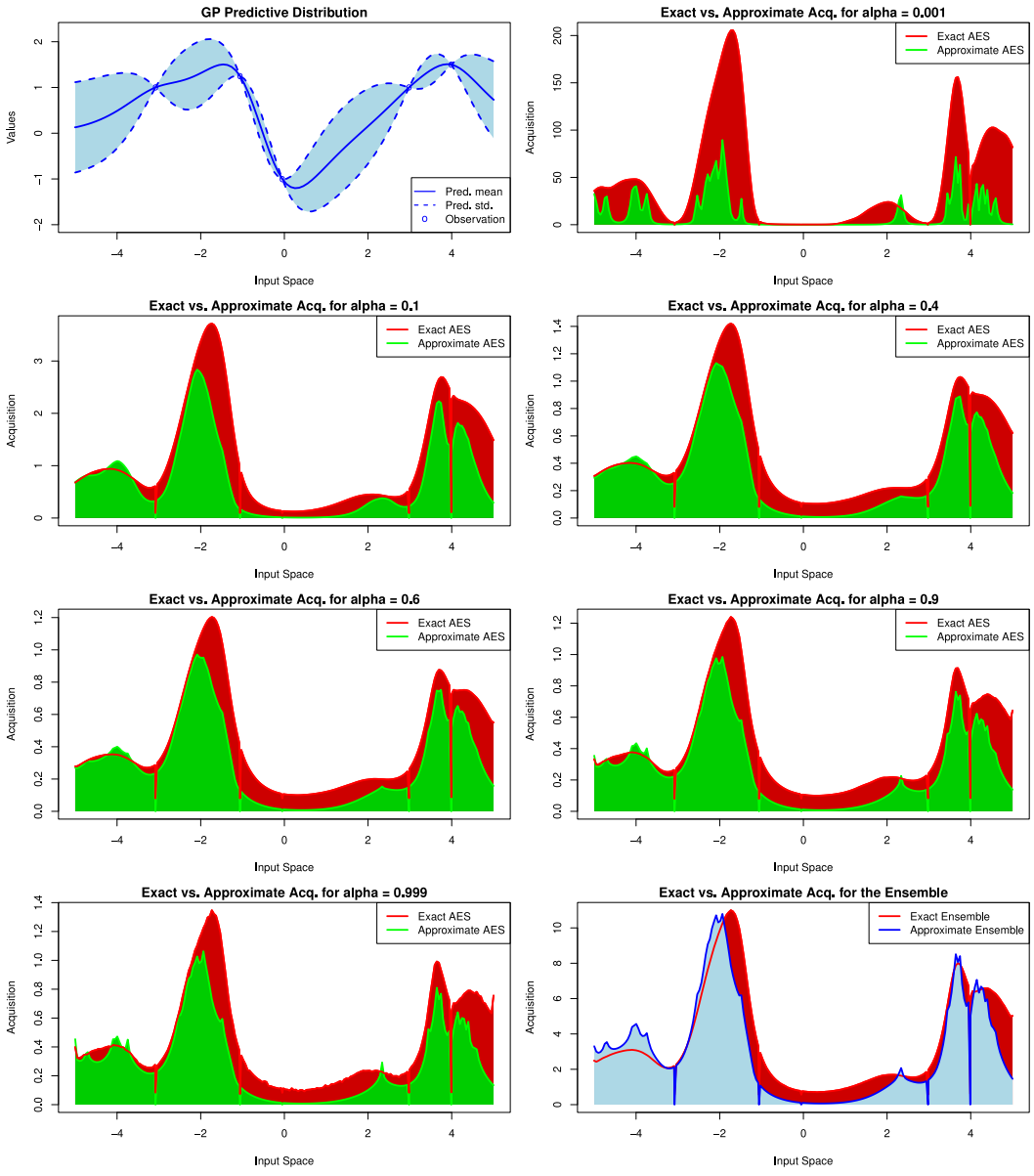

Fig. G.12. (top-left) GP predictive distribution for the objective. From (top-right) to (bottom-left) we show the AES acquisition function computed with the proposed approximation and its corresponding exact estimate, both obtained using $S = 32$ samples of $\{\mathbf{x}^\star, y^\star\}$. We report results for a representative set of α values. (bottom-right) Comparison of the ensemble methods using the proposed approximation and the exact estimate. Best viewed in color.

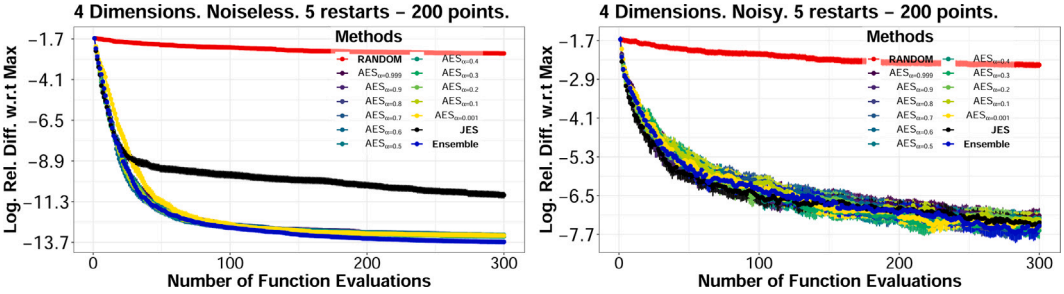

Fig. H.13. Average logarithm relative difference between the objective at each method's recommendation and the objective at the global maximum, with respect to the number of evaluations. Results are shown for the 4-dimensional problem when the number of restarts is increased to 5. Best viewed in color.

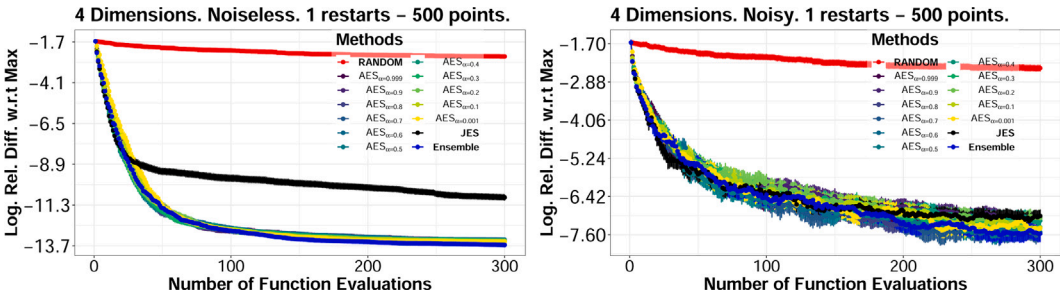

Fig. H.14. Average logarithm relative difference between the objective at each method's recommendation and the objective at the global maximum, with respect to the number of evaluations. Results are shown for the 4-dimensional problem when the number of points to choose the starting point of the optimization is increased to 500. Best viewed in color.

Table I.8
Average execution time and standard error per iteration (in s) in the noisy synthetic experiments.

	4D	6D	8D	12D
EI	**3,697±0,019**	**4,411±0,194**	**6,524±0,190**	**7,314±0,204**
JES	7,797±0,309	10,644±0,447	14,631±0,795	18,166±0,685
$AES_{\alpha=0.001}$	7,580±0,346	10,251±0,369	12,902±0,498	16,848±0,714
$AES_{\alpha=0.1}$	7,472±0,332	10,609±0,461	13,787±0,328	16,723±0,682
$AES_{\alpha=0.2}$	7,184±0,266	10,327±0,511	13,641±0,492	16,859±0,513
$AES_{\alpha=0.3}$	7,523±0,270	10,679±0,485	13,137±0,457	16,950±0,571
$AES_{\alpha=0.5}$	7,332±0,339	11,009±0,470	13,850±0,466	17,846±0,637
$AES_{\alpha=0.5}$	7,201±0,269	11,257±0,433	14,306±0,429	18,064±0,605
$AES_{\alpha=0.6}$	7,251±0,269	10,863±0,498	14,386±0,540	17,316±0,692
$AES_{\alpha=0.7}$	7,855±0,262	10,418±0,445	14,793±0,494	16,529±0,577
$AES_{\alpha=0.8}$	7,416±0,255	10,306±0,474	14,373±0,364	17,404±0,753
$AES_{\alpha=0.9}$	7,455±0,271	10,469±0,557	13,789±0,501	16,211±0,718
$AES_{\alpha=0.999}$	7,314±0,245	10,117±0,509	13,797±0,482	17,172±0,650
Ensemble	31,968±1,164	37,580±1,602	47,170±1,275	52,989±2,055

Table I.9
Average execution time and standard error per iteration (in s) in the synthetic experiments, as a function of the number of samples employed.

	4D noiseless	4D noisy	8D noiseless	8D noisy
JES_1	**3,372±0,111**	3,376±0,136	5,729±0,202	6,444±0,231
JES_2	3,631±0,152	**3,277±0,123**	**5,285±0,254**	**6,083±0,334**
JES_4	3,873±0,126	3,614±0,126	6,273±0,188	6,434±0,296
JES_8	4,147±0,160	4,160±0,170	6,642±0,239	7,138±0,330
JES_{16}	5,483±0,188	4,913±0,201	8,021±0,327	9,255±0,373
JES_{32}	7,898±0,419	7,797±0,309	12,741±0,506	14,631±0,795
JES_{64}	12,717±0,632	11,343±0,523	17,719±1,078	16,619±0,868
$Ensemble_1$	5,384±0,155	5,356±0,219	8,048±0,403	8,621±0,360
$Ensemble_2$	6,569±0,196	6,693±0,221	9,219±0,464	11,002±0,384
$Ensemble_4$	8,539±0,301	8,430±0,323	11,068±0,433	11,841±0,493
$Ensemble_8$	12,081±0,408	11,437±0,472	15,276±0,641	14,756±0,683
$Ensemble_{16}$	19,677±0,738	18,310±0,674	24,766±1,160	26,403±1,144
$Ensemble_{32}$	30,097±1,233	31,968±1,164	42,338±1,266	47,170±1,275
$Ensemble_{64}$	63,710±2,323	61,623±2,410	72,065±2,614	75,727±3,092

Table I.10
Average execution time and standard error per iteration (in s) in the noisy benchmark experiments.

	Hartman3D	Styblinskitang4D	Hartman6D	Cosine8D
EI	**2,714±0,079**	**3,289±0,099**	**3,998±0,159**	**5,525±0,276**
PES	5,524±0,225	8,030±0,520	10,092±0,353	9,304±0,409
MES	4,523±0,195	5,604±0,152	8,233±0,339	9,393±0,262
JES	6,083±0,264	7,279±0,189	10,585±0,375	10,040±0,319
Ensemble	39,886±1,290	31,018±0,806	37,428±1,263	51,396±0,866

Data availability

Code available at: github.com/fernandezdaniel/alphaES.

References

[1] J. Snoek, H. Larochelle, R.P. Adams, Practical Bayesian optimization of machine learning algorithms, in: Advances in Neural Information Processing Systems, vol. 25, 2012.

[2] E. Brochu, V.M. Cora, N. De Freitas, A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning, Technical Report TR-2009-023, University of British Columbia, 2009.

[3] B. Shahriari, K. Swersky, Z. Wang, R. Adams, N. De Freitas, Taking the human out of the loop: A review of Bayesian optimization, Proceedings of the IEEE 104 (2015) 148–175.

[4] R. Garnett, Bayesian Optimization, Cambridge University Press, 2023.

[5] C.E. Rasmussen, C.K. Williams, Gaussian Processes for Machine Learning, MIT Press Cambridge, MA, 2006.

[6] L. Cornejo-Bueno, E.C. Garrido-Merchán, D. Hernández-Lobato, S. Salcedo-Sanz, Bayesian optimization of a hybrid system for robust ocean wave features prediction, Neurocomputing 275 (2018) 818–828.

[7] G. Agarwal, H.A. Doan, L.A. Robertson, L. Zhang, R.S. Assary, Discovery of energy storage molecular materials using quantum chemistry-guided multiobjective Bayesian optimization, Chem. Mater. 33 (20) (2021) 8133–8144.

[8] R. Martinez-Cantin, Bayesian optimization with adaptive kernels for robot control, in: 2017 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2017, pp. 3350–3356.

[9] E.C. Garrido-Merchán, G.G. Piris, M.C. Vaca, Bayesian optimization of ESG (environmental social governance) financial investments, Environ. Res. Commun. 5 (5) (2023) 055003.

[10] B. Solnik, D. Golovin, G. Kochanski, J.E. Karro, S. Moitra, D. Sculley, Bayesian optimization for a better dessert, in: Proceedings of the 2017 NIPS Workshop on Bayesian Optimization, 2017.

[11] J.M. Hernández-Lobato, M.A. Gelbart, B. Reagen, R. Adolf, D. Hernández-Lobato, P. Whatmough, D. Brooks, G.-Y. Wei, R.P. Adams, Designing neural network hardware accelerators with decoupled objective evaluations, in: NIPS Workshop on Bayesian Optimization, 2016.

[12] J.M. Hernández-Lobato, M.W. Hoffman, Z. Ghahramani, Predictive entropy search for efficient global optimization of black-box functions, in: Advances in Neural Information Processing Systems, vol. 27, 2014.

[13] J. Villemonteix, E. Vazquez, E. Walter, An informational approach to the global optimization of expensive-to-evaluate functions, J. Global Optim. 44 (2009) 509.

[14] P. Hennig, C.J. Schuler, Entropy search for information-efficient global optimization, J. Mach. Learn. Res. 13 (6) (2012).

[15] C. Hvarfner, F. Hutter, L. Nardi, Joint entropy search for maximally-informed Bayesian optimization, in: Advances in Neural Information Processing Systems, vol. 35, 2022, pp. 11494–11506.

[16] B. Tu, A. Gandy, N. Kantas, B. Shafei, Joint entropy search for multi-objective Bayesian optimization, in: Advances in Neural Information Processing Systems, 2022, pp. 9922–9938.

[17] S.-i. Amari, Differential-Geometrical Methods in Statistics, vol. 28, Springer-Verlag, 1985.

[18] T. Minka, et al., Divergence Measures and Message Passing, Technical Report, Microsoft Research, 2005.

[19] T.G. Dietterich, Ensemble methods in machine learning, in: International Workshop on Multiple Classifier Systems, 2000, pp. 1–15.

[20] M. Balandat, B. Karrer, D. Jiang, S. Daulton, B. Letham, A.G. Wilson, E. Bakshy, BoTorch: A framework for efficient Monte-Carlo Bayesian optimization, in: Advances in Neural Information Processing Systems, vol. 33, 2020, pp. 21524–21538.

[21] Z. Wang, S. Jegelka, Max-value entropy search for efficient Bayesian optimization, in: International Conference on Machine Learning, PMLR, 2017, pp. 3627–3635.

[22] A. Cichocki, S.-C. Amari, Families of alpha-beta-and gamma-divergences: flexible and robust measures of similarities, Entropy 12 (2010) 1532–1568.

[23] J.M. Hernandez-Lobato, Y. Li, M. Rowland, D. Hernández-Lobato, T. Bui, R.E. Turner, Black-box alpha divergence minimization, in: International Conference on Machine Learning, 2016, pp. 1511–1520.
[24] S. Rodríguez-Santana, D. Hernández-Lobato, Adversarial α-divergence minimization for Bayesian approximate inference, Neurocomputing 471 (2022) 260–274.
[25] Q.P. Nguyen, B.K.H. Low, P. Jaillet, Rectified max-value entropy search for Bayesian optimization, 2022, arXiv preprint arXiv:2202.13597.
[26] A. Rahimi, B. Recht, et al., Random features for large-scale kernel machines, in: NIPS, vol. 3, Citeseer, 2007, pp. 1177–1184.
[27] C. Villacampa-Calvo, D. Hernández-Lobato, Alpha divergence minimization in multi-class Gaussian process classification, Neurocomputing 378 (2020) 210–227.
[28] C. Villacampa-Calvo, G. Hernández-Munoz, D. Hernández-Lobato, Alpha-divergence minimization for deep gaussian processes, Int. J. Approx. Reason. 150 (2022) 139–171.
[29] M. Hoffman, E. Brochu, N. De Freitas, Portfolio allocation for Bayesian optimization, in: Uncertainty in Artificial Intelligence, 2011, pp. 327–336.
[30] B. Shahriari, Z. Wang, M.W. Hoffman, A. Bouchard-Côté, N. de Freitas, An entropy search portfolio for Bayesian optimization, 2014, arXiv preprint arXiv:1406.4625.
[31] M.W. Hoffman, Z. Ghahramani, Output-space predictive entropy search for flexible global optimization, in: NIPS Workshop on Bayesian Optimization, 2015, pp. 1–5.
[32] B. Ru, M.A. Osborne, M. McLeod, D. Granziol, Fast information-theoretic Bayesian optimisation, in: International Conference on Machine Learning, PMLR, 2018, pp. 4384–4392.
[33] Q.P. Nguyen, Z. Wu, B.K.H. Low, P. Jaillet, Trusted-maximizers entropy search for efficient Bayesian optimization, in: Uncertainty in Artificial Intelligence, PMLR, 2021, pp. 1486–1495.
[34] W. Neiswanger, L. Yu, S. Zhao, C. Meng, S. Ermon, Generalizing Bayesian optimization with decision-theoretic entropies, in: Advances in Neural Information Processing Systems, vol. 35, 2022, pp. 21016–21029.
[35] Y. Li, Y. Gal, Dropout inference in Bayesian neural networks with alpha-divergences, in: International Conference on Machine Learning, 2017, pp. 2052–2061.
[36] T.D. Bui, J. Yan, R.E. Turner, A unifying framework for Gaussian process pseudo-point approximations using power expectation propagation, J. Mach. Learn. Res. 18 (2017) 1–72.
[37] T.D. Bui, D. Hernández-Lobato, J.M. Hernández-Lobato, Y. Li, R.E. Turner, NIPS workshop on advances in approximate Bayesian inference, 2016.
[38] L.I. Midgley, V. Stimper, G.N.C. Simm, B. Schölkopf, J.M. Hernández-Lobato, Flow annealed importance sampling bootstrap, in: International Conference on Learning Representations, 2023.
[39] J. Mockus, V. Tiesis, A. Zilinskas, The application of Bayesian methods for seeking the extremum, in: Towards Global Optimization, vol. 2, 1978, pp. 117–129.
[40] J.M. Hernández-Lobato, M. Gelbart, M. Hoffman, R. Adams, Z. Ghahramani, Predictive entropy search for Bayesian optimization with unknown constraints, in: International Conference on Machine Learning, 2015, pp. 1699–1707.
[41] D. Hernández-Lobato, J.M. Hernández-Lobato, A. Shah, R. Adams, Predictive entropy search for multi-objective Bayesian optimization, in: International Conference on Machine Learning, PMLR, 2016, pp. 1492–1501.
[42] M. Jamil, X.-S. Yang, A literature survey of benchmark functions for global optimisation problems, Int. J. Math. Model. Numer. Optim. 4 (2) (2013) 150–194.
[43] X.-S. Yang, Engineering Optimization: An Introduction with Metaheuristic Applications, John Wiley & Sons, 2010.
[44] K.N. M. Kelly, The UCI Machine Learning Repository, https://archive.ics.uci.edu.