BORE: Bayesian Optimization by Density-Ratio Estimation

Louis C. Tiao [1 2] **Aaron Klein** [3] **Matthias Seeger** [3] **Edwin V. Bonilla** [2 1] **Cédric Archambeau** [3] **Fabio Ramos** [1 4]

Abstract

Bayesian optimization (BO) is among the most effective and widely-used blackbox optimization methods. BO proposes solutions according to an explore-exploit trade-off criterion encoded in an acquisition function, many of which are computed from the posterior predictive of a probabilistic surrogate model. Prevalent among these is the expected improvement (EI). The need to ensure analytical tractability of the predictive often poses limitations that can hinder the efficiency and applicability of BO. In this paper, we cast the computation of EI as a binary classification problem, building on the link between class-probability estimation and density-ratio estimation, and the lesser-known link between density-ratios and EI. By circumventing the tractability constraints, this reformulation provides numerous advantages, not least in terms of expressiveness, versatility, and scalability.

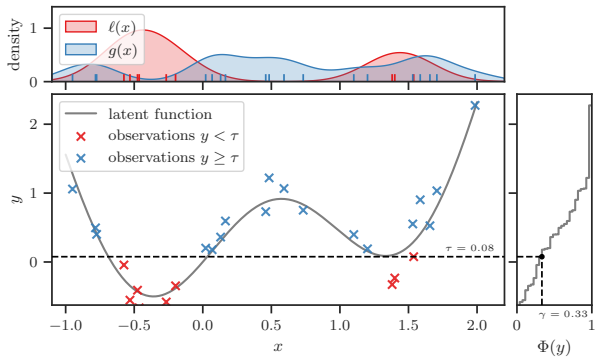


Figure 1. Optimizing a synthetic function $f(x) = \sin(3x) + x^2 - 0.7x$ with observation noise $\varepsilon \sim \mathcal{N}(0, 0.2^2)$. In the main pane, the noise-free function is represented by the solid gray curve, and $N = 27$ noisy observations are represented by the crosses '$\times$'. Observations with output y in the top-performing $\gamma = 1/3$ proportion are shown in *red*; otherwise, they are shown in *blue*. Their corresponding densities, $\ell(x)$ and $g(x)$, respectively, are shown in the top pane. BORE exploits the correspondence between the EI acquisition function and the *ratio* of densities $\ell(x)/g(x)$.

1. Introduction

Bayesian optimization (BO) is a sample-efficient methodology for the optimization of expensive blackbox functions (Brochu et al., 2010; Shahriari et al., 2015). In brief, BO proposes candidate solutions according to an *acquisition function* that encodes the explore-exploit trade-off. At the core of BO is a probabilistic surrogate model based on which the acquisition function can be computed.

Of the many acquisition functions that have been devised, the expected improvement (EI) (Mockus et al., 1978; Jones et al., 1998) remains predominant, due in large to its effectiveness in spite of its relative simplicity. In particular, while acquisition functions are generally difficult to compute, let alone optimize (Wilson et al., 2018), EI has a closed-form expression when the model's posterior predictive is Gaussian. However, while this condition makes EI easier to work with, it can also preclude the use of richer families of models: one must ensure analytical tractability of the predictive, often at the expense of expressiveness, or otherwise resort to sampling-based approximations (Balandat et al., 2020).

By virtue of its flexibility, well-calibrated predictive uncertainty, and conjugacy properties, Gaussian process (GP) regression (Williams & Rasmussen, 1996) is a widely-used probabilistic model in BO. To extend GP-based BO to problems with discrete variables (Garrido-Merchán & Hernández-Lobato, 2020), structures with conditional dependencies (Jenatton et al., 2017), or to capture nonstationary phenomenon (Snoek et al., 2014), it is common to apply simple modifications to the covariance function, as this can often be done without compromising the tractability of the predictive. Suffice it to say, there exist estimators more naturally adept at dealing with these conditions (e.g. decision trees in the case of discrete variables). Indeed, to scale BO to problem settings that produce vast numbers of observations,

[1]University of Sydney, Sydney, Australia [2]CSIRO's Data61, Sydney, Australia [3]Amazon, Berlin, Germany [4]NVIDIA, Seattle, WA, USA. Correspondence to: Louis Tiao <louis.tiao@sydney.edu.au>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

such as in transfer learning (Swersky et al., 2013), existing works have resorted to different families of models, such as random forests (RFs) (Hutter et al., 2011) and Bayesian neural networks (BNNs) (Snoek et al., 2015; Springenberg et al., 2016; Perrone et al., 2018). However, these are either subject to constraints and simplifying assumptions, or must resort to Monte Carlo (MC) methods that make EI more cumbersome to evaluate and optimize.

Recognizing that the surrogate model is only a means to an end—namely, of formulating an acquisition function, we turn the spotlight away from the model and toward the acquisition function itself. To this end, we seek an alternative formulation of EI, specifically, one that potentially opens the door to more powerful estimators for which the predictive would otherwise be unwieldy or simply intractable to compute. In particular, Bergstra et al. (2011) demonstrate that the EI function can be expressed as the *relative* ratio between two densities (Yamada et al., 2011). To estimate this ratio, they propose a method known as the tree-structured Parzen estimator (TPE), which naturally handles discrete and tree-structured inputs, and scales linearly with the number of observations. However, in spite of its many advantages, TPE is not without deficiencies.

In this paper, we make the following contributions: (i) We revisit the TPE approach from first principles and identify its shortcomings in tackling the general density-ratio estimation (DRE) problem (§ 2). (ii) We propose a simple yet powerful alternative that casts the computation of EI as probabilistic classification (§ 3). This approach is built on the aforementioned link between EI and the relative density-ratio, and the correspondence between DRE and class-probability estimation (CPE). As such, it retains the strengths of the TPE method while ameliorating many of its weaknesses. Perhaps most significantly, it enables one to leverage virtually any state-of-the-art classification method available. In § 4 we discuss how our work relates to the existing state-of-the-art methods for blackbox optimization and demonstrate, through comprehensive experiments in § 5, that our approach competes well with these methods on a diverse range of problems.

2. Background

Given a blackbox function $f : \mathcal{X} \to \mathbb{R}$, the goal of BO is to find an input $\mathbf{x} \in \mathcal{X}$ at which it is minimized, given a set of N input-output observations $\mathcal{D}_N = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$, where output $y_n = f(\mathbf{x}_n) + \varepsilon$ is assumed to be observed with noise $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. In particular, having specified a probabilistic surrogate model $\mathcal{M}$, its posterior predictive $p(y \mid \mathbf{x}, \mathcal{D}_N)$ is used to compute the acquisition function $\alpha(\mathbf{x}; \mathcal{D}_N)$, a criterion that encapsulates the explore-exploit trade-off. Accordingly, candidate solutions are obtained by maximizing this criterion, $\mathbf{x}_{N+1} = \arg\max_{\mathbf{x} \in \mathcal{X}} \alpha(\mathbf{x}; \mathcal{D}_N)$. We now focus our discussion on the expected improvement (EI) function.

2.1. Expected improvement (EI)

We first specify a utility function that quantifies the nonnegative amount by which y improves upon some threshold τ, $U(\mathbf{x}, y, \tau) \coloneqq \max(\tau - y, 0)$. Then, the EI function (Mockus et al., 1978) is defined as the expected value of $U(\mathbf{x}, y, \tau)$ over the predictive

$$\alpha(\mathbf{x}; \mathcal{D}_N, \tau) \coloneqq \mathbb{E}_{p(y \mid \mathbf{x}, \mathcal{D}_N)}[U(\mathbf{x}, y, \tau)]. \tag{1}$$

By convention, τ is set to the *incumbent*, or the lowest function value so far observed $\tau = \min_n y_n$ (Wilson et al., 2018). Suppose the predictive takes the form of a Gaussian,

$$p(y \mid \mathbf{x}, \mathcal{D}_N) = \mathcal{N}\left(y \mid \mu(\mathbf{x}), \sigma^2(\mathbf{x})\right). \tag{2}$$

This leads to

$$\alpha(\mathbf{x}; \mathcal{D}_N, \tau) = \sigma(\mathbf{x}) \cdot [\nu(\mathbf{x}) \cdot \Psi(\nu(\mathbf{x})) + \psi(\nu(\mathbf{x}))], \tag{3}$$

where $\nu(\mathbf{x}) \coloneqq \frac{\tau - \mu(\mathbf{x})}{\sigma(\mathbf{x})}$, and Ψ, ψ denote the cdf and pdf of the normal distribution, respectively. While this exact expression is both easy to evaluate and optimize, the conditions necessary to satisfy eq. 2 can often come at the expense of flexibility and expressiveness. Instead, let us consider a fundamentally different way to express EI itself.

2.2. Relative density-ratio

Let $\ell(\mathbf{x})$ and $g(\mathbf{x})$ be a pair of densities. The γ-*relative density-ratio* of $\ell(\mathbf{x})$ and $g(\mathbf{x})$ is defined as

$$r_\gamma(\mathbf{x}) \coloneqq \frac{\ell(\mathbf{x})}{\gamma\ell(\mathbf{x}) + (1-\gamma)g(\mathbf{x})}, \tag{4}$$

where $\gamma\ell(\mathbf{x}) + (1-\gamma)g(\mathbf{x})$ denotes the γ-*mixture density* with mixing proportion $0 \leq \gamma < 1$ (Yamada et al., 2011). Note that for $\gamma = 0$, we recover the *ordinary* density-ratio $r_0(\mathbf{x}) = \ell(\mathbf{x})/g(\mathbf{x})$. Further, observe that $r_\gamma(\mathbf{x}) = h_\gamma(r_0(\mathbf{x}))$ where $h_\gamma : u \mapsto \left(\gamma + u^{-1}(1-\gamma)\right)^{-1}$ for $u > 0$.

We now discuss the conditions under which EI can be expressed as the ratio in eq. 4. First, set the threshold τ as the γ-th quantile of the observed y values, $\tau \coloneqq \Phi^{-1}(\gamma)$ where $\gamma = \Phi(\tau) \coloneqq p(y \leq \tau)$. Thereafter, define the pair of densities as $\ell(\mathbf{x}) \coloneqq p(\mathbf{x} \mid y \leq \tau; \mathcal{D}_N)$ and $g(\mathbf{x}) \coloneqq p(\mathbf{x} \mid y > \tau; \mathcal{D}_N)$.

An illustrated example is shown in Figure 1. Under these conditions, Bergstra et al. (2011) demonstrate that the EI function can be expressed as the relative density-ratio, up to some constant factor

$$\alpha\left(\mathbf{x}; \mathcal{D}_N, \Phi^{-1}(\gamma)\right) \propto r_\gamma(\mathbf{x}). \tag{5}$$

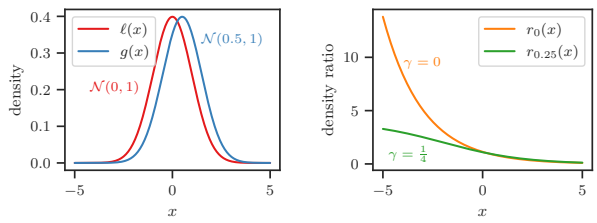


Figure 2. Gaussian densities (left) and their γ-relative density-ratios (right), which diverges when $\gamma = 0$ and converges to 4 when $\gamma = 1/4$.

For completeness, we provide a self-contained derivation in Appendix A. Thus, this reduces the problem of maximizing EI to that of maximizing the relative density-ratio,

$$\begin{aligned}\mathbf{x}_{N+1} &= \arg\max_{\mathbf{x}\in\mathcal{X}} \alpha\left(\mathbf{x}; \mathcal{D}_N, \Phi^{-1}(\gamma)\right) \\ &= \arg\max_{\mathbf{x}\in\mathcal{X}} r_\gamma(\mathbf{x}). \end{aligned} \tag{6}$$

To estimate the unknown relative density-ratio, one can appeal to a wide variety of approaches from the DRE literature (Sugiyama et al., 2012). We refer to this strategy as Bayesian optimization by density-ratio estimation (BORE).

2.3. Tree-structured Parzen estimator

The tree-structured Parzen estimator (TPE) (Bergstra et al., 2011) is an instance of the BORE framework that seeks to solve the optimization problem of eq. 6 by taking the following approach:

1. Since $r_\gamma(\mathbf{x}) = h_\gamma(r_0(\mathbf{x}))$ where h_γ is strictly non-decreasing, focus instead on maximizing[1] $r_0(\mathbf{x})$,

$$\mathbf{x}_\star = \arg\max_{\mathbf{x}\in\mathcal{X}} r_0(\mathbf{x}).$$

2. Estimate the ordinary density-ratio $r_0(\mathbf{x})$ by separately estimating its constituent numerator $\ell(\mathbf{x})$ and denominator $g(\mathbf{x})$, using a tree-based variant of kernel density estimation (KDE) (Silverman, 1986).

It is not hard to see why TPE might be favorable compared to methods based on GP regression—one now incurs an $\mathcal{O}(N)$ computational cost as opposed to the $\mathcal{O}(N^3)$ cost of GP posterior inference. Furthermore, it is equipped to deal with tree-structured, mixed continuous, ordered, and unordered discrete inputs. In spite of its advantages, TPE is not without shortcomings.

[1] $r_0(\mathbf{x})$ denotes $\gamma = 0$ solely in $r_\gamma(\mathbf{x})$ of eq. 4—it does *not* signify threshold $\tau := \Phi^{-1}(0)$, which would lead to density $\ell(\mathbf{x})$ containing no mass. We address this subtlety in Appendix B.

2.4. Potential pitfalls

The shortcomings of this approach are already well-documented in the DRE literature (Sugiyama et al., 2012). Nonetheless, we reiterate here a select few that are particularly detrimental in the context of global optimization. Namely, the first major drawback of TPE lies within step 1:

Singularities. Relying on the ordinary density-ratio can result in numerical instabilities since it is unbounded—often diverging to infinity, even in simple toy scenarios (see Figure 2 for a simple example). In contrast, the γ-relative density-ratio is always bounded above by γ^{-1} when $\gamma > 0$ (Yamada et al., 2011). The other potential problems of TPE lie within step 2:

Vapnik's principle. Conceptually, independently estimating the densities is actually a more cumbersome approach that violates Vapnik's principle—namely, that when solving a problem of interest, one should refrain from solving a more general problem as an intermediate step (Vapnik, 2013). In this instance, *density* estimation is a more general problem that is arguably more difficult than *density-ratio* estimation (Kanamori et al., 2010).

Kernel bandwidth. KDE depends crucially on the selection of an appropriate kernel bandwidth, which is notoriously difficult (Park & Marron, 1990; Sheather & Jones, 1991). Furthermore, even with an optimal selection of a single fixed bandwidth, it cannot simultaneously adapt to low- and high-density regions (Terrell & Scott, 1992).

Error sensitivity. These difficulties are exacerbated by the fact that one is required to select *two* bandwidths, whereby the optimal bandwidth for one individual density is not necessarily appropriate for estimating the density-*ratio*—indeed, it may even have deleterious effects. This also makes the approach unforgiving to misspecification of the respective estimators, particularly in that of the denominator $g(\mathbf{x})$, which has a disproportionately large influence on the resulting density-ratio.

Curse of dimensionality. For these reasons and more, KDE often falls short in high-dimensional regimes. In contrast, direct DRE methods have consistently been shown to scale better with dimensionality (Sugiyama et al., 2008).

Optimization. Ultimately, we care not only about *estimating* the density-ratio, but also *optimizing* it wrt to inputs for the purpose of candidate suggestion. Being nondifferentiable, the ratio of TPEs is cumbersome to optimize.

3. Methodology

We propose a different approach to BORE—importantly, one that circumvents the issues of TPE—by seeking to *directly* estimate the unknown ratio $r_\gamma(\mathbf{x})$.

There exists a multitude of direct DRE methods. Here, we focus on a conceptually simple and widely-used method based on class-probability estimation (CPE) (Qin, 1998; Cheng et al., 2004; Bickel et al., 2007; Sugiyama et al., 2012; Menon & Ong, 2016).

First, let $\pi(\mathbf{x}) = p(z = 1 \,|\, \mathbf{x})$ denote the *class-posterior probability*, where z is the binary class label

$$z := \begin{cases} 1 & \text{if } y \leq \tau, \\ 0 & \text{if } y > \tau. \end{cases}$$

By definition, we have $\ell(\mathbf{x}) = p(\mathbf{x} \,|\, z = 1)$ and $g(\mathbf{x}) = p(\mathbf{x} \,|\, z = 0)$. We plug these into eq. 4 and apply Bayes' rule, letting the $p(\mathbf{x})$ terms cancel each other out to give

$$r_\gamma(\mathbf{x}) = \left(\frac{p(z = 1 \,|\, \mathbf{x})}{p(z = 1)}\right) \times \left(\gamma \cdot \frac{p(z = 1 \,|\, \mathbf{x})}{p(z = 1)} + (1 - \gamma) \cdot \frac{p(z = 0 \,|\, \mathbf{x})}{p(z = 0)}\right)^{-1} \tag{7}$$

Since $p(z = 1) = \gamma$ by definition, eq. 7 simplifies to

$$r_\gamma(\mathbf{x}) = \gamma^{-1} \pi(\mathbf{x}). \tag{8}$$

Refer to Appendix C for derivations. Thus, eq. 8 establishes the link between the class-posterior probability and the relative density-ratio. In particular, the latter is equivalent to the former up to constant factor γ^{-1}. Refer to Appendix A.1 for a discussion on how $\pi(\mathbf{x})$ relates to the probability of improvement (PI) (Kushner, 1964).

Let us estimate the probability $\pi(\mathbf{x})$ using a probabilistic classifier—a function $\pi_{\boldsymbol{\theta}} : \mathcal{X} \to [0, 1]$ parameterized by $\boldsymbol{\theta}$. To recover the true class-posterior probability, we minimize a *proper scoring rule* (Gneiting & Raftery, 2007), such as the log loss

$$\mathcal{L}(\boldsymbol{\theta}) := -\frac{1}{N} \left(\sum_{n=1}^{N} z_n \log \pi_{\boldsymbol{\theta}}(\mathbf{x}_n) + (1 - z_n) \log \left(1 - \pi_{\boldsymbol{\theta}}(\mathbf{x}_n)\right) \right). \tag{9}$$

Thereafter, we approximate the relative density-ratio up to constant γ through

$$\pi_{\boldsymbol{\theta}}(\mathbf{x}) \simeq \gamma \cdot r_\gamma(\mathbf{x}), \tag{10}$$

with equality at $\boldsymbol{\theta}_\star = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta})$. Refer to Appendix D for derivations. Hence, in the so-called BO loop (summarized in Algorithm 1), we alternately optimize (i) the classifier parameters $\boldsymbol{\theta}$ wrt to the log loss (to improve the approximation of eq. 10; Line 6), and (ii) the classifier input $\mathbf{x}$ wrt to its output (to suggest the next candidate to evaluate; Line 8). An animation of Algorithm 1 is provided in Appendix E.

Algorithm 1: Bayesian optimization by density-ratio estimation (BORE).

Input: blackbox $f : \mathcal{X} \to \mathbb{R}$, proportion $\gamma \in (0, 1)$, probabilistic classifier $\pi_{\boldsymbol{\theta}} : \mathcal{X} \to [0, 1]$.

1 **while** *under budget* **do**
2 $\quad \tau \leftarrow \Phi^{-1}(\gamma)$ // compute γ-th quantile of $\{y_n\}_{n=1}^N$
3 $\quad z_n \leftarrow \mathbb{I}[y_n \leq \tau]$ for $n = 1, \ldots, N$ // assign labels
4 $\quad \tilde{\mathcal{D}}_N \leftarrow \{(\mathbf{x}_n, z_n)\}_{n=1}^N$ // construct auxiliary dataset
5 $\quad$ /* update classifier by optimizing parameters $\boldsymbol{\theta}$ wrt log loss */
6 $\quad \boldsymbol{\theta}_\star \leftarrow \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta})$ // depends on $\tilde{\mathcal{D}}_N$, see eq. 9
7 $\quad$ /* suggest candidate by optimizing input $\mathbf{x}$ wrt classifier */
8 $\quad \mathbf{x}_N \leftarrow \arg\max_{\mathbf{x} \in \mathcal{X}} \pi_{\boldsymbol{\theta}_\star}(\mathbf{x})$ // see eq. 10
9 $\quad y_N \leftarrow f(\mathbf{x}_N)$ // evaluate blackbox function
10 $\quad \mathcal{D}_N \leftarrow \mathcal{D}_{N-1} \cup \{(\mathbf{x}_N, y_N)\}$ // update dataset
11 $\quad N \leftarrow N + 1$
12 **end**

In traditional GP-based EI, Line 8 typically consists of maximizing the EI function expressed in the form of eq. 3, while Line 6 consists of optimizing the GP hyperparameters wrt the marginal likelihood. By analogy with our approach, the parameterized function $\pi_{\boldsymbol{\theta}}(\mathbf{x})$ is *itself* an approximation to the EI function to be maximized directly, while the approximation is tightened through by optimizing the classifier parameters wrt the log loss. In short, we have reduced the problem of computing EI to that of learning a probabilistic classifier, thereby unlocking a broad range of estimators beyond those so far used in BO. Importantly, this enables one to employ virtually any state-of-the-art classification method available and to parameterize the classifier using arbitrarily expressive approximators that potentially have the capacity to deal with non-linear, non-stationary, and heteroscedastic phenomena frequently encountered in practice.

3.1. Choice of proportion γ

The proportion $\gamma \in (0, 1)$ influences the explore-exploit trade-off. Intuitively, a smaller setting of γ encourages exploitation and leads to fewer modes and sharper peaks in the acquisition function. To see this, consider that there are by definition fewer candidate inputs $\mathbf{x}$ for which its corresponding output y can be expected to improve over the first quartile ($\gamma = 1/4$) of the observed output values than, say, the third quartile ($\gamma = 3/4$). That being said, given that the class balance rate is by definition γ, a value too close to 0 may lead to instabilities in classifier learning. A potential strategy to combat this is to begin with a perfect balance ($\gamma = 1/2$) and then to decay γ as optimization progresses. In this work, we keep γ fixed throughout optimization. This, on the other hand, has the benefit of providing guarantees about how the classification task evolves. In particular, in each iteration, after having observed a new evaluation, we

are guaranteed that the binary label of *at most* one existing instance can flip. This property can be exploited to make classifier learning of Line 6 more efficient by adopting online learning techniques that avoid learning from scratch in each iteration. An extended discussion is included in Appendix F.

3.2. Choice of probabilistic classifier

We examine a few variations of BORE that differ in the choice of classifier and discuss their strengths and weaknesses across different global optimization problem settings.

Multi-layer perceptrons. We propose BORE-MLP, a variant based on multi-layer perceptrons (MLPs). This choice is appealing not only for (i) its flexibility and universal approximation guarantees (Hornik et al., 1989) but because (ii) one can easily adopt stochastic gradient descent (SGD) methods to scale up its parameter learning (LeCun et al., 2012), and (iii) it is differentiable end-to-end, thus enabling the use of quasi-Newton methods such as L-BFGS (Liu & Nocedal, 1989) for candidate suggestion. Lastly, since SGD is online by nature, (iv) it is feasible to adapt weights from previous iterations instead of training from scratch. A notable weakness is that MLPs can be over-parameterized and therefore considerably data-hungry.

Tree-based ensembles. We consider two further variants: BORE-RF and BORE-XGB, both based on ensembles of decision trees—namely, random forest (RF) (Breiman, 2001) and gradient-boosted trees (XGBOOST) (Chen & Guestrin, 2016), respectively. These variants are attractive since they inherit from decision trees the ability to (i) deal with discrete and conditional inputs by design, (ii) work well in high-dimensions, and (iii) are scalable and easily parallelizable. Further, (iv) online extensions of RFs (Saffari et al., 2009) may be applied to avoid training from scratch. A caveat is that, since their response surfaces are discontinuous and nondifferentiable, decision trees are difficult to maximize. Therefore, we appeal to random search and evolutionary strategies for candidate suggestion. Further details and a comparison of various approaches is included in Appendix G.1.

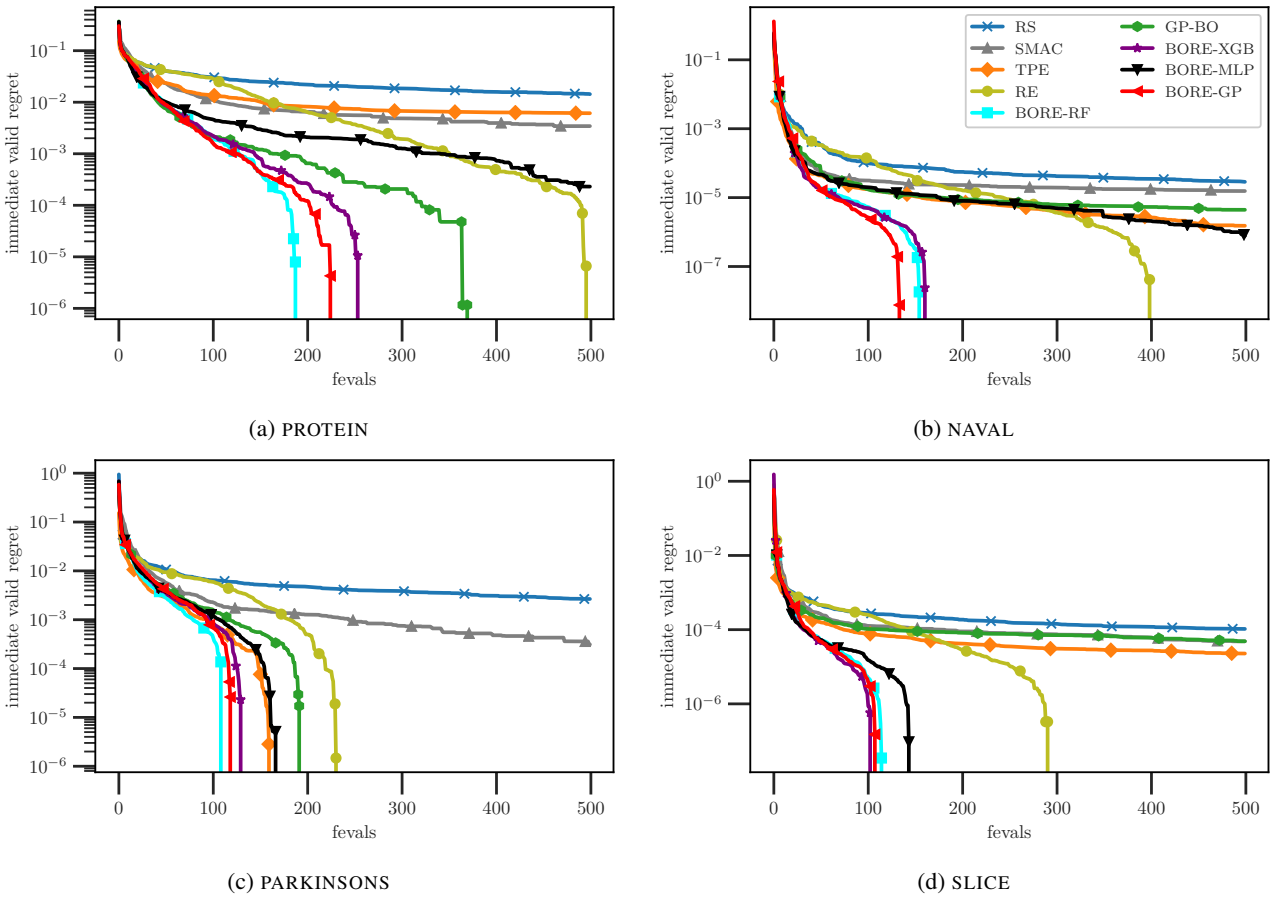
In theory, for the approximation of eq. 10 to be tight, the classifier is required to produce well-calibrated probabilities (Menon & Ong, 2016). A potential drawback of the BORE-RF variant is that RFs are generally not trained by minimizing a proper scoring rule. As such, additional techniques may be necessary to improve calibration (Niculescu-Mizil & Caruana, 2005).

Gaussian processes. The last variant we consider is BORE-GP, based on a GP classifier (GPC) (Williams & Barber, 1998). Like the GP regression model, GPC offers (i) a high degree of flexibility, at least on smooth functions up to moderate dimensionalities, and (ii) well-calibrated uncertainty estimates (useful for marginalizing out the hyperparameters from the acquisition function, as we discuss in Appendix L). On the other hand, GPC not only loses one of the foremost appeals of GP regression, namely, analytical tractability of the predictive, but it is also not necessarily better equipped to deal with more problematic settings (discrete variables, high-dimensionalities, etc), and its scalability is contingent upon the choice of inference approximation being utilized.

4. Related Work

The literature on BO is vast and ever-expanding (Brochu et al., 2010; Shahriari et al., 2015; Frazier, 2018). Some specific threads pertinent to our work include achieving scalability through neural networks (NNs), as in BANANAS (White et al., 2019), ABLR (Perrone et al., 2018), BOHAMIANN (Springenberg et al., 2016), and DNGO (Snoek et al., 2015), and handling discrete and conditional variables using tree ensembles, as with RFs in SMAC (Hutter et al., 2011). To negotiate the tractability of the predictive, these methods must either make simplifications or resort to approximations. In contrast, by seeking to directly approximate the acquisition function, BORE is unencumbered by such constraints. Refer to Appendix L for an expanded discussion. Beyond the classical PI (Kushner, 1964) and EI functions (Jones et al., 1998), a multitude of acquisition functions has been devised, including the upper confidence bound (UCB) (Srinivas et al., 2009), knowledge gradient (KG) (Scott et al., 2011), entropy search (ES) (Hennig & Schuler, 2012), and predictive ES (PES) (Hernández-Lobato et al., 2014). Nonetheless, EI remains ubiquitous in large because it is conceptually simple, easy to evaluate and optimize, and consistently performs well in practice.

There is a substantial body of existing works on density-ratio estimation (Sugiyama et al., 2012). Recognizing the deficiencies of the KDE approach, myriad alternatives have since been proposed, including KL importance estimation procedure (KLIEP) (Sugiyama et al., 2008), kernel mean matching (KMM) (Gretton et al., 2009), unconstrained least-squares importance fitting (ULSIF) (Kanamori et al., 2009), and relative ULSIF (RULSIF) (Yamada et al., 2011). In this work, we restrict our focus on CPE, an effective and versatile approach that has found widespread adoption in a diverse range of applications, e.g. in covariate shift adaptation (Bickel et al., 2007), energy-based modelling (Gutmann & Hyvärinen, 2012), generative adversarial networks (GANs) (Goodfellow et al., 2014; Nowozin et al., 2016), likelihood-free inference (Tran et al., 2017; Thomas et al., 2020), and more. Of particular relevance is its use in Bayesian experimental design (BED), a close relative of BO, in which it is similarly used to approximate the expected utility function (Kleinegesse & Gutmann, 2019).



(a) PROTEIN (b) NAVAL

(c) PARKINSONS (d) SLICE

Figure 3. Immediate regret over function evaluations on the HPOBench neural network tuning problems ($D = 9$).

5. Experiments

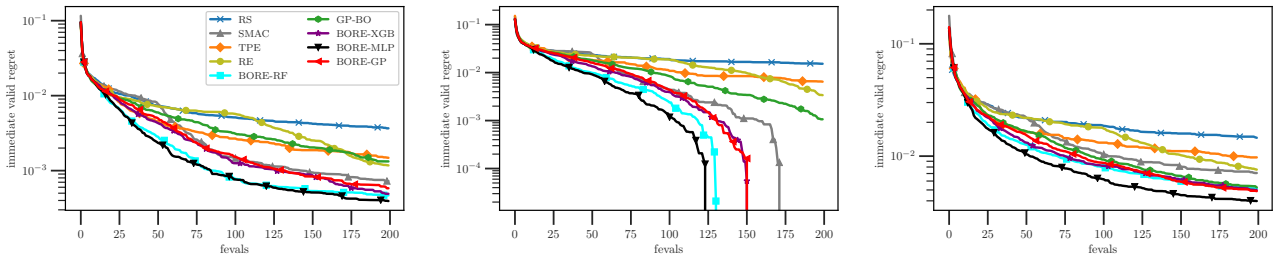
We describe the experiments conducted to empirically evaluate our method. To this end, we consider a variety of problems, ranging from automated machine learning (AUTOML), robotic arm control, to racing line optimization.

We provide comparisons against a comprehensive selection of state-of-the-art baselines. Namely, across all problems, we consider random search (RS) (Bergstra & Bengio, 2012), GP-BO (using EI with $\gamma = 0$) (Jones et al., 1998), TPE (Bergstra et al., 2011), and SMAC (Hutter et al., 2011). We also consider evolutionary strategies: differential evolution (DE) (Storn & Price, 1997) for problems with continuous domains, and regularized evolution (RE) (Real et al., 2019) for those with discrete domains. Further information about these baselines and the source code for their implementations are included in Appendix I.

To quantitatively assess performance we report the *immediate regret* (in benchmarks for which the exact global minimum is known), defined as the absolute error between the global minimum and the lowest function value attained thus far. Unless otherwise stated we report, for each benchmark and method, results aggregated across 100 replicated runs.

We set $\gamma = 1/3$ across all variants and benchmarks. For candidate suggestion in the tree-based variants, we use RS with a function evaluation limit of 500 for problems with discrete domains, and DE with a limit of 2,000 for those with continuous domains. Our open-source implementation is available at `https://github.com/ltiao/bore`. Further details concerning the experimental set-up and the implementation of each variant are included in Appendix J.

Neural network tuning (HPOBench). First, we consider the problem of training a two-layer feed-forward NN for regression. Specifically, a NN is trained for 100 epochs with the ADAM optimizer (Kingma & Ba, 2014), and the objective is the validation mean-squared error (MSE). The hyperparameters are the *initial learning rate*, *learning rate schedule*, *batch size*, along with the layer-specific *widths*, *activations*, and *dropout rates*. We consider four datasets: PROTEIN, NAVAL, PARKINSONS, and SLICE, and utilize HPOBench (Klein & Hutter, 2019) which tabulates, for each dataset, the MSEs resulting from all possible (62,208) configurations. Additional details are included in Appendix K.1,



(a) CIFAR-10 (b) CIFAR-100 (c) ImageNet-16

Figure 4. Immediate regret over function evaluations on the NASBench201 neural architecture search problems ($D = 6$).

and the results are shown in Figure 3. We see across all datasets that the BORE-RF and -XGB variants consistently outperform all other baselines, converging rapidly toward the global minimum after 1-2 hundred evaluations—in some cases, earlier than any other baseline by over two hundred evaluations. Notably, with the exception being BORE-MLP on the PARKINSONS dataset, all BORE variants outperform TPE, in many cases by a sizable margin.

Neural architecture search (NASBench201). Next, we consider a neural architecture search (NAS) problem, namely, that of designing a neural cell. A cell is represented by a directed acyclic graph (DAG) with 4 nodes, and the task is to assign an *operation* to each of the 6 possible arcs from a set of five operations. We utilize NASBench201 (Dong & Yang, 2020), which tabulates precomputed results from all possible $5^6 = 15,625$ combinations for each of the three datasets: CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009), and ImageNet-16 (Chrabaszcz et al., 2017). Additional details are included in Appendix K.2, and the results are shown in Figure 4. We find across all datasets that the BORE variants consistently achieve the lowest final regret among all baselines. Not only that, the BORE variants, in particular BORE-MLP, maintains the lowest regret at anytime (i.e. at any optimization iteration), followed by BORE-RF, then BORE-XGB/-GP. In this problem, the inputs are purely categorical, whereas in the previous problem they are a mix of categorical and ordinal. For the BORE-MLP variant, categorical inputs are one-hot encoded, while ordinal inputs are handled by simply rounding to their nearest integer index. The latter is known to have shortcomings (Garrido-Merchán & Hernández-Lobato, 2020), and might explain why BORE-MLP is the most effective variant in this problem but the least effective in the previous one.

Robot arm pushing. We consider the 14D control problem first studied by Wang & Jegelka (2017). The problem is concerned with tuning the controllers of robot hands to push objects to some desired locations. Specifically, there are two robots, each tasked with manipulating an object. For each robot, the control parameters include the *location* and *orientation* of its hands, the *moving direction*, *pushing speed*,

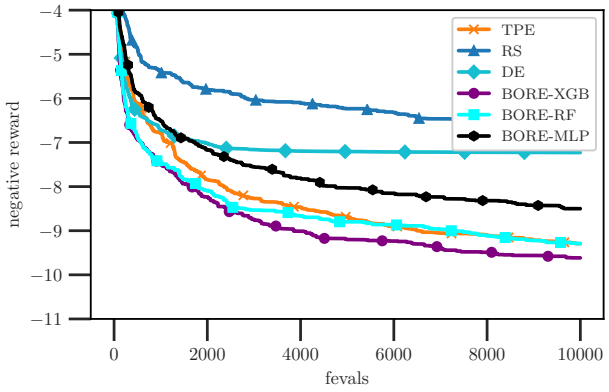
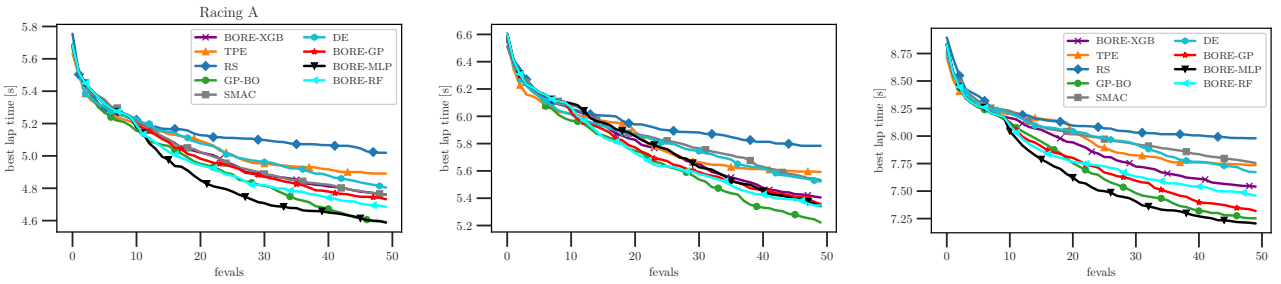


Figure 5. Negative reward over function evaluations on the Robot Pushing task ($D = 14$).

and *duration*. Due to the prohibitively large number of function evaluations (∼10,000) required to achieve reasonable performance, we omit all GP-based methods from our comparisons on this benchmark. Further, we reduce the number of replicated runs of each method to 50. Additional details are included in Appendix K.3, and the results are shown in Figure 5. We see that BORE-XGB attains the highest reward, followed by BORE-RF and TPE (which attain roughly the same performance), and then BORE-MLP.

Racing line optimization. We consider the problem of computing the optimal racing line for a given track and vehicle with known dynamics. We adopt the set-up of Jain & Morari (2020), who consider the dynamics of miniature scale cars traversing the tracks at UC BERKELEY and ETH ZÜRICH. The racing line is a trajectory determined by D waypoints placed along the length of the track, where the ith waypoint deviates from the centerline of the track by $x_i \in \left[-\frac{W}{2}, \frac{W}{2}\right]$ for some track width W. The task is to minimize the lap time $f(\mathbf{x})$, the minimum time required to traverse the trajectory parameterized by $\mathbf{x} = [x_1 \cdots x_D]^\top$. Additional details are included in Appendix K.4, and the results are shown in Figure 6. First, we see that the BORE variants consistently outperform all baselines except for GP-BO. This is to be expected since the function is continuous,



(a) UC BERKELEY ($D = 12$) (b) ETH ZÜRICH A ($D = 20$) (c) ETH ZÜRICH B ($D = 21$)

Figure 6. Best lap times (in seconds) over function evaluations in the racing line optimization problem on various racetracks.

smooth, and has ∼20 dimensions or less. Nonetheless, we find that the BORE-MLP variant performs as well as, or marginally better than, GP-BO on two tracks. In particular, on the UC BERKELEY track, we see that BORE-MLP achieves the best lap times for the first ∼40 evaluations, and is caught up to by GP-BO in the final 10. On ETH ZÜRICH track B, BORE-MLP consistently maintains a narrow lead.

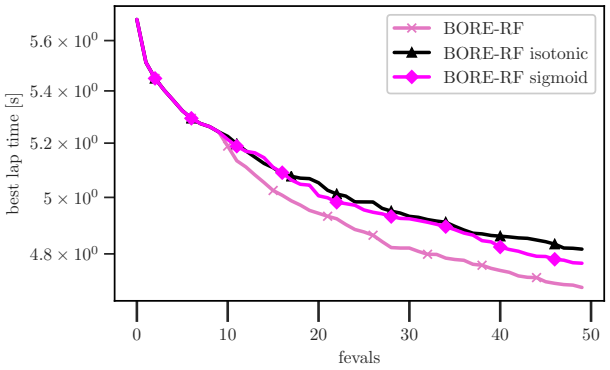


Figure 7. Effects of calibrating RFs in the BORE-RF variant. Results of racing line optimization on the UC BERKELEY track.

Effects of calibration. As discussed in § 3.2, calibrating RFs may have a profound effect on the BORE-RF variant. We consider two popular approaches (Niculescu-Mizil & Caruana, 2005), namely, Platt scaling (Platt et al., 1999) and isotonic regression (Zadrozny & Elkan, 2001; 2002). The results shown in Figure 7 suggest that applying these calibration techniques may have deleterious effects. However, this can also be adequately explained by overfitting due to insufficient calibration samples (in the case of isotonic regression, ∼1,000 samples are necessary). Therefore, we may yet observe the benefits of calibration in problem settings that yield large amounts of data.

We provide further ablation studies in Appendix G.

6. Discussion and Outlook

We examine the limitations of our method, discuss how these may be addressed, and outline additional future directions.

Exploration. Similar to the TPE method, BORE generally has a tendency to favor exploitation over exploration. In the case of TPE, the maximizer of the acquisition function $\ell(\mathbf{x})/g(\mathbf{x})$ will be located at the mode of $\ell(\mathbf{x})$, which has mass concentrated around inputs for which its output value is within the smallest proportion γ of all observed output values (i.e. inputs with label $z = 1$). Recall the classical formulation of EI from eq. 3 in which the explore-exploit trade-off is explicitly encoded in mathematical terms. Assuming we had access to its global optimum, then by design the solution is a candidate that strikes a good balance between exploration and exploitation. Indeed, by virtue of having lower predictive uncertainty, previously evaluated candidates will tend to have lower acquisition values, which helps to encourage exploration. In contrast, for TPE and BORE, the previously evaluated candidates labeled $z = 1$ will tend to retain high acquisition values. Therefore, in the worst-case scenario, the global optimum of the acquisition function may become stuck at some local optimum of the blackbox function, or a point within some neighborhood thereof. In practice, implementations of TPE avoid this scenario by introducing stochasticity in the acquisition optimization, e.g. by randomly sampling from $\ell(\mathbf{x})$ and suggesting the sample that maximizes $\ell(\mathbf{x})/g(\mathbf{x})$. We surmise that BORE was able to avoid such pathological cases in our experiments due in part to the sources of randomness inherent to the acquisition optimization method of choice.

A further detail to note is that the labels z do not remain static throughout optimization. In other words, the classification dataset is different for each new iteration. Recall that, by construction, only a fraction γ of the observations can have positive labels $z = 1$. With each iteration, observing a new value of y leads to a change in the threshold τ. Since only a fraction γ of observations can lie below this threshold, the labels of existing observations must accordingly flip intermittently throughout optimization. Thus, as

the probabilistic classifier $\pi_{\boldsymbol{\theta}}(\mathbf{x})$ adapts to these updates, the regions in which it outputs high probabilities will also shift accordingly. Consequently, the classifier response surface will either become multimodal (leading to exploration) or become narrower and more sharply-peaked in the same region (leading to exploitation).

Although not considered in this work, the behavior described above can make simple ϵ-greedy strategies particularly effective at stimulating exploration. Future work will consider batch extensions based on methods such as quantile Stein variational gradient descent (SVGD), which can encourage high diversity and good worst-case performance in the query batch (Gong et al., 2019).

Hyperparameter estimation. Firstly, a noteworthy consequence of seeking to directly approximate EI under its alternative formulation is that the classifier parameters $\boldsymbol{\theta}$ in BORE can be interpreted as *hyperparameters* (in the same way that the parameters of the GP kernel are hyperparameters), a deterministic treatment of which based on point estimates can often be viable. For example, in the BORE-MLP variant, $\boldsymbol{\theta}$ consists of the layer weights, which we are able to estimate using type-II maximum likelihood. In contrast, to utilize NNs in traditional BO, generally the layer weights $\boldsymbol{\omega}$ are parameters that must first be marginalized out in order to compute the predictive $p_{\boldsymbol{\theta}}(y \mid \mathbf{x}, \mathcal{D}_N) = \int p_{\boldsymbol{\theta}}(y \mid \mathbf{x}, \boldsymbol{\omega}) p_{\boldsymbol{\theta}}(\boldsymbol{\omega} \mid \mathcal{D}_N) \, \mathrm{d}\boldsymbol{\omega}$, while the hyperparameters $\boldsymbol{\theta}$, consisting of e.g. the prior and likelihood precisions, may optionally be marginalized out as well (though usually point estimates suffice). Refer to Appendix L for an expanded discussion on this distinction. As with the GP hyperparameters in GP-BO, in order to encourage exploration, it may be beneficial to consider placing a prior on $\boldsymbol{\theta}$ and marginalizing out its uncertainty (Snoek et al., 2012). Further, compared against GP-BO, a potential downside of BORE is that there may be vastly more *meta-hyperparameters* settings from which to choose. Whereas in GP-BO these might consist of, e.g. the choice of kernel and its isotropy, there are potentially many more possibilities in BORE. In BORE-MLP, this may consist of, e.g. layer depth, widths, activations, etc—the tuning of which is often the reason one appeals to BO in the first place. While we obtained remarkable results with the proposed variants without needing to deviate from the sensible defaults, in general, for further improvements in calibration and sample diversity, it may be beneficial to consider marginalizing out even the meta-hyperparameters (Wenzel et al., 2020).

Direct DRE. Another avenue to explore is the potential benefits of other direct DRE methods, in particular RULSIF (Yamada et al., 2011), which is the only method of those aforementioned in § 4 that directly estimates the *relative* density-ratio. Furthermore, since RULSIF is parameterized by a sum of Gaussian kernels, it enables the use of well-established mode-finding approaches, such as the *mean-shift* algorithm (Comaniciu & Meer, 2002), for candidate suggestion. Along the same avenue, but in a different direction, one may also consider employing DRE losses for classifier learning (Menon & Ong, 2016).

Extended BO. Lastly, a fertile ground for future work lies in the extension of BORE with classifier designs suitable for BO in more sophisticated paradigms, such as in the multi-task (Swersky et al., 2013), multi-fidelity (Kandasamy et al., 2017), and multi-objective settings (Hernández-Lobato et al., 2016). Of particular interest is the use of model architectures that are effective for BO on sequential inputs (Moss et al., 2020) which can be applied to molecular structures (Gómez-Bombarelli et al., 2018) and beyond.

7. Conclusion

We have presented a novel methodology for BO based on the observation that the problem of computing EI can be reduced to that of probabilistic classification. This observation is made through the well-known link between CPE and DRE, and the lesser-known insight that EI can be expressed as a relative density-ratio between two unknown distributions.

We discussed important ways in which TPE, an early attempt to exploit the latter link, falls short. Further, we demonstrated that our CPE-based approach to BORE, in particular, our variants based on the MLP, RF, XGBOOST, and GP classifiers, consistently outperform TPE, and compete well against the state-of-the-art derivative-free global optimization methods.

Overall, the simplicity and effectiveness of BORE make it a promising approach for blackbox optimization, and its high degree of extensibility provides numerous exciting avenues for future work.

Acknowledgements

We are grateful to the anonymous reviewers for their helpful feedback and suggestions. LT is supported by the Australian Government Research Training Program (RTP) Scholarship and the CSIRO's Data61 Postgraduate Scholarship.

References

Balandat, M., Karrer, B., Jiang, D., Daulton, S., Letham, B., Wilson, A. G., and Bakshy, E. BoTorch: A framework for efficient Monte-Carlo Bayesian optimization. *Advances in Neural Information Processing Systems*, 33, 2020.

Bergstra, J. and Bengio, Y. Random search for hyper-parameter optimization. *The Journal of Machine Learning Research*, 13(1):281–305, 2012.

Bergstra, J. S., Bardenet, R., Bengio, Y., and Kégl, B. Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems*, pp. 2546–2554, 2011.

Bickel, S., Brückner, M., and Scheffer, T. Discriminative learning for differing training and test distributions. In *Proceedings of the 24th International Conference on Machine Learning*, pp. 81–88, 2007.

Breiman, L. Random forests. *Machine Learning*, 45(1): 5–32, 2001.

Brochu, E., Cora, V. M., and De Freitas, N. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010.

Chen, T. and Guestrin, C. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.

Cheng, K. F., Chu, C.-K., et al. Semiparametric density estimation under a two-sample density ratio model. *Bernoulli*, 10(4):583–604, 2004.

Chrabaszcz, P., Loshchilov, I., and Hutter, F. A downsampled variant of imagenet as an alternative to the cifar datasets. *arXiv preprint arXiv:1707.08819*, 2017.

Comaniciu, D. and Meer, P. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

Dong, X. and Yang, Y. Nas-bench-102: Extending the scope of reproducible neural architecture search. *arXiv preprint arXiv:2001.00326*, 2020.

Frazier, P. I. A tutorial on Bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018.

Garrido-Merchán, E. C. and Hernández-Lobato, D. Dealing with categorical and integer-valued variables in Bayesian optimization with Gaussian processes. *Neurocomputing*, 380:20–35, 2020.

Gneiting, T. and Raftery, A. E. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276, 2018.

Gong, C., Peng, J., and Liu, Q. Quantile stein variational gradient descent for batch Bayesian optimization. In *International Conference on Machine Learning*, pp. 2347–2356. PMLR, 2019.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27:2672–2680, 2014.

Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., and Schölkopf, B. Covariate shift by kernel mean matching. *Dataset Shift in Machine Learning*, 3(4): 5, 2009.

Gutmann, M. U. and Hyvärinen, A. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *The Journal of Machine Learning Research*, 13(1):307–361, 2012.

Hennig, P. and Schuler, C. J. Entropy search for information-efficient global optimization. *The Journal of Machine Learning Research*, 13(1):1809–1837, 2012.

Hernández-Lobato, D., Hernandez-Lobato, J., Shah, A., and Adams, R. Predictive entropy search for multi-objective Bayesian optimization. In *International Conference on Machine Learning*, pp. 1492–1501, 2016.

Hernández-Lobato, J. M., Hoffman, M. W., and Ghahramani, Z. Predictive entropy search for efficient global optimization of black-box functions. *Advances in Neural Information Processing Systems*, 27:918–926, 2014.

Hornik, K., Stinchcombe, M., White, H., et al. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

Hutter, F., Hoos, H. H., and Leyton-Brown, K. Sequential model-based optimization for general algorithm configuration. In *International Conference on Learning and Intelligent Optimization*, pp. 507–523. Springer, 2011.

Jain, A. and Morari, M. Computing the racing line using Bayesian optimization. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pp. 6192–6197. IEEE, 2020.

Jenatton, R., Archambeau, C., González, J., and Seeger, M. Bayesian optimization with tree-structured dependencies. In *International Conference on Machine Learning*, pp. 1655–1664, 2017.

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

Kanamori, T., Hido, S., and Sugiyama, M. A least-squares approach to direct importance estimation. *The Journal of Machine Learning Research*, 10:1391–1445, 2009.

Kanamori, T., Suzuki, T., and Sugiyama, M. Theoretical analysis of density ratio estimation. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 93(4):787–798, 2010.

Kandasamy, K., Dasarathy, G., Schneider, J., and Póczos, B. Multi-fidelity Bayesian optimisation with continuous approximations. *arXiv preprint arXiv:1703.06240*, 2017.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Klein, A. and Hutter, F. Tabular benchmarks for joint architecture and hyperparameter optimization. *arXiv preprint arXiv:1905.04970*, 2019.

Kleinegesse, S. and Gutmann, M. U. Efficient Bayesian experimental design for implicit models. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 476–485. PMLR, 2019.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Kushner, H. J. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. 1964.

LeCun, Y. A., Bottou, L., Orr, G. B., and Müller, K.-R. Efficient backprop. In *Neural Networks: Tricks of the Trade*, pp. 9–48. Springer, 2012.

Liu, D. C. and Nocedal, J. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1-3):503–528, 1989.

Menon, A. and Ong, C. S. Linking losses for density ratio and class-probability estimation. In *International Conference on Machine Learning*, pp. 304–313, 2016.

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

Moss, H. B., Beck, D., González, J., Leslie, D. S., and Rayson, P. Boss: Bayesian optimization over string spaces. *arXiv preprint arXiv:2010.00979*, 2020.

Niculescu-Mizil, A. and Caruana, R. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning*, pp. 625–632, 2005.

Nowozin, S., Cseke, B., and Tomioka, R. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*, pp. 271–279, 2016.

Park, B. U. and Marron, J. S. Comparison of data-driven bandwidth selectors. *Journal of the American Statistical Association*, 85(409):66–72, 1990.

Perrone, V., Jenatton, R., Seeger, M. W., and Archambeau, C. Scalable hyperparameter transfer learning. In *Advances in Neural Information Processing Systems*, pp. 6845–6855, 2018.

Platt, J. et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 10(3):61–74, 1999.

Qin, J. Inferences for case-control and semiparametric two-sample density ratio models. *Biometrika*, 85(3):619–630, 1998.

Real, E., Aggarwal, A., Huang, Y., and Le, Q. V. Regularized evolution for image classifier architecture search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 4780–4789, 2019.

Saffari, A., Leistner, C., Santner, J., Godec, M., and Bischof, H. On-line random forests. In *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pp. 1393–1400. IEEE, 2009.

Scott, W., Frazier, P., and Powell, W. The correlated knowledge gradient for simulation optimization of continuous parameters using Gaussian process regression. *SIAM Journal on Optimization*, 21(3):996–1026, 2011.

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and De Freitas, N. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015.

Sheather, S. J. and Jones, M. C. A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society: Series B (Methodological)*, 53(3):683–690, 1991.

Silverman, B. W. *Density estimation for statistics and data analysis*, volume 26. CRC Press, 1986.

Snoek, J., Larochelle, H., and Adams, R. P. Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, 25: 2951–2959, 2012.

Snoek, J., Swersky, K., Zemel, R., and Adams, R. Input warping for Bayesian optimization of non-stationary functions. In *International Conference on Machine Learning*, pp. 1674–1682, 2014.

Snoek, J., Rippel, O., Swersky, K., Kiros, R., Satish, N., Sundaram, N., Patwary, M., Prabhat, M., and Adams, R. Scalable Bayesian optimization using deep neural networks. In *International Conference on Machine Learning*, pp. 2171–2180, 2015.

Springenberg, J. T., Klein, A., Falkner, S., and Hutter, F. Bayesian optimization with robust Bayesian neural networks. In *Advances in Neural Information Processing Systems*, pp. 4134–4142, 2016.

Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

Storn, R. and Price, K. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 1997.

Sugiyama, M., Nakajima, S., Kashima, H., Buenau, P. V., and Kawanabe, M. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Advances in Neural Information Processing Systems*, pp. 1433–1440, 2008.

Sugiyama, M., Suzuki, T., and Kanamori, T. *Density Ratio Estimation in Machine Learning*. Cambridge University Press, 2012.

Swersky, K., Snoek, J., and Adams, R. P. Multi-task Bayesian optimization. In *Advances in Neural Information Processing Systems*, pp. 2004–2012, 2013.

Terrell, G. R. and Scott, D. W. Variable kernel density estimation. *The Annals of Statistics*, pp. 1236–1265, 1992.

Thomas, O., Dutta, R., Corander, J., Kaski, S., Gutmann, M. U., et al. Likelihood-free inference by ratio estimation. *Bayesian Analysis*, 2020.

Tran, D., Ranganath, R., and Blei, D. Hierarchical implicit models and likelihood-free variational inference. In *Advances in Neural Information Processing Systems*, pp. 5523–5533, 2017.

Vapnik, V. *The nature of statistical learning theory*. Springer Science & Business Media, 2013.

Wang, Z. and Jegelka, S. Max-value entropy search for efficient Bayesian optimization. *arXiv preprint arXiv:1703.01968*, 2017.

Wenzel, F., Snoek, J., Tran, D., and Jenatton, R. Hyperparameter ensembles for robustness and uncertainty quantification. *arXiv preprint arXiv:2006.13570*, 2020.

White, C., Neiswanger, W., and Savani, Y. Bananas: Bayesian optimization with neural architectures for neural architecture search. *arXiv preprint arXiv:1910.11858*, 2019.

Williams, C. K. and Barber, D. Bayesian classification with Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1342–1351, 1998.

Williams, C. K. and Rasmussen, C. E. Gaussian processes for regression. In *Advances in Neural Information Processing Systems*, pp. 514–520, 1996.

Wilson, J., Hutter, F., and Deisenroth, M. Maximizing acquisition functions for Bayesian optimization. *Advances in Neural Information Processing Systems*, 31:9884–9895, 2018.

Yamada, M., Suzuki, T., Kanamori, T., Hachiya, H., and Sugiyama, M. Relative density-ratio estimation for robust distribution comparison. In *Advances in Neural Information Processing Systems*, pp. 594–602, 2011.

Zadrozny, B. and Elkan, C. Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers. In *ICML*, volume 1, pp. 609–616. Citeseer, 2001.

Zadrozny, B. and Elkan, C. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 694–699, 2002.