Preferential Bayesian Optimization

Javier González [1] Zhenwen Dai [1] Andreas Damianou [1] Neil D. Lawrence [1 2]

Abstract

Bayesian optimization (BO) has emerged during the last few years as an effective approach to optimizing *black-box* functions where direct queries of the objective are expensive. In this paper we consider the case where direct access to the function is not possible, but information about user preferences is. Such scenarios arise in problems where human preferences are modeled, such as A/B tests or recommender systems. We present a new framework for this scenario that we call *Preferential Bayesian Optimization (*PBO*)* which allows us to find the optimum of a latent function that can only be queried through pairwise comparisons, the so-called *duels*. PBO extends the applicability of standard BO ideas and generalizes previous discrete dueling approaches by modeling the probability of the winner of each duel by means of a Gaussian process model with a Bernoulli likelihood. The latent preference function is used to define a family of acquisition functions that extend usual policies used in BO. We illustrate the benefits of PBO in a variety of experiments, showing that PBO needs drastically fewer comparisons for finding the optimum. According to our experiments, the way of modeling correlations in PBO is key in obtaining this advantage.

1. Introduction

Let $g : \mathcal{X} \rightarrow \Re$ be a well-behaved *black-box* function defined on a bounded subset $\mathcal{X} \subseteq \Re^q$. We are interested in solving the global optimization problem of finding

$$\mathbf{x}_{min} = \arg\min_{\mathbf{x}\in\mathcal{X}} g(\mathbf{x}). \quad (1)$$

We assume that g is not directly accessible and that queries to g can only be done in pairs of points or *duels* $[\mathbf{x}, \mathbf{x}'] \in \mathcal{X} \times \mathcal{X}$ from which we obtain binary feedback $\{0, 1\}$ that represents whether or not $\mathbf{x}$ is preferred over $\mathbf{x}'$ (has lower value)[1]. We will consider that $\mathbf{x}$ is the winner of the duel if the output is $\{1\}$ and that $\mathbf{x}'$ wins the duel if the output is $\{0\}$. The goal here is to find $\mathbf{x}_{min}$ by reducing as much as possible the number of queried duels.

Our setup is different to the one typically used in BO where direct feedback from g in the domain is available (Jones, 2001; Snoek et al., 2012). In our context, the objective is a latent object that is only accessible via indirect observations. However, although the scenario described in this work has not received a wider attention, there exist a variety of real wold scenarios in which the objective function needs to be optimized via preferential returns. Most cases involve modeling *latent human preferences*, such as web design via A/B testing, the use of recommender systems (Brusilovsky et al., 2007) or the ranking of game players skills (Herbrich et al., 2007). In prospect theory, the models used are based on comparisons with some reference point, as it has been demonstrated that humans are better at evaluating differences rather than absolute magnitudes (Kahneman and Tversky, 1979).

Optimization methods for pairwise preferences have been studied in the armed-bandits context (Yuea et al., 2012). Zoghi et al. (2014) propose a new method for the K-armed duelling bandit problem based on the Upper Confidence Bound algorithm. Jamieson et al. (2015) study the problem by allowing noise comparisons between the duels. Zoghi et al. (2015b) choose actions using contextual information. Dudík et al. (2015) study the Copeland's dueling bandits, a case in which a Condorcet winner, or an arm that uniformly wins the duels with all the other arms may not exist. Szörényi et al. (2015) study Online Rank Elicitation problem in the duelling bandits setting. An analysis on Thompson sampling in duelling bandits is done by Wu et al. (2016). Yue and Joachims (2011) proposes a method that does not need transitivity and comparison outcomes to have independent and time-stationary distributions.

Preference learning has also been studied (Chu and Ghahramani, 2005) in the context of Gaussian process (GP) models by using a likelihood able to model preferential returns. Sim-

[1] Amazon Research Cambridge, UK [2] University of Sheffield, UK. Correspondence to: Javier González <gojav@amazon.com>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017. Copyright 2017 by the author(s).

[1] In this work we use $[\mathbf{x}, \mathbf{x}']$ to represent the vector resulting from concatenating both elements involved in the duel.

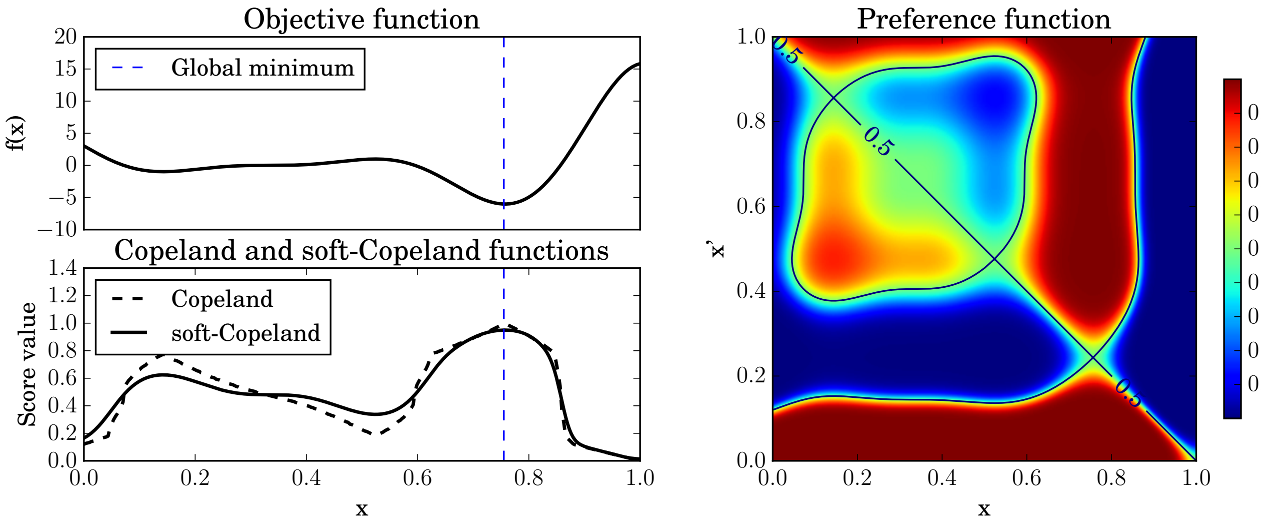

Figure 1. Illustration of the key elements of an optimization problem with pairwise preferential returns in a one-dimensional example. *Top-left*: Objective (Forrester) function to minimize. This function is only accessible through pairwise comparisons of inputs $\mathbf{x}$ and $\mathbf{x}'$. *Right*: true preference function $\pi_f([\mathbf{x}, \mathbf{x}'])$ that represents the probability that $\mathbf{x}$ will win a duel over $\mathbf{x}'$. Note that, by symmetry, $\pi_f([\mathbf{x}, \mathbf{x}']) = 1 - \pi_f([\mathbf{x}', \mathbf{x}])$. *Bottom left*: The normalised Copeland's and soft-Copeland function whose maximum is located at the same point of the minimum of f.

ilarly, Brochu (2010) used a probabilistic model to actively learn preferences in the context of discovering optimal parameters for simple graphics and animations engines. To select new duels, standard acquisition functions like the Expected Improvement (EI) (Mockus, 1977) are extended on top of a GP model with likelihood for duels. Although this approach is simple an effective in some cases, new duels are selected greedily, which may lead to over-exploitation.

In this work we propose a new approach aiming at combining the good properties of the arm-bandit methods with the advantages of having a probabilistic model able to capture correlations across the points in the domain $\mathcal{X}$. Following the above mentioned literature in the bandits settings, the key idea is to learn a preference function in the space of the duels by using a Gaussian process. This allows us to select the most relevant comparisons non-greedily and improve the state-of-the-art in this domain.

This paper is organized as follows. In Section 2 we introduce the point of view that it is followed in this work to model latent preferences. We define concepts such as the Copeland score function and the Condorcet's winner which form the basis of our approach. Also in Section 2, we show how to learn these objects from data. In Section 3 we generalize most commonly used acquisition functions to the dueling case. In Section 4 we illustrate the benefits of the proposed framework compared to state-of-the art methods in the literature. We conclude in Section 5 with a discussion and some future lines of research.

2. Learning latent preferences

The approach followed in this work is inspired by the work of (Ailon et al., 2014) in which cardinal bandits are reduced to ordinal ones. Similarly, here we focus on the idea of reducing the problem of finding the optimum of a latent function defined on $\mathcal{X}$ to determine a sequence of duels on $\mathcal{X} \times \mathcal{X}$.

We assume that each duel $[\mathbf{x}, \mathbf{x}']$ produces in a joint reward $f([\mathbf{x}, \mathbf{x}'])$ that is never directly observed. Instead, after each pair is proposed, the obtained feedback is a binary return $y \in \{0, 1\}$ representing which of the two locations is preferred. In this work, we assume that $f([\mathbf{x}, \mathbf{x}']) = g(\mathbf{x}') - g(\mathbf{x})$, but other alternatives are possible. Note that the more $\mathbf{x}$ is preferred over $\mathbf{x}'$ the bigger is the reward.

Since the preferences of humans are often unclear and may conflict, we model preferences as a stochastic process. In particular, the model of preference is a Bernoulli probability function

$$p(y = 1 | [\mathbf{x}, \mathbf{x}']) = \pi_f([\mathbf{x}, \mathbf{x}'])$$

and

$$p(y = 0 | [\mathbf{x}, \mathbf{x}']) = \pi_f([\mathbf{x}, \mathbf{x}'])$$

where $\pi : \Re \times \Re \to [0, 1]$ is an inverse link function. Via the latent loss, f maps each query $[\mathbf{x}, \mathbf{x}']$ to the probability of having a preference on the left input $\mathbf{x}$ over the right input $\mathbf{x}'$. The inverse link function has the property that $\pi_f([\mathbf{x}', \mathbf{x}]) = 1 - \pi_f([\mathbf{x}, \mathbf{x}'])$. A natural choice for π_f is the logistic function

$$\pi_f([\mathbf{x},\mathbf{x}']) = \sigma(f([\mathbf{x},\mathbf{x}'])) = \frac{1}{1+e^{-f([\mathbf{x},\mathbf{x}'])}}, \qquad (2)$$

but others are possible. Note that for any duel $[\mathbf{x},\mathbf{x}']$ in which $g(\mathbf{x}) \leq g(\mathbf{x}')$ it holds that $\pi_f([\mathbf{x},\mathbf{x}']) \geq 0.5$. π_f is therefore a *preference function* that fully specifies the problem.

We introduce here the concept of *normalised Copeland score*, already used in the literature of raking methods (Zoghi et al., 2015a), as

$$S(\mathbf{x}) = \text{Vol}(\mathcal{X})^{-1}\int_{\mathcal{X}} \mathbb{I}_{\{\pi_f([\mathbf{x},\mathbf{x}'])\geq 0.5\}} d\mathbf{x}',$$

where $\text{Vol}(\mathcal{X}) = \int_{\mathcal{X}} d\mathbf{x}'$ is a normalizing constant that bounds $S(\mathbf{x})$ in the interval $[0,1]$. If $\mathcal{X}$ is a finite set, the Copeland score is simply the proportion of duels that a certain element $\mathbf{x}$ will win with probability larger than 0.5. Instead of the Copeland score, in this work we use a soft version of it, in which the probability function π_f is integrated over $\mathcal{X}$ without further truncation. Formally, we define the soft-Copeland score as

$$C(\mathbf{x}) = \text{Vol}(\mathcal{X})^{-1}\int_{\mathcal{X}} \pi_f([\mathbf{x},\mathbf{x}'])d\mathbf{x}', \qquad (3)$$

which aims to capture the 'averaged' probability of $\mathbf{x}$ being the winner of a duel.

Following the armed-bandits literature, we say that $\mathbf{x}_c$ is a *Condorcet winner* if it is the point with maximal soft-Copeland score. It is straightforward to see that if $\mathbf{x}_c$ is a Condorcet winner with respect to the soft-Copeland score, it is a global minimum of f in $\mathcal{X}$: the integral in (3) takes maximum value for points $\mathbf{x} \in \mathcal{X}$ such that $f([\mathbf{x},\mathbf{x}']) = g(\mathbf{x}') - g(\mathbf{x}) > 0$ for all $\mathbf{x}'$, which only occurs if $\mathbf{x}_c$ is a minimum of f. This implies that if by observing the results of a set of duels we can learn the preference function π_f the optimization problem of finding the minimum of f can be addressed by finding the Condorcet winner of the Copeland score. See Figure 1 for an illustration of this property.

2.1. Learning the preference function $\pi_f([\mathbf{x},\mathbf{x}'])$ with Gaussian processes

Assume that N duels have been performed so far resulting in a dataset $\mathcal{D} = \{[\mathbf{x}_i,\mathbf{x}'_i], y_i\}_{i=1}^N$. Given $\mathcal{D}$, inference over the latent function f and its warped version π_f can be carried out by using Gaussian processes (GP) for classification (Rasmussen and Williams, 2005).

In a nutshell, a GP is a probability measure over functions such that any linear restriction is multivariate Gaussian. Any GP is fully determined by a positive definite covariance operator. In standard regression cases with Gaussian likelihoods, closed forms for the posterior mean and variance are available. In the preference learning, like the one we face here, the basic idea behind Gaussian process modeling is to place a GP prior over some latent function f that captures the membership of the data to the two classes and to squash it through the logistic function to obtain some prior probability π_f. In other words, the model for a GP for classification looks similar to eq. (2) but with the difference that f is an stochastic process as it is π_f. The stochastic latent function f is a *nuisance function* as we are not directly interested in its values but instead on particular values of π_f at test locations $[\mathbf{x}_\star,\mathbf{x}'_\star]$.

Inference is divided in two steps. First we need to compute the distribution of the latent variable corresponding to a test case, $p(f_\star|\mathcal{D},[\mathbf{x}_\star,\mathbf{x}'_\star],\theta)$ and later use this distribution over the latent $f_\star$ to produce a prediction

$$\begin{aligned}\pi_f([\mathbf{x}_\star,\mathbf{x}'_\star];\mathcal{D},\theta) &= p(y_\star = 1|\mathcal{D},[\mathbf{x},\mathbf{x}'],\theta) \qquad (4)\\ &= \int \sigma(f_\star)p(f_\star|\mathcal{D},[\mathbf{x}_\star,\mathbf{x}'_\star],\theta)df_\star\end{aligned}$$

where the vector θ contains the hyper-parameters of the model that can also be marginalized out. In this scenario, GP predictions are not straightforward (in contrast to the regression case), since the posterior distribution is analytically intractable and approximations at required (see (Rasmussen and Williams, 2005) for details). The important message here is, however, that given data from the locations and result of the duels we can learn the preference function π_f by taking into account the correlations across the duels, which makes the approach to be very data efficient compared to bandits scenarios where correlations are ignored.

2.2. Computing the soft-Copeland score and the Condorcet winner

The soft-Copeland function can be obtained by integrating $\pi_f([\mathbf{x},\mathbf{x}'])$ over $\mathcal{X}$, so it is possible to learn the soft-Copeland function from data by integrating $\pi_f([\mathbf{x},\mathbf{x}'],\mathcal{D})$. Unfortunately, a closed form solution for

$$\text{Vol}(\mathcal{X})^{-1}\int_{\mathcal{X}} \pi_f([\mathbf{x},\mathbf{x}'];\mathcal{D},\theta)d\mathbf{x}'$$

does not necessarily exist. In this work we use Monte-Carlo integration to approximate the Copeland score at any $\mathbf{x}\in\mathcal{X}$ via

$$C(\mathbf{x};\mathcal{D},\theta) \approx \frac{1}{M}\sum_{k=1}^{M}\pi_f([\mathbf{x},\mathbf{x}_k]);\mathcal{D},\theta), \qquad (5)$$

where $\mathbf{x}_1,\ldots,\mathbf{x}_M$ are a set of landmark points to perform the integration. For simplicity, in this work we select the landmark points uniformly, although more sophisticated probabilistic approaches can be applied (Briol et al., 2015).

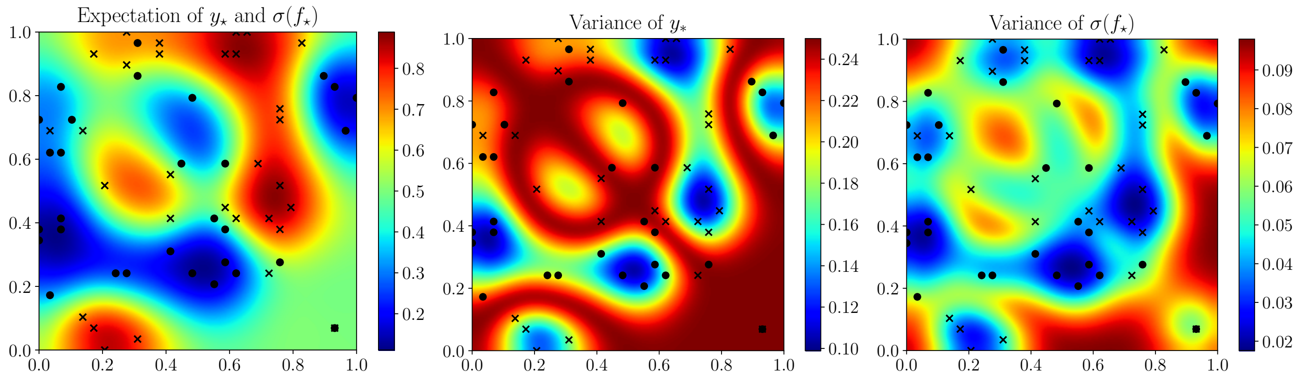

Figure 2. Differences between the sources of uncertainty that can be used for the exploration of the duels. The three figures show different elements of a GP used for preferential learning in the context of the optimization of the (latent) Forrester function. The model is learned using the result of 30 duels. *Left*: Expectation of $y_\star$, which coincides with the expectation of $\sigma(f_\star)$ and that is denoted as $\pi_f([\mathbf{x}_\star, \mathbf{x}'_\star]$. *Center*: Variance of output of the duels $y_\star$, that is computed as $\pi_f([\mathbf{x}_\star, \mathbf{x}'_\star](1 - \pi_f([\mathbf{x}_\star, \mathbf{x}'_\star])$. Note that the variance does not necessarily decrease in locations where observations are available. *Right*: Variance of the latent function $\sigma(f_\star)$. The variance of $\sigma(f_\star)$ decreases in regions where data are available, which make it appropriate for duels exploration contrast to the variance of $y_\star$.

The Condorcet winner can be computed by taking

$$x_c = \arg\max_{\mathbf{x}\in\mathcal{X}} C(\mathbf{x}; \mathcal{D}, \theta),$$

which can be done using a standard numerical optimizer. x_c is the point that has, on average, the maximum probability of wining most of the duels (given the data set $\mathcal{D}$) and therefore it is the most likely point to be the optimum of g.

3. Sequential Learning of the Condorcet winner

In this section we analyze the case in which n extra duels can be carried out to augment the dataset $\mathcal{D}$ before we have to report a solution to (1). This is similar to the set-up in (Brochu, 2010) where *interactive* Bayesian optimization is proposed by allowing a human user to sequentially decide the result of a number of duels.

In the sequel, we will denote by $\mathcal{D}_j$ the data set resulting of augmenting $\mathcal{D}$ with j new pairwise comparisons. Our goal in this section is to define a sequential policy for querying duels: $\alpha([\mathbf{x}, \mathbf{x}']; \mathcal{D}_j, \theta)$. This policy will enable us to identify as soon as possible the minimum of the the latent function g. Note that here, differently to the situation in standard Bayesian optimization, the search space of the acquisition, $\mathcal{X} \times \mathcal{X}$ is not the same as domain $\mathcal{X}$ of the latent function that we are optimizing. Our best guess about its optimum, however, is the location of the Condorcet's winner.

We approach the problem by proposing three dueling acquisition functions: (i) pure exploration (PE), the Copeland Expected improvement (CEI) and duelling-Thompson sampling, which makes explicitly use of the generative capabilities of our model. We analyze the three approaches in terms of what the balance *exploration-exploitation* means in our context. For simplicity in the notation, in the sequel we drop the dependency of all quantities on the parameters θ.

3.1. Pure Exploration

The first question that arises when defining a new acquisition for duels, is what *exploration* means in this context. Given a model as described in Section 2.1, the output variables $y_\star$ follow a Bernoulli distribution with probability given by the preference function π_f. A straightforward interpretation of pure exploration would be to search for the duel of which the outcome is most uncertain (has the highest variance of $y_\star$). The variance of $y_\star$ is given by

$$\mathbb{V}[y_\star|[\mathbf{x}_\star, \mathbf{x}'_\star], \mathcal{D}_j] = \pi_f([\mathbf{x}_\star, \mathbf{x}'_\star]; \mathcal{D}_j)(1 - \pi_f([\mathbf{x}_\star, \mathbf{x}'_\star]; \mathcal{D}_j)).$$

However, as preferences are modeled with a Bernoulli model, the variance of $y_\star$ does not necessarily reduce with sufficient observations. For example, according to eq. (2), for any two values $\mathbf{x}_\star$ and $\mathbf{x}'_\star$ such that $g(\mathbf{x}_\star) \approx g(\mathbf{x}'_\star)$, $\pi_f([\mathbf{x}_\star, \mathbf{x}'_\star], \mathcal{D}_j)$ will tend to be close to 0.5, and therefore it will have maximal variance even if we have already collected several observations in that region of the duels space.

Alternatively, exploration can be carried out by searching for the duel where GP is most uncertain about the probability of the outcome (has the highest variance of $\sigma(f_\star)$), which is the result of transforming out epistemic uncertainty about f, modeled by a GP, through the logistic function. The first order moment of this distribution coincides with the expectation of $y_\star$ but its variance is

$$\begin{aligned}\mathbb{V}[\sigma(f_\star)] &= \int \left(\sigma(f_\star) - \mathbb{E}[\sigma(f_\star)]\right)^2 p(f_\star|\mathcal{D}, [\mathbf{x}, \mathbf{x}'])df_\star \\ &= \int \sigma(f_\star)^2 p(f_\star|\mathcal{D}, [\mathbf{x}, \mathbf{x}'])df_\star - \mathbb{E}[\sigma(f_\star)]^2,\end{aligned}$$

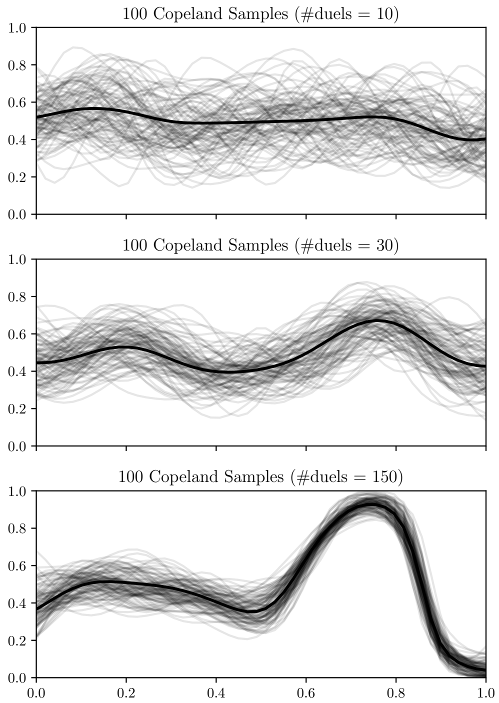

Figure 3. 100 continuous samples of the Copeland score function (grey) in the Forrester example generated using Thompson sampling. The three plots show the samples obtained once the model has been trained with different number of duels (10, 30 and 150). In black we show the Coplenad function computed using the preference function. The more samples are available more exploitation is encouraged in the first element of the duel as the probability of selecting $\mathbf{x}_{next}$ as the true optimum increases.

which explicitly takes into account the uncertainty over f. Hence, pure exploration of duels space can be carried out by maximizing

$$\alpha_{\text{PE}}([\mathbf{x},\mathbf{x}']|\mathcal{D}_j) = \mathbb{V}[\sigma(f_\star)|[\mathbf{x}_\star,\mathbf{x}'_\star]|\mathcal{D}_j].$$

Remark that in this case, duels that have been already visited will have a lower chance of being visited again even in cases in which the objective takes similar values in both players. See Figure 2 for an illustration of this property.

In practice, this acquisition requires to compute and intractable integral, that we approximate in practice using Monte-Carlo.

3.2. Copeland Expected Improvement

An alternative way to define an acquisition function is by generalizing the idea of the Expected Improvement (Mockus, 1977). The idea of the EI is to compute, in expectation, the marginal gain with respect to the current best observed output. In our context, as we do not have direct access to the objective, our only way of evaluating the quality of a single point is by computing its Copeland score. To generalize the idea to our context we need to find a couple of duels able to maximally improve the expected score of the Condorcet winner.

Denote by $c_j^\star$ the value of the Condorcet's winner when j duels have been already run. For any new proposed duel $[\mathbf{x},\mathbf{x}']$, two outcomes $\{0,1\}$ are possible that correspond to cases wether $\mathbf{x}$ or $\mathbf{x}'$ wins the duel. We denote by the $c_{\mathbf{x},j}^\star$ the value of the estimated Condorcet winner resulting of augmenting $\mathcal{D}$ with $\{[\mathbf{x},\mathbf{x}'],1\}$ and by $c_{\mathbf{x}',j}^\star$ the equivalent value but augmenting the dataset with $\{[\mathbf{x},\mathbf{x}'],0\}$. We define the one-lookahead Copeland Expected Improvement at iteration j as:

$$\alpha_{CEI}([\mathbf{x},\mathbf{x}']|\mathcal{D}_j) = \mathbb{E}\left[(0, c - c_j^\star)_+|\mathcal{D}_j\right] \qquad (6)$$

where $(\cdot)_+ = \max(0,\cdot)$ and the expectation is take over c, the value at the Condorcet winner given the result of the duel. The next duel is selected as the pair that maximizes the CEI. Intuitively, the CEI evaluated at $[\mathbf{x},\mathbf{x}']$ is a weighted sum of the total increase of the best possible value of the Copeland score in the two possible outcomes of the duel. The weights are chosen to be the probability of the two outcomes, which are given by π_f. The CEI can be computed in closed form as

$$\begin{aligned}\alpha_{CEI}([\mathbf{x},\mathbf{x}']|\mathcal{D}_j) &= \pi_f([\mathbf{x},\mathbf{x}']|\mathcal{D}_j)(c_{\mathbf{x},j}^\star - c_j^\star)_+ \\ &+ \pi_f([\mathbf{x}',\mathbf{x}]|\mathcal{D}_j)(c_{\mathbf{x}',j}^\star - c_j^\star)_+\end{aligned}$$

The computation of this acquisition is computationally demanding as it requires updating of the GP classification model for every fantasized output at any point in the domain. As we show in the experimental section, and similarly with what is observed in the literature about the EI, this acquisition tends to be greedy in certain scenarios leading to over exploitation (Hernández-Lobato et al., 2014; Hennig and Schuler, 2012). Although non-myopic generalizations of this acquisition are possible to address this issue in the same fashion as in (González et al., 2016b) these are be intractable.

3.3. Dueling-Thompson sampling

As we have previously detailed, pure explorative approaches that do not exploit the available knowledge about the current best location and CEI is expensive to compute and tends to over-exploit. In this section we propose an alternative acquisition function that is fast to compute and explicitly balances *exploration* and *exploitation*. We call this acquisition dueling-Thompson sampling and it is inspired by Thompson sampling approaches. It follows a two-step policy for duels:

1. *Step 1, selecting* $\mathbf{x}$*:* First, generate a sample $\tilde{f}$ from the model using continuous Thompson sampling [2] and

[2] Approximated continuous samples from a GP with shift-

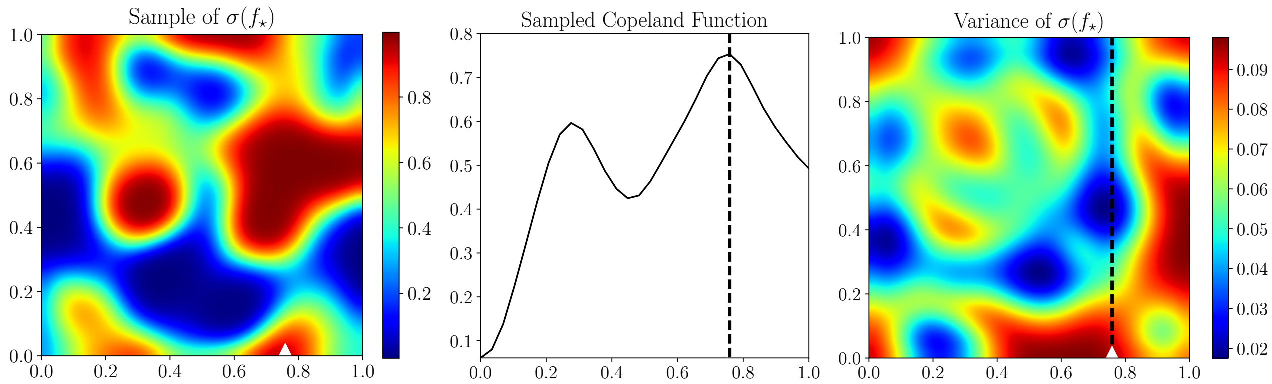

Figure 4. Illustration of the steps to propose a new duel using the duelling-Thompson acquisition. The duel is computed using the same model as in Figure 2. The white triangle represents the final selected duel. *Left:* Sample from $f_\star$ squashed through the logistic function σ. *Center:* Sampled soft-Copeland function, which results from integrating the the sample from $\sigma(f_\star)$ on the left across the vertical axis. The first element of the duel $\mathbf{x}$ is selected as the location of the maximum of the sampled soft-Copeland function (vertical dotted line). *Right:* The second element of the duel, $\mathbf{x}'$, is selected by maximizing the variance of $\sigma(f_\star)$ marginally given $\mathbf{x}$ (maximum across the vertical dotted line).

compute the associated soft-Copland's score by integrating over $\mathcal{X}$. The first element of the new duel, $\mathbf{x}_{next}$, is selected as:

$$\mathbf{x}_{next} = \arg\max_{\mathbf{x}\in\mathcal{X}} \int_{\mathcal{X}} \pi_{\tilde{f}}([\mathbf{x},\mathbf{x}'];\mathcal{D}_j)d\mathbf{x}'.$$

The term $\text{Vol}(\mathcal{X})^{-1}$ in the Copeland score has been dropped here as it does not change the location of the optimum. The goal of this step is to balance exploration and exploitation in the selection of the Condorcet winner, it is the same fashion Thompson sampling does: it is likely to select a point close to the current Condorcet winner but the policy also allows exploration of other locations as we base our decision on a stochastic $\tilde{f}$. Also, the more evaluations are collected, the more greedy the selection will be towards the Condorcet winner. See Figure 3 for an illustration of this effect.

2. *Step 2, selecting* $\mathbf{x}'$*:* Given $\mathbf{x}_{next}$ the second element of the duel is selected as the location that maximizes the variance of $\sigma(f_\star)$ in the direction of $\mathbf{x}_{next}$. More formally, $\mathbf{x}'_{next}$ is selected as

$$\mathbf{x}'_{next} = \arg\max_{\mathbf{x}'_\star\in\mathcal{X}} \mathbb{V}[\sigma(f_\star)|[\mathbf{x}_\star,\mathbf{x}'_\star],\mathcal{D}_j,\mathbf{x}_\star=\mathbf{x}_{next}]$$

This second step is purely explorative in the direction of x_{new} and its goal is to find informative comparisons to run with current good locations identified in the previous step.

Algorithm 1 The PBO algorithm.

```
Input: Dataset $\mathcal{D}_0 = \{[\mathbf{x}_i,\mathbf{x}'_i],y_i\}_{i=1}^N$ and number of remaining evaluations $n$, acquisition for duels $\alpha([\mathbf{x},\mathbf{x}'])$.
for $j = 0$ to $n$ do
    1. Fit a GP with kernel $k$ to $\mathcal{D}_j$ and learn $\pi_{f,j}(\mathbf{x})$.
    2. Compute the acquisition for duels $\alpha$.
    3. Next duel: $[\mathbf{x}_{j+1},\mathbf{x}'_{j+1}] = \arg\max\alpha([\mathbf{x},\mathbf{x}'])$.
    4. Run the duel $[\mathbf{x}_{j+1},\mathbf{x}'_{j+1}]$ and obtain $y_{j+1}$.
    5. Augment $\mathcal{D}_{j+1} = \{\mathcal{D}_j \cup ([\mathbf{x}_{j+1},\mathbf{x}'_{j+1}],y_{j+1})\}$.
end for
Fit a GP with kernel $k$ to $\mathcal{D}_n$.
Returns: Report the current Condorcet's winner $\mathbf{x}_n^\star$.
```

In summary the dueling-Thompson sampling approach selects the next duel as:

$$\arg\max_{[\mathbf{x},\mathbf{x}']} \alpha_{DTS}([\mathbf{x},\mathbf{x}']|\mathcal{D}_j) = [\mathbf{x}_{next},\mathbf{x}'_{next}]$$

where $\mathbf{x}_{next}$ and $\mathbf{x}_{next}$ are defined above. This policy combines a selection of a point with high chances of being the optimum with a point whose result of the duel is uncertain with respect of the previous one. Note that this has some interesting connections with uncertain sampling as presented in (Houlsby et al., 2011). See Figure 4 for a visual illustration of the two steps in toy example. See Algorithm 1 for a full description of the PBO approach.

3.4. Generalizations to multiple returns scenarios

A natural extension of the PBO set-up detailed above are cases in which multiple comparisons of inputs are simultaneously allowed. This is equivalent to providing a ranking over a set of points $\mathbf{x}_1,\ldots,\mathbf{x}_k$. Rankings are trivial to map to pairwise preferences by using the pairwise ordering to

invariant kernel can be obtained by using Bochner's theorem (Bochner et al., 1959). In a nutshell, the idea is to approximate the kernel by means of the inner product of a finite number Fourier features that are sampled according to the measure associated to the kernel (Rahimi and Recht, 2008; Hernández-Lobato et al., 2014).

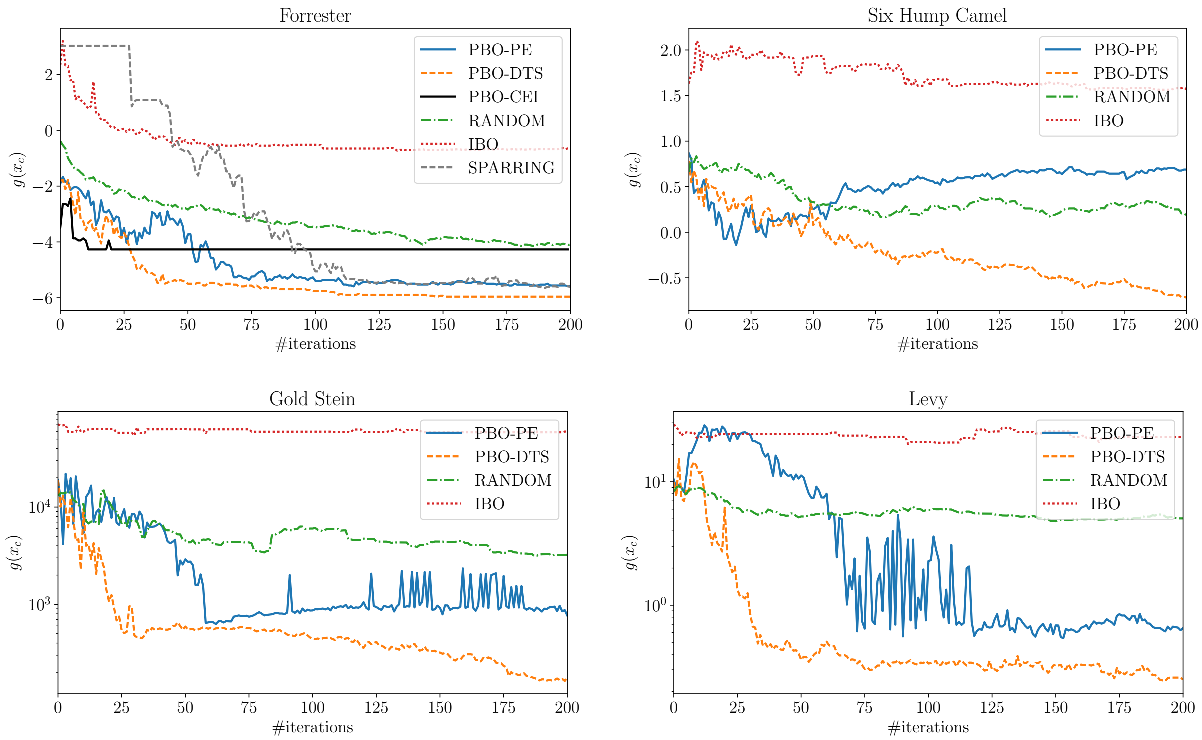

Figure 5. Averaged results across 4 latent objective functions and 6 different methods. The CEI is only computed in the Forrester function as it is intractable when the dimension increases. Results are computed over 20 replicates in which 5 randomly selected initial points are used to initialize the models and that are kept the same across the six methods. The horizontal axis of the plots represents the number of evaluation and the vertical axis represent the value (log scale in the second row) of the true objective function at the best guess (Condorcet winner) at each evaluation. Note that this is only possible as we know the true objective function.The curves are not necessarily monotonically decreasing as we do not show the current best solution but the current solution at each iteration (proposed location by each method at each step in a real scenario).

obtain the result of the duels. The problem is, therefore, equivalent from a modeling perspective. However, from the point of view of selecting the k locations to rank in each iteration, generalization of the above mentioned acquisitions are required. Although this is not the goal of this work, it is interesting to remark that this problem has strong connections with the one of computing batches in BO (González et al., 2016a).

4. Experiments

We present three experiments which validate our approach in terms of performance and illustrate its key properties. The set-up is as follows: we have a non-linear black-box function $g(\mathbf{x})$ of which we look for its minimum as described in equation (1). However, we can only query this function through pairwise comparisons. The outcome of a pairwise comparison is generated as described in Section 2, i.e., the outcome is drawn from a Bernoulli distribution of which the sample probability is computed according to equation (2).

We have considered for $g(\mathbf{x})$ the Forrester, the 'six-hump camel', 'Gold-Stein' and 'Levy' as latent objective functions to optimize. The Forrester function is 1-dimensional, whereas the rest are defined in domains of dimension 2. The explicit formulation of these objectives and the domains in which they are optimized are available as part of standard optimization benchmarks[3]. The PBO framework is applicable in the continuous setting. In this section, however, the search of the optimum of the objectives is performed in a grid of size (33 per dimension for all cases), which has practical advantages: the integral in eq. (5) can easily be treated as a sum and, more importantly, we can compare PBO with bandit methods that are only defined in discrete domains. Each comparison starts with 5 initial (randomly selected) duels and a total budget of 200 duels are run, after which, the best location of the optimum should be reported. Further, each algorithm runs for 20 times (trials) with different initial duels (the same for all methods) [4] [5]. We report the average performance across all trials, which is defined as the value of g (known in all cases) evaluated at the current

[3] https://www.sfu.ca/ssurjano/optimisation.html

[4] RAMDOM runs for 100 trials to give a reliable curve

[5] PBO-CEI only runs 5 trials for Forrester as it is very slow.

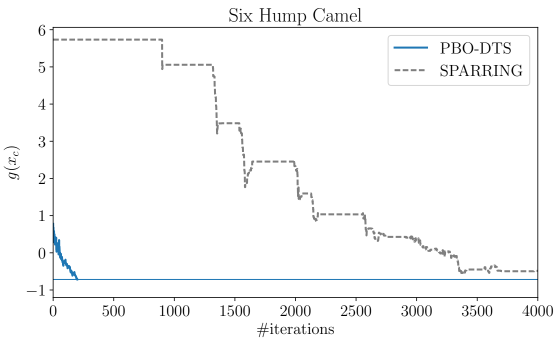

Figure 6. Comparison of the bandits Sparring algorithm and PBO-DTS on the Six-Hump Camel for an horizon in which the bandit method is run for 4000 iterations. The horizontal line represents the solution proposed by PBO-DTS after 200 duels. As the plot illustrates, modeling correlations across the 900 arms (points in 2D) with a GP drastically improves the speed at which the optimum of the function is found. Note that the bandit method needs to visit at least each arm before starting to get any improvement while PBO-DTS is able to make a good (initial) guess with the first 5 random duels used in the experiment. Similar results are obtained for the rest of functions.

Condorcet winner, $\mathbf{x}_c$ considered to be the best by each algorithm at each one of the 200 steps taken until the end of the budget. Note that each algorithm chooses $\mathbf{x}_c$ differently, which leads to different performance at step 0. Therefore, we present plots of #iterations versus $g(\mathbf{x}_c)$.

We compare 6 methods. Firstly, the three variants within the PBO framework: PBO with pure exploration (PBO-PE, see section 3.1), PBO with the Copeland Expected Improvement (PBO-CEI, see section 3.2) and PBO with dueling Thomson sampling (PBO-DTS, see section 3.3). We also compare against a random policy where duels are drawn from a uniform distribution (RAMDOM) [6] and with the interactive Bayesian optimization (IBO) method of (Brochu, 2010). IBO selects duels by using an extension of Expected Improvement on a GP model that encodes the information of the preferential returns in the likelihood. Finally, we compared against all three cardinal bandit algorithms proposed by Ailon et al. (2014), namely *Doubler*, *MultiSBM* and *Sparring*. Ailon et al. (2014) observes that *Sparring* has the best performance, and it also outperforms the other two bandit-based algorithms in our experiments. Therefore, we only report the performance for *Sparring* to keep the plots clean. In a nutshell, the *Sparring* considers two bandit players (agents), one for each element of the duel, which use the Upper Confidence Bound criterion and where the input grid is acting as the set of arms. The decision for which pair of arms to query is according to the strategies and beliefs of each agent. In this case, correlations in f are not captured.

Figure 5 shows the performance of the compared methods, which is consistent across the four plots: IBO shows a poor performance, due to the combination of the greedy nature of the acquisitions and the poor calibration of the model. The RAMDOM policy converges to a sub-optimal result and the PBO approaches tend to be the superior ones. In particular, we observe that PBO-DTS is consistently proven as the best policy, and it is able to balance correctly exploration and exploitation in the duels space. Contrarily, PBO-CEI, which is only used in the first experiment due to the excessive computational overload, tends to over exploit. PBO-PE obtains reasonable results but tends to work worse in larger dimensions where is harder to cover the space.

Regarding the bandits methods, they need a much larger number of evaluations to converge compared to methods that model correlations across the arms. They are also heavily affected by an increase of the dimension (number of arms). The results of the *Sparring* method are shown for the Forrester function but are omitted in the rest of the plots (the number of used evaluations used is smaller than the numbers of the arms and therefore no real learning can happen within the budget). However, in Figure 6 we show the comparison between *Sparring* and PBO-DTS for an horizon in which the bandit method has almost converged. The gain obtained by modeling correlations among the duels is evident.

5. Conclusions

We have explored a new framework, PBO, for optimizing black-box functions in which only preferential returns are available. The fundamental idea is to model comparisons of pairs of points with a Gaussian, which leads to the definition of new policies for augmenting the available dataset. We have proposed three acquisitions for duels, PE, CEI and DTS, and explored their connections with existing policies in standard BO. Via simulation, we have demonstrated the superior performance of DTS, both because it finds a good balance between exploration and exploitation in the duels space and because it is computationally tractable. In comparison with other alternatives out of the PBO framework, such as IBO or other bandit methods, DTS shows the state-of-the-art performance. There exist several future extensions of our current approach like tackling the existing limitation on the dimension of the input space, which is doubled with respect to the original dimensionality of the problem. Also further theoretical analysis will be carried out on the proposed acquisitions.

Acknowledgements

The authors would like to thank Mike Croucher for the inspirational ideas that motivated the development of this work during the Gaussian process summer school held in Sheffield in September 2015.

[6] $\mathbf{x}_c$ is chosen as the location that wins most frequently.

References

Nir Ailon, Zohar Shay Karnin, and Thorsten Joachims. Reducing dueling bandits to cardinal bandits. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pages 856–864, 2014.

Salomon Bochner, Monotonic Functions, Stieltjes Integrals, Harmonic Analysis, Morris Tenenbaum, and Harry Pollard. *Lectures on Fourier Integrals. (AM-42)*. Princeton University Press, 1959. ISBN 9780691079943.

François-Xavier Briol, Chris J. Oates, Mark Girolami, and Michael A. Osborne. Frank-Wolfe Bayesian quadrature: Probabilistic integration with theoretical guarantees. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, NIPS'15, pages 1162–1170, Cambridge, MA, USA, 2015. MIT Press.

Eric Brochu. *Interactive Bayesian Optimization: Learning Parameters for Graphics and Animation*. PhD thesis, University of British Columbia, Vancouver, Canada, December 2010.

Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl, editors. *The Adaptive Web: Methods and Strategies of Web Personalization*. Springer-Verlag, Berlin, Heidelberg, 2007. ISBN 978-3-540-72078-2.

Wei Chu and Zoubin Ghahramani. Preference learning with Gaussian processes. In *Proceedings of the 22Nd International Conference on Machine Learning*, ICML '05, pages 137–144, New York, NY, USA, 2005. ACM. ISBN 1-59593-180-5.

Miroslav Dudík, Katja Hofmann, Robert E. Schapire, Aleksandrs Slivkins, and Masrour Zoghi. Contextual dueling bandits. In *Proceedings of The 28th Conference on Learning Theory, COLT 2015, Paris, France, July 3-6, 2015*, pages 563–587, 2015.

Javier González, Zhenwen Dai, Philipp Hennig, and N. Lawrence. Batch bayesian optimization via local penalization. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics (AISTATS 2016)*, volume 51 of *JMLR Workshop and Conference Proceedings*, pages 648–657, 2016a.

Javier González, Michael A. Osborne, and Neil D. Lawrence. GLASSES: relieving the myopia of bayesian optimisation. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, AISTATS 2016, Cadiz, Spain, May 9-11, 2016*, pages 790–799, 2016b.

Philipp Hennig and Christian J. Schuler. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13, 2012.

Ralf Herbrich, Tom Minka, and Thore Graepel. Trueskill$^{\mathrm{TM}}$: A bayesian skill rating system. In P. B. Schölkopf, J. C. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 569–576. MIT Press, 2007.

José Miguel Hernández-Lobato, Matthew W Hoffman, and Zoubin Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 918–926. Curran Associates, Inc., 2014.

Neil Houlsby, Ferenc Huszar, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *CoRR*, abs/1112.5745, 2011.

Kevin G. Jamieson, Sumeet Katariya, Atul Deshpande, and Robert D. Nowak. Sparse dueling bandits. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2015, San Diego, California, USA, May 9-12, 2015*, 2015.

Donald R. Jones. A taxonomy of global optimization methods based on response surfaces. *Journal of global optimization*, 21(4):345383, 2001.

Daniel Kahneman and Amos Tversky. Prospect theory: An analysis of decision under risk. *Econometrica*, 47(2): 263–91, 1979.

Jonas Mockus. On Bayesian methods for seeking the extremum and their application. In *IFIP Congress*, pages 195–200, 1977.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1177–1184. Curran Associates, Inc., 2008.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems 25*, pages 2951–2959, 12/2012 2012.

Balázs Szörényi, Róbert Busa-Fekete, Adil Paul, and Eyke Hüllermeier. Online rank elicitation for plackett-luce: A dueling bandits approach. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 604–612, 2015.

Huasen Wu, Xin Liu, and R. Srikant. Double Thompson sampling for dueling bandits. *CoRR*, abs/1604.07101, 2016.

Yisong Yue and Thorsten Joachims. Beat the mean bandit. In *ICML*, pages 241–248, 2011.

Yisong Yuea, Josef Broderb, Robert Kleinbergc, and Thorsten Joachims. The k-armed dueling bandits problem. *Journal of Computer and System Sciences*, 78(5): 1538 – 1556, 2012. ISSN 0022-0000. {JCSS} Special Issue: Cloud Computing 2011.

Masrour Zoghi, Shimon Whiteson, Remi Munos, and Maarten de Rijke. Relative upper confidence bound for the K-armed dueling bandit problem. In *ICML 2014: Proceedings of the Thirty-First International Conference on Machine Learning*, pages 10–18, June 2014.

Masrour Zoghi, Zohar S Karnin, Shimon Whiteson, and Maarten de Rijke. Copeland dueling bandits. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 307–315. Curran Associates, Inc., 2015a.

Masrour Zoghi, Zohar S. Karnin, Shimon Whiteson, and Maarten de Rijke. Copeland dueling bandits. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 307–315, 2015b.