PROCEEDINGS A

royalsocietypublishing.org/journal/rspa

Research

Cite this article: Yang Y, Blanchard A, Sapsis T, Perdikaris P. 2022 Output-weighted sampling for multi-armed bandits with extreme payoffs. *Proc. R. Soc. A* **478**: 20210781. https://doi.org/10.1098/rspa.2021.0781

Received: 6 October 2021
Accepted: 24 March 2022

Subject Areas:
computer modelling and simulation, artificial intelligence, computational mathematics

Keywords:
scientific machine learning, active learning, uncertainty quantification, Bayesian statistics

Author for correspondence:
Paris Perdikaris
e-mail: pgp@seas.upenn.edu

Electronic supplementary material is available online at https://doi.org/10.6084/m9.figshare.c.5953885.

THE ROYAL SOCIETY
PUBLISHING

Output-weighted sampling for multi-armed bandits with extreme payoffs

Yibo Yang[1], Antoine Blanchard[2], Themistoklis Sapsis[2] and Paris Perdikaris[1]

[1]Department of Mechanical Engineering, and Applied Mechanics, University of Pennsylvania, Philadelphia, PA 19104, USA
[2]Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

AB, 0000-0001-6514-0619; TS, 0000-0003-0302-0691; PP, 0000-0002-2816-3229

We present a new type of acquisition function for online decision-making in multi-armed and contextual bandit problems with extreme payoffs. Specifically, we model the payoff function as a Gaussian process and formulate a novel type of upper confidence bound acquisition function that guides exploration towards the bandits that are deemed most relevant according to the variability of the observed rewards. This is achieved by computing a tractable likelihood ratio that quantifies the importance of the output relative to the inputs and essentially acts as an *attention mechanism* that promotes exploration of extreme rewards. Our formulation is supported by asymptotic zero-regret guarantees, and its performance is demonstrated across several synthetic benchmarks, as well as two realistic examples involving noisy sensor network data. Finally, we provide a JAX library for efficient bandit optimization using Gaussian processes.

1. Introduction

Online decision-making defines an important branch of modern machine learning in which uncertainty quantification plays a prominent role. In most stochastic optimization settings, evaluating the unknown function is expensive, hence new information needs to be acquired judiciously.

Classical applications include recommendation systems for articles and products, where the goal is to maximize the total revenue of the product maker given limited user feedback [1,2]; control and reinforcement learning, where the reward is obtained after a sequence of experiments or actions and the objective is not only to obtain optimal rewards but also avoid the potentially negative effects of uncertainty [3,–6]; environment monitoring, where sensor data is used to identify areas of interest as in traffic flow estimation [7] and room temperature monitoring [8]; and optimal design of expensive experiments [9,10].

More recently, new applications have appeared beyond machine learning, including optimal sampling in cardiac electro-physiology and bio-engineering [11,12], multi-fidelity design of experiments [13,14], hyper-parameter tuning in high-dimensional design spaces [15–17] and prediction of extreme events in complex dynamical systems [18,19].

Many of these applications can be formulated as multi-armed bandit problems, for which effective sampling algorithms exist [4,7,8,20,22–]. These algorithms are generally characterized by two key ingredients. First, they involve building a model for the latent payoff function given scarce and possibly noisy observations of past rewards. To enable effective sampling and exploration of the decision space, uncertainty in the model predictions needs to be accounted for in the predictive posterior distribution of the latent payoffs, which can be obtained via either a frequentist or a Bayesian approach. The second critical ingredient pertains to designing a data acquisition policy that can leverage the model predictive uncertainty to effectively balance the trade-off between exploration and exploitation while ensuring a consistent asymptotic behaviour for the cumulative regret.

(a) Previous work

Multi-armed bandit problems provide a general setting for developing online decision-making algorithms and rigorously studying their performance. Early research in this setting includes the celebrated ϵ-greedy algorithm [22], where random exploration is introduced with a small probability ϵ to prevent the algorithm from focusing on local sub-optimal solutions. Despite its widespread applicability, ϵ-greedy algorithms employ a heuristic treatment of uncertainty, and often require careful tuning in order to prevent sub-optimal exploration.

To this end, the upper confidence bound (UCB) policy [20,23] was proposed to provide a natural way to estimate sub-optimal choices using a model's predictive posterior uncertainty. However, the original UCB formulation does not take into account correlations between different bandits in a multi-armed setting and, therefore, typically requires a large number of data points to be collected before convergence can be observed. Variants of the UCB algorithm have been adapted to the contextual bandit problems, where contextual information is available to each bandit as additional aid in designing a recommendation. A common setting involves problems with linear payoffs, where the payoff function is modelled via Bayesian linear regression [21]. Gaussian process models have also been employed to account for correlated payoffs, and the corresponding GP-UCB criteria have shown great promise in data-scarce and 'cold start' scenarios [7,8,24].

Thompson sampling [25,26] provides an alternative approach to balancing the exploration–exploitation trade-off that only requires access to posterior samples of a parametrized payoff function. Although the algorithm was largely ignored at the time of its inception by Thompson [25], the results of Chapelle & Li [27] have initiated a wave of resurgence, leading to significant advances in applications (e.g. recommendation systems [2], hyper-parameter optimization [28], reinforcement learning [3,–5]), as well as theoretical analyses (e.g. optimal regret bounds [29–31]). More recently, Bayesian deep learning models have been considered [32] for modelling more complex and high-dimensional payoff functions. However, their effectiveness, interpretability and convergence behaviour are still under investigation [33].

Here, we would also like to emphasize a critical difference between contextual bandits problems and Bayesian optimization or active learning. In the bandit setting, the goal is to minimize the cumulative regret while searching for the optimal bandit. By contrast, Bayesian optimization and active learning focus on the quality of the final state of a given objective, and therefore typically discard the intermediate states produced during the search process. As a common example of a contextual bandit problem, consider the task of selecting which news article to show first on the main page of your website in order to optimize the click-through rate. The context would reflect information about different users, e.g. where they come from, previously visited pages of the site, device information, geo-location, etc. An action is a choice of what news article to display, tailored to each individual user. An outcome is whether the user clicked on a link or not. The rewards in this case are binary: 0 if there is no click, 1 if there is a click. In the contextual bandit problem, a learning algorithm can be employed to design a policy that maximizes the cumulative rewards. In this process, the learner repeatedly observes a context, chooses an action, and observes a reward for the chosen action only. The ultimate goal is to choose actions in dynamic environments where the available options may change rapidly, while the cost of data acquisition is high, therefore necessitating a judicious sampling of the decision space.

(b) Our contributions

(i) Primary contribution

All aforementioned approaches have enjoyed success across various applications, however, they lack a mechanism for distinguishing and promoting the input/context variables that have the greatest influence on the observed payoffs. Short of such a mechanism, regions in the decision space that may have negligible effect on the payoffs will still be sampled as long as they are uncertain. As we will demonstrate, this undesirable behaviour can have a deteriorating impact on convergence, and this effect is exacerbated in the presence of extreme payoffs (i.e. situations in which a small number of bandits yield rewards significantly greater than the rest of the bandit population).

Motivated by the recent findings in [34–36], we introduce a novel UCB-type objective for online decision-making in multi-armed and contextual bandit problems that can overcome the aforementioned shortcomings. This is achieved by introducing an importance weight to effectively promote the exploration of ‘heavy-tailed’ (i.e. rare and extreme) payoffs. We show how such importance weight can be derived from a likelihood ratio that quantifies the relative importance between inputs/contexts and observed rewards, introducing an effective *attention mechanism* that favours exploration of bandits with unusually large rewards over bandits associated with frequent, average payoffs. Our formulation is supported by theoretical guarantees demonstrating that the proposed acquisition strategy will asymptotically yield a zero-regret policy. This output-weighted approach has been shown to outperform classical acquisition functions in active learning [37] and Bayesian optimization [36] tasks, and here we set sail for the first time into investigating its effectiveness in online decision-making tasks, with a specific focus on multi-armed and contextual bandit problems subject to extreme payoffs.

(ii) Comparison to previous work

We demonstrate the effectiveness of the proposed methodology across a collection of synthetic benchmarks, as well as two realistic examples involving noisy sensor network data (specifically, temperature and air quality measurements). In all cases, we provide comprehensive quantitative comparisons between the proposed output-weighted sampling criterion and the most widely used criteria in current practice, including the UCB [20], GP-UCB [7], Thompson sampling [25,27] and expected improvement [38] methods.

(iii) Secondary contributions

We have developed an open-source Python package for bandit optimization using Gaussian processes.[1] Our implementation leverages the high-performance package JAX [39] and thus enables (a) gradient-based optimization of the proposed output-weighted sampling criteria

[1] See https://github.com/PredictiveIntelligenceLab/jax-bandits.

for general Gaussian process priors, (b) the use of GPU acceleration and (c) scalability and parallelization across multiple computing nodes. This package can be readily used to reproduce all data and results presented in this paper.

2. Methods

(a) Multi-armed bandits

The multi-armed bandit problem is a prototypical paradigm for sequential decision-making. The decision set consists of a discrete collection of M arms where the ith arm may be associated with some contextual information $\mathbf{x}_i \in \mathbb{R}^d$. Pulling arm i produces a reward $y \in \mathbb{R}$ which is determined by some unknown latent function

$$y_t = f(\mathbf{x}_i) + \epsilon_t, \tag{2.1}$$

where $\epsilon_t \sim \mathcal{N}(0, \sigma_n^2)$ accounts for observation noise.

At each round t, we select an arm i and obtain a reward y_t. The goal of sequential decision-making is to find a strategy for bandit selection that maximizes the total reward $\sum_{t=1}^{T} y_t$ for a given budget T. In other words, the goal is to first identify the bandits that provide the best rewards,

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} f(\mathbf{x}), \tag{2.2}$$

using as few arm pulls as possible, and then to keep on exploiting these optimal bandits to maximize the total reward.

As an alternative metric of success, it is useful to consider the simple regret $r_t = f(\mathbf{x}^*) - f(\mathbf{x})$, as maximizing the total reward is essentially equivalent to minimizing the *cumulative* regret

$$R_T = \sum_{t=1}^{T} r_t. \tag{2.3}$$

The holy grail of online decision-making is to design an effective no-regret policy satisfying

$$\lim_{T\to\infty} \frac{R_T}{T} = 0. \tag{2.4}$$

(b) Gaussian processes

Gaussian process (GP) regression provides a flexible probabilistic framework for modelling nonlinear black-box functions [40]. Given a dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ of input–output pairs (i.e, context–reward pairs), and an observation model of the form $y = f(\mathbf{x}) + \epsilon$, the goal is to infer the latent function f as well as the unknown noise variance σ_n^2 corrupting the observations.

In GP regression, no assumption is made on the form of the latent function f to be learned; rather, a prior probability measure is assigned to every function in the function space. Starting from a zero-mean Gaussian prior assumption on f,

$$f(\mathbf{x}) \sim \mathrm{GP}(\mathbf{0}, k(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta})), \tag{2.5}$$

the goal is to identify an optimal set of hyper-parameters $\boldsymbol{\Theta} = \{\boldsymbol{\theta}, \sigma_n^2\}$, and then use the optimized model to predict the rewards of unseen bandits. The covariance function $k(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta})$ plays a key role in this procedure as it encodes prior belief or domain expertise one may have about the underlying function f. In the absence of any domain-specific knowledge, it is common to assume that f is a smooth continuous function and employ the squared exponential covariance kernel with automatic relevance determination (ARD) which accounts for anisotropy with respect to each input variable [40].

Unlike previous works [7,8], we do not assume that the payoff function f actually comes from a GP prior or that it has low RKHS norm. Instead, we compute an optimal set of hyper-parameters at each round t by minimizing the negative log-marginal likelihood of the GP model [40]. In our setup, the likelihood is Gaussian and can be computed analytically as

$$\mathcal{L}(\boldsymbol{\Theta}) = \frac{1}{2}\log|\mathbf{K} + \sigma_n^2\mathbf{I}| + \frac{1}{2}\mathbf{y}^{\mathsf{T}}(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y} + \frac{N}{2}\log(2\pi), \tag{2.6}$$

where $\mathbf{K}$ is an $N \times N$ covariance matrix constructed by evaluating the kernel function on the input training data $\mathbf{X}$. The minimization problem is solved with an L-BFGS optimizer with random restarts [41].

Once the GP model has been trained, the predictive distribution at any given bandit $\mathbf{x}$ can be computed by conditioning on the observed data

$$p(\mathbf{y} \mid \mathbf{x}, \mathcal{D}) \sim \mathcal{N}(\mu(\mathbf{x}), \sigma^2(\mathbf{x})), \tag{2.7}$$

where

$$\mu(\mathbf{x}) = k(\mathbf{x}, \mathbf{X})(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y} \tag{2.8a}$$

and

$$\sigma^2(\mathbf{x}) = k(\mathbf{x}, \mathbf{x}) - k(\mathbf{x}, \mathbf{X})(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}k(\mathbf{X}, \mathbf{x}). \tag{2.8b}$$

Here, $\mu(\mathbf{x})$ can be used to make predictions of the underlying function $f(\mathbf{x})$ and $\sigma^2(\mathbf{x})$ to quantify the associated uncertainty. Note that $\mu(\mathbf{x})$ and $\sigma^2(\mathbf{x})$ are implicitly conditioned on $\mathcal{D}$, but here we will omit this dependency to simplify our notation. Also, in all examples studied in this manuscript, we assume that the noise process that corrupts the observed rewards has a constant variance σ_n^2 that does not depend on the context variables $\mathbf{x}$. As such, the posterior mean $\mu(\mathbf{x})$ function can be understood as our best approximation for the latent function $f(\mathbf{x})$ that generated the rewards y observed by our model during training (as described in equation (2.1)).

(c) Online decision-making

A critical ingredient in online decision-making is the choice of the *acquisition function*, which effectively determines which bandits the algorithm should try out and which ones to ignore [7,8]. A popular choice of acquisition function is the 'vanilla' upper confidence bound (V-UCB),

$$a_{\text{V-UCB}}(\mathbf{x}) = \mu(\mathbf{x}) + \kappa\sigma(\mathbf{x}), \tag{2.9}$$

and the closely related GP-UCB criterion [7],

$$a_{\text{GP-UCB}}(\mathbf{x}) = \mu(\mathbf{x}) + \beta_t^{1/2}\sigma(\mathbf{x}), \tag{2.10}$$

where κ and $\beta_t = 2\log(|D|t^2\pi^2/(6\delta))$ are parameters that aim to balance exploration and exploitation (see [7] for more details). Higher values of these parameters lead to stronger exploration while smaller values place more emphasis on exploitation. Here, $|D|$ is the number of bandits in the absence of context, and the dimension of the context otherwise. In V-UCB, κ is typically considered constant, while in GP-UCB, β_t depends on the round t and comes with convergence guarantees when the payoff function is not too complex [7].

In this work, we also consider the expected improvement, whose convergence properties have been well studied [38],

$$a_{\text{EI}}(\mathbf{x}) = \sigma(\mathbf{x})[\lambda(\mathbf{x})\Phi(\lambda(\mathbf{x})) + \phi(\lambda(\mathbf{x}))], \tag{2.11}$$

where $\Phi(\cdot)$ and $\phi(\cdot)$ are the cumulative distribution function and probability density function of standard normal distribution, respectively. In (2.11), we have defined $\lambda(\mathbf{x}) = (\mu(\mathbf{x}) - y^* - \xi)/\sigma(\mathbf{x})$, with y^* the best reward recorded so far and ξ a user-specified parameter controlling the exploration–exploitation trade-off. Higher ξ values lead to more exploration. The quantity $\tilde{y}(\mathbf{x})$ in (2.12) denotes a random sample drawn from the posterior distribution of the GP model, that is, $\tilde{y}(\mathbf{x}) \sim \mathcal{N}(\mu(\mathbf{x}), \sigma^2(\mathbf{x}))$. Finally, another commonly used acquisition function for bandit problems is Thompson sampling,

$$a_{\text{TS}}(\mathbf{x}) = \tilde{y}(\mathbf{x}), \tag{2.12}$$

also known to deliver competitive results in practice [27,33,42].

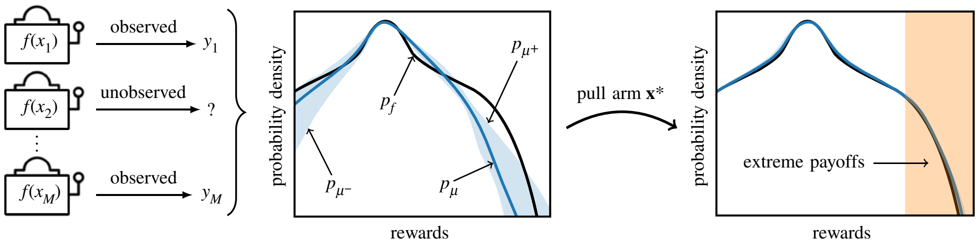


Figure 1. Sketch of the acquisition scheme from which the likelihood ratio is derived. The best next bandit $\mathbf{x}^*$ maximizes the reduction of the uncertainty in the tails of the payoff distribution (quantified by the log-difference between p_{μ^+} and p_{μ^-}, the respective densities of $\mu^{\pm} = \mu \pm \sigma^2$). (Online version in colour.)

The goal in bandit optimization is to determine the best bandit to sample next by maximizing the acquisition function

$$\mathbf{x}_{t+1} = \arg\max_{\mathbf{x}} a(\mathbf{x}|\mathcal{D}), \tag{2.13}$$

where a can be any of (2.9), (2.10), (2.11) or (2.12), and $\mathcal{D}$ contains all the observed context–reward pairs up to round t.

(d) Output-weighted sampling

Blanchard & Sapsis [36] recently introduced an efficient and minimally intrusive approach for accelerating the stochastic optimization process in cases where certain regions of the input space have a considerably larger impact on the output of the latent function than others (i.e. extreme payoffs in the bandit problem) by incorporating a sampling weight into several of the acquisition functions commonly used in practice. The sampling weight, referred to as the ‘likelihood ratio’, was derived from a heavy-tail argument whereby the best next input point to visit is selected so as to most reduce the uncertainty in the *tails* of the output statistics where the extreme payoffs reside (figure 1).

The likelihood ratio is defined as

$$w(\mathbf{x}) = \frac{p_{\mathbf{x}}(\mathbf{x})}{p_\mu(\mu(\mathbf{x}))}, \tag{2.14}$$

and was derived in [36]. Here, $p_{\mathbf{x}}(\mathbf{x})$ is a prior distribution that can be used to distill prior beliefs about the importance of each bandit or environmental conditions. In this work, we assume that no such prior information is available and treat every bandit equally by specifying a uniform prior, $p_{\mathbf{x}}(\mathbf{x}) = 1$ for all $\mathbf{x}$. The term $p_\mu(\mu(\mathbf{x}))$ denotes the output density of the payoff function and plays an important role to determine the best arms to pull.

The intuition behind the likelihood ratio is as follows. Assuming enough data has been collected, the GP posterior mean $\mu(\mathbf{x})$ provides a good estimation about the distribution of rewards for the bandits. Bandits with unusually large rewards are associated with small values of p_μ, while bandits with frequent, average rewards are associated with large values of p_μ. Because the output density p_μ appears in the denominator of (2.14), the likelihood ratio assigns more weight to bandits with extreme payoffs. As such, the likelihood serves as an *attention mechanism* which encourages the algorithm to explore bandits whose rewards are thought to be abnormally large, while penalizing the other mediocre bandits by assigning them small weights. This attention mechanism is expected to provide the greatest gains in situations where the payoff of most bandits is concentrated around the average, and the highest possible payoff is several standard deviations away from most payoff values. In other words, if the distribution of payoffs generated by the bandits does not exhibit a heavy right tail, the benefits of using an output-weighted acquisition function will likely be marginal.

To obtain a well-behaved (i.e. smooth and bounded) analytical approximation of the likelihood ratio, we use a Gaussian mixture model (GMM) [43],

$$w(\mathbf{x}) \approx \sum_{k=1}^{n_{\text{GMM}}} \alpha_k \mathcal{N}(\mathbf{x}; \boldsymbol{\gamma}_k, \boldsymbol{\Sigma}_k), \tag{2.15}$$

where $\mathcal{N}(\mathbf{x}; \boldsymbol{\gamma}_k, \boldsymbol{\Sigma}_k)$ denotes the kth component of the mixture model with mean $\boldsymbol{\gamma}_k$ and covariance $\boldsymbol{\Sigma}_k$.

The resulting output-weighted acquisition function for the bandit optimization problem is given by

$$a_{\text{LW-UCB}}(\mathbf{x}) = \mu(\mathbf{x}) + \kappa w(\mathbf{x}) \sigma(\mathbf{x}), \tag{2.16}$$

where the subscript 'LW-UCB' stands for 'likelihood-weighted UCB'. Equation (2.16) is subject to the same bandit-selection policy as the acquisition functions in §c:

$$\mathbf{x}_{t+1} = \arg\max_{\mathbf{x}} a_{\text{LW-UCB}}(\mathbf{x}|\mathcal{D}). \tag{2.17}$$

In general, the minimization problem can be efficiently solved with an L-BFGS optimizer with random restarts [41], where the gradient of the acquisition function with respect to the inputs $\mathbf{x}$ can be computed analytically for the squared exponential covariance kernel [36], or using automatic differentiation [44] for more general kernel choices. The workflow for output-weighted sampling with LW-UCB is summarized in algorithm 1. In general, the computation of the likelihood ratio is not restricted to a Gaussian mixture approximation. However, the Gaussian mixture approximation helps the proposed algorithm in two ways. First, it provides a probability density over the output weights that naturally up-weights the importance of rare payoffs and down-weights the relatively average payoffs. Second, it acts as a smoother for the weight function, facilitating computation of gradients of the acquisition function (either analytically or via automatic differentiation) in gradient-based optimization. As such, the proposed approach yields an acquisition function $a_{\text{LW-UCB}}(\mathbf{x})$ that can be cheaply evaluated at any continuous query point $\mathbf{x}$, and is differentiable with respect to $\mathbf{x}$. However, in the bandit setting, there is no need to for gradient-based optimization of $a_{\text{LW-UCB}}(\mathbf{x})$, and instead we perform a simple grid search over all bandits to identify which one should be evaluated next.

Algorithm 1 . The LW-UCB algorithm.

```
%%Result: Write here the result
Input: Small initial dataset $\mathcal{D}=\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$;
while $t < T$ do
    Fit GP model to dataset $\mathcal{D}$ using (2.6) and obtain posterior mean (2.8a) and variance (2.8b);
    Compute likelihood ratio (2.14) and fit Gaussian mixture model (2.15) to it;
    Select best next bandit $\mathbf{x}_{t+1}$ by maximizing (2.16);
    Collect new reward $y_{t+1}=f(\mathbf{x}_{t+1})+\epsilon_{t+1}$ and append $(\mathbf{x}_{t+1}, y_{t+1})$ to dataset $\mathcal{D}$;
end
```

(e) Theoretical guarantees

Following the original work of Srinivas *et al.* [7], we are able to derive theoretical zero-regret guarantees suggesting that the proposed LW-UCB acquisition criterion achieves asymptotic convergence. This is a straightforward extension of the theoretical result reported in [7], with the main difference being that here we need to consider a context-dependent exploration term that represents the effect of the likelihood weights $w(\mathbf{x})$. Specifically, following the notation of Srinivas *et al.* [7], we consider a finite-size decision set D and the acquisition function

$$a(\mathbf{x}) = \mu_{t-1}(\mathbf{x}) + \beta_t^{1/2} w_t(\mathbf{x}) \sigma_{t-1}(\mathbf{x}), \tag{2.18}$$

where the sampling weights $w_t(\mathbf{x})$ are non-vanishing and bounded but otherwise arbitrary. Note that in all experiments presented in §3, we have fixed $\beta_t = 1$ for simplicity. Our main result is summarized in the following theorem.

Theorem 2.1. *Assume that $b \le w_t(\mathbf{x}) \le B$ for all $\mathbf{x} \in D$ and $t \ge 1$, with $b > 0$. Let $\delta \in (0, 1)$ and $\beta_t = 2\log(|D|\pi^2 t^2/(6\delta))/b^2$. The regret bound for the acquisition scheme in (2.18) is such that*

$$\mathbb{P}\left\{\forall T \ge 1, R_T \le B\sqrt{C_1 T \beta_T \gamma_T}\right\} \ge 1 - \delta, \tag{2.19}$$

where γ_T is the maximum information gain after T rounds (see equation (7) in Srinivas et al. [7]).

Proof. The proof is provided in the electronic supplementary material, document accompanying this manuscript. ■

We also note that Sapsis *et al.* [45] have recently provided new theoretical results on the optimality of output-weighted acquisition criteria. These results are focused on quantifying the convergence rate of approximation errors, suggesting that asymptotic convergence depends on the number of data that one can collect, in relation to the underlying reproducing kernel Hilbert space (RKHS) defined by the chosen GP kernel function (see §2.2 in [45] for more details).

3. Results

In all numerical studies considered in this work, we initialize the algorithm with $n = 3$ random input–output pairs and compare the performance of EI, TS, V-UCB, GP-UCB and LW-UCB. Our metric of success is the log-cumulative regret over time. Unless otherwise indicated, we conduct a series of 100 random experiments, each with a different choice of initial data, and report the median of the metric of interest. Variability across experiments is quantified using the median absolute deviation.

(a) Synthetic benchmarks

We demonstrate the performance of LW-UCB for three synthetic test functions. We consider 2500 bandits arranged on a uniform 50×50 grid with rewards being given by the value of the test function at that point in the domain. The rewards collected during optimization are corrupted by small Gaussian noise with $\sigma_n = 10^{-4}$.

We begin with the Cosine function from [46],

$$f(\mathbf{x}) = 1 - [u^2 + v^2 - 0.3\cos(3\pi u) - 0.3\cos(3\pi v)], \tag{3.1}$$

where $u = 1.6x_1 - 0.5$, $v = 1.6x_2 - 0.5$ and $\mathbf{x} \in [0, 1]^2$. For $n_{\mathrm{GMM}} = 2$, figure 2*a* shows that LW-UCB performs better than the other methods as it leads to faster identification of the best bandit. Moreover, figure 2*a* demonstrates how the likelihood ratio highlights the importance of the bandits and favours exploration of those with the highest rewards. We also note the subpar performance of EI, consistent with the discussion in [47].

Next, we consider the Michalewicz function [46],

$$f(\mathbf{x}) = \sin(\pi x_1)\sin^{20}(\pi x_1^2) + \sin(\pi x_2)\sin^{20}(2\pi x_2^2), \tag{3.2}$$

with $\mathbf{x} \in [0, 1]^2$. This function is more challenging than the Cosine function as it exhibits large areas of 'flatland' (i.e. many mediocre bandits) and a very deep and narrow well located slightly off centre (i.e. rare bandits with extreme payoffs). For $n_{\mathrm{GMM}} = 4$, figure 2*b* shows that LW-UCB outperforms the competition by a substantial margin. Figure 2*b* also makes it visually clear that the likelihood ratio assigns more weight to the best bandits. Interestingly, we have found that the

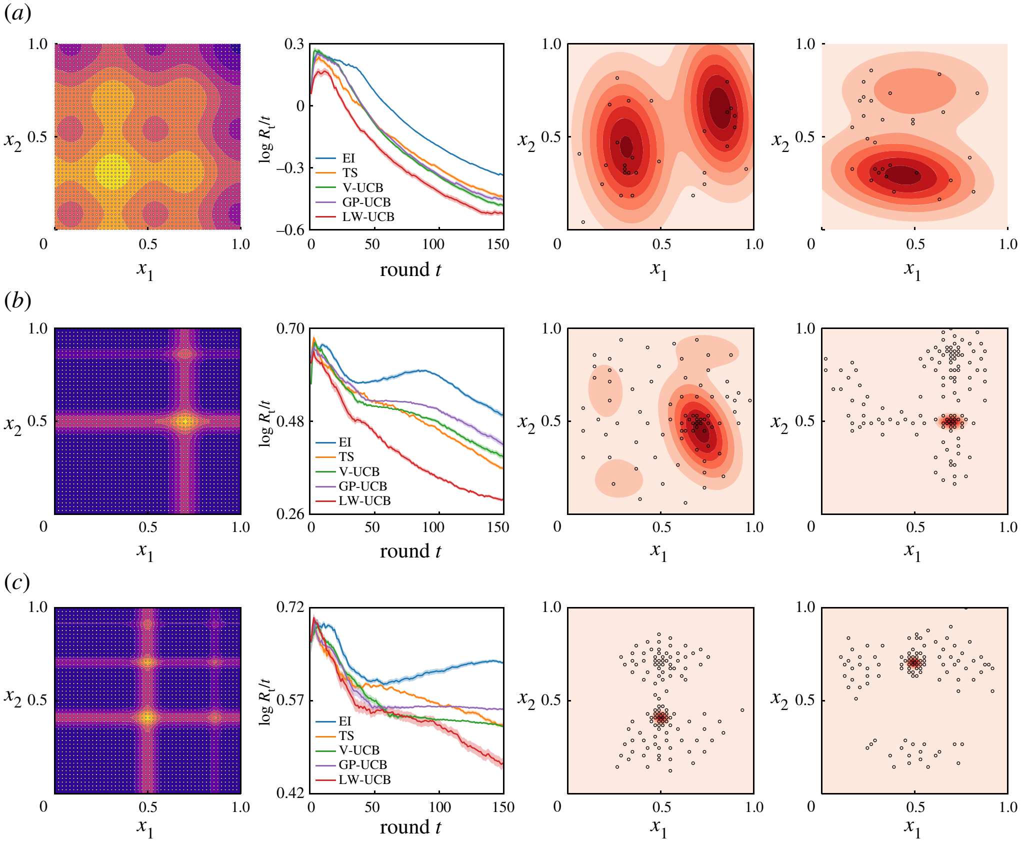


Figure 2. Synthetic benchmarks. From left to right: locations of the bandits (white circles) and associated rewards (background colour); cumulative regret for various acquisition functions; for two representative trials of LW-UCB, distribution of the likelihood ratio (background colour) learned by the GP model from the visited bandits (open circles) after $t = 150$ rounds. (*a*) Cosine function, (*b*) Michalewicz function and (*c*) modified Michalewicz function. (Online version in colour.)

likelihood ratio sometimes discovers a broader area where other sub-optimal solutions are also captured.

For an even more challenging test case, we introduce a modified version of the Michalewicz function which features multiple small 'islands' associated with extreme payoffs. Specifically, the function

$$f(\mathbf{x}) = \sin(\pi x_1)\sin^{20}(2\pi x_1^2) + \sin(\pi x_2)\sin^{20}(3\pi x_2^2), \tag{3.3}$$

has six extreme local minima and a number of steep valleys in the domain $\mathbf{x} \in [0, 1]^2$, making it quite difficult for the algorithms to identify the best bandits. Figure 2*c* shows that despite the added difficulty, LW-UCB again exhibits outstanding convergence behaviour, with the other acquisition functions struggling to identify the best bandits and therefore yielding poor performance. We also note that the likelihood ratio not only emphasizes the best area for rewards but is also able to identify sub-optimal solutions of somewhat lesser importance, demonstrating the ability of our approach to provide a good balance between exploration and exploitation.

To investigate the effect of the likelihood ratio on run-time, we record the time required to perform one iteration of the Bayesian algorithm. (This includes training the GP model, computing the likelihood ratio and the GMM approximation for LW-UCB, and optimizing the acquisition function.) Consistent with [36], table 1 shows that the run-times for LW-UCB are on the same order of magnitude as the other criteria. The additional cost is attributable to the computation and sampling of the likelihood ratio, and presumably can be alleviated using recent advances in

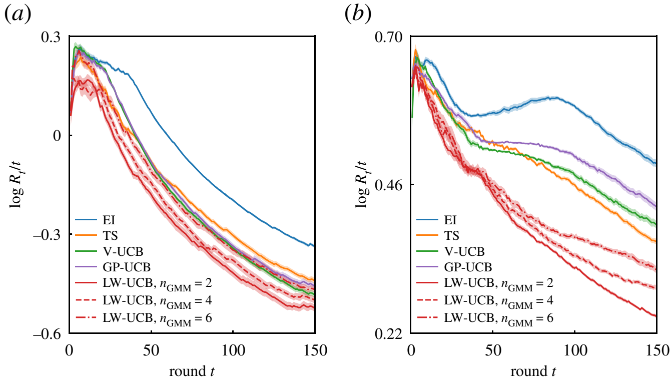


Figure 3. For the synthetic functions in §a, performance of LW-UCB with various values of n_{GMM} compared to the other acquisition functions considered in this work. (*a*) Cosine, (*b*) Michalewicz and (*c*) modified Michalewicz. (Online version in colour.)

Table 1. Single-iteration run-time (in seconds) averaged over ten experiments.

	Cosine	Michalewicz	modified Michalewicz
EI	0.49	0.52	0.68
TS	0.55	0.53	0.63
V-UCB	1.36	1.28	1.50
GP-UCB	1.36	1.28	1.50
LW-UCB	4.19	3.94	4.51

sampling methods for GP posteriors [48]. We also note that the cost of training the GP model at round t scales as $\mathcal{O}(t^3)$—a cost which is expected to dominate that of fitting the kernel density estimator and the Gaussian mixture needed to compute the LW-UCB acquisition. As such, we expect that the cost of using the LW-UCB criterion should become similar to the cost of UCB as the total number of rounds T is increased.

We have also investigated the sensitivity of the LW-UCB criterion to the size of the GMM used in the approximation of the likelihood ratio. For the three synthetic functions (3.1)–(3.3), we repeated the experiments with two additional values of n_{GMM}. Figure 3 shows that the performance of LW-UCB is essentially independent of the number of Gaussian components used in (2.15) when the latent function is relatively simple, and that larger values of n_{GMM} are preferable when the complexity of the landscape grows and the number of optimal regions increases.

(b) A systematic study: wheel bandits

In this section, we consider a variant of the contextual wheel bandit problem discussed in [33]. The feasible domain is the unit disc ($0 \le r \le 1$) which is divided into five disjoint sectors. The inner disc ($0 \le r \le \rho$) is sub-optimal with reward 0.2. The upper left, lower right and lower left quadrants of the outer ring ($\rho \le r \le 1$) are also sub-optimal, with rewards 0.05, 0.1 and 0, respectively (figure 4). The optimal bandits are located in the upper right quadrant of the outer ring and return a reward of 1, significantly higher than the other quadrants. The parameter ρ determines the difficulty of the problem. For small ρ, the optimal region accounts for a large fraction of the domain, while for large ρ the difficulty significantly increases. We generate the bandits on a 70 × 70 uniform grid and retain those lying inside the unit disc. Each bandit produces noisy rewards with $\sigma_n = 10^{-3}$.

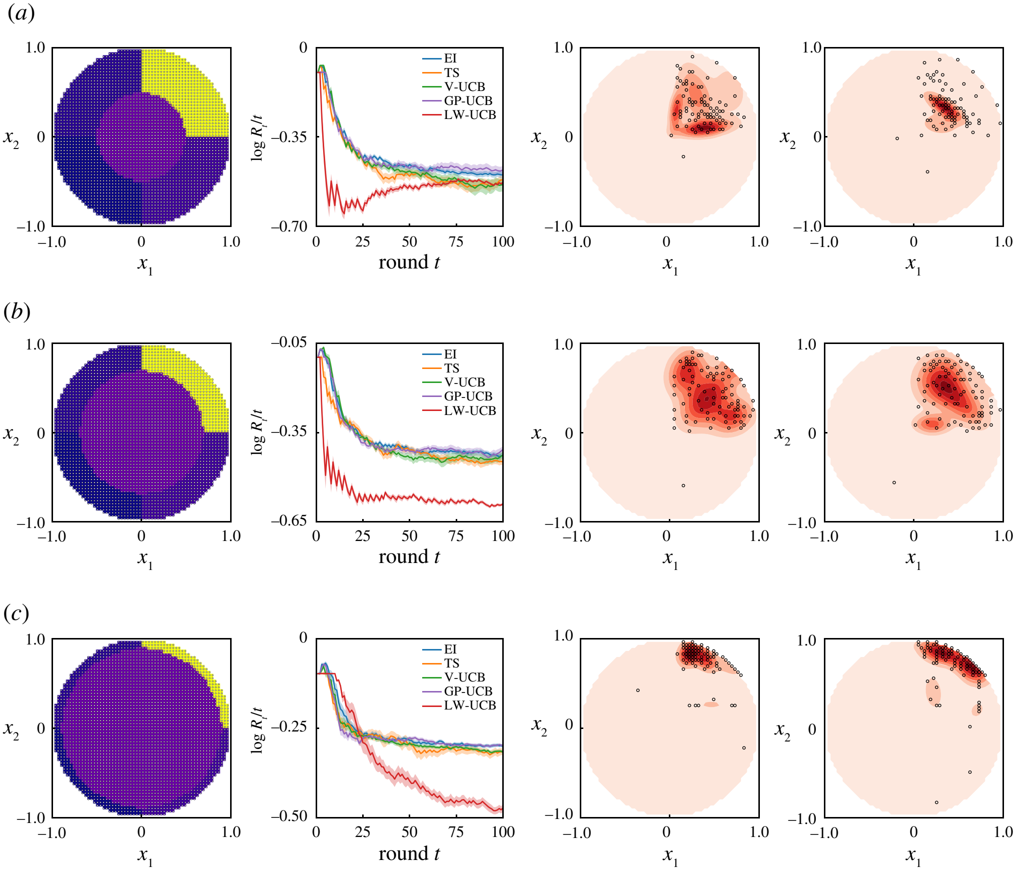


Figure 4. Wheel bandit problem. From left to right: locations of the bandits (white circles) and associated rewards (background colour); cumulative regret for various acquisition functions; and for two representative trials of LW-UCB, distribution of the likelihood ratio (background colour) learned by the GP model from the visited bandits (open circles) after $t = 100$ rounds. (*a*) $\rho = 0.5$, (*b*) $\rho = 0.7$ and (*c*) $\rho = 0.9$. (Online version in colour.)

For $n_{\mathrm{GMM}} = 4$, figure 4 shows that the proposed LW-UCB criterion leads to significant gains in performance compared to conventional acquisition functions, especially as the value of ρ increases and the optimal bandits become scarcer. Figure 4 also shows that the attention mechanism embedded in the likelihood ratio encourages exploration of the extreme-reward region. It is also interesting to note that in all cases investigated, the expected improvement, Thompson sampling, V-UCB and GP-UCB deliver nearly identical performance, even in the asymptotic regime, unlike LW-UCB which provides consistently faster convergence.

(c) Spatio-temporal environment monitoring with sensor networks

Finally, we demonstrate the approach using two real-world datasets: the temperature dataset considered in [7] and the air quality dataset considered in [49].

(i) Temperature dataset

The temperature dataset[2] contains temperature measurements collected by 46 sensors deployed in the Intel Berkeley Research lab (figure 5*a*). As in [7], our goal is to find locations of highest

[2] See http://db.csail.mit.edu/labdata/labdata.html.

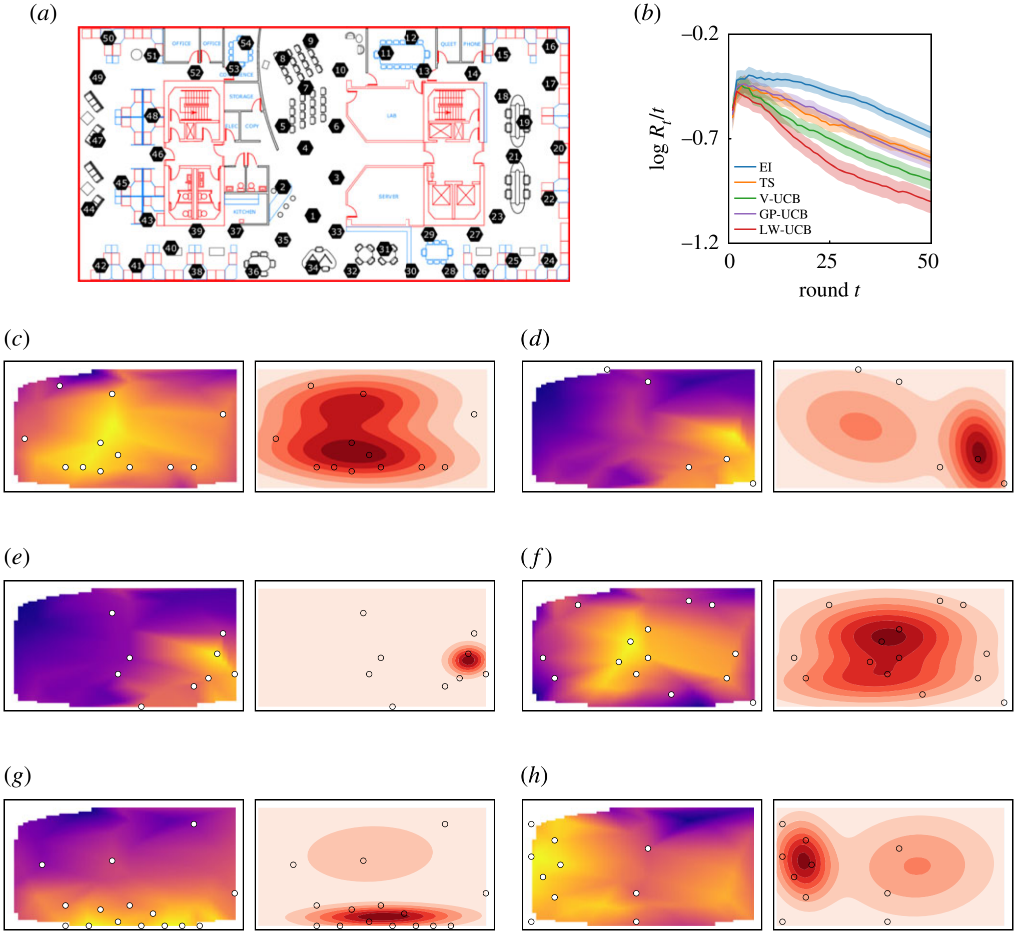
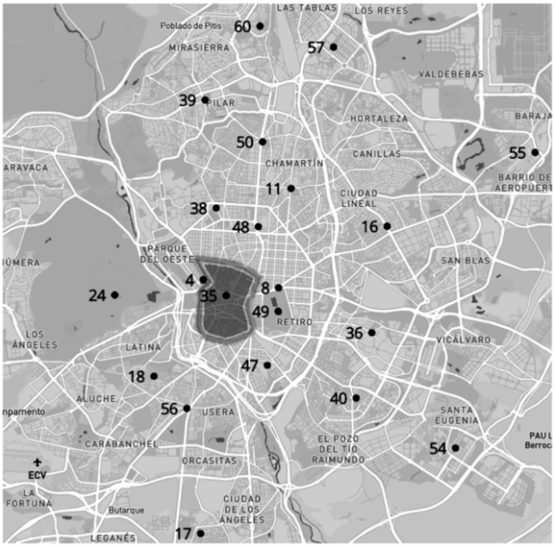


Figure 5. Spatio-temporal monitoring of room temperature with sensor networks. (*a*) Sensor locations; (*b*) cumulative regret for various acquisition functions; and (*c*–*h*) for six representative trials of LW-UCB, spatial distribution of temperature (left panel) and the likelihood ratio (right panel) learned by the GP model from the activated sensors (circles) after $t = 50$ rounds. (Online version in colour.)

temperature by sequentially activating the available sensors while using as few sensor switches as possible in order to save electric power. Our working dataset consists of 500 temperature snapshots collected every 10 min over a three-day period. For each temperature snapshot, we initialize the algorithm by randomly activating $n = 3$ sensors. The sensors (i.e. the bandits) produce rewards that are corrupted by small Gaussian noise with $\sigma_n = 10^{-4}$. We use $n_{\mathrm{GMM}} = 2$ for the GMM approximation of the likelihood ratio.

For this real-world problem, figure 5*b* shows that LW-UCB performs better than the other acquisition schemes. Figure 5*c*–*h* shows that the likelihood ratio draws the algorithm's attention to the bandits whose rewards are high by artificially inflating the model uncertainty for these bandits. We note that, in contrast to the examples considered previously, here the bandits are few and far between. For instance, there is no sensor data available in the server room and the stairwell (figure 5*a*). Because of the sparsity of the data, finding the best sensor to activate is more challenging. But this does not seem to negatively affect the LW-UCB acquisition criterion, which is able to identify and explore the relevant areas more intelligently than the other acquisition functions.



sensor id.	sensor location
4	Pza. de España
8	Escuelas Aguirre
11	Avda. Ramón y Cajal
16	Arturo Soria
17	Villaverde
18	Farolillo
24	Casa de Campo
35	Pza. Del Carmen
38	Moratalaz
39	Cuatro Caminos
40	Barrio del Pilar
47	Vallecas
48	Mendez Alvaro
49	Castellana
50	Parque del Retiro
54	Plaza Castilla
55	Ensanche de Vallecas
56	Urb. Embajada
57	Pza. Elíptica
	Sanchinarro
58	El Pardo
60	Tres Olivos

Figure 6. Spatio-temporal monitoring of air quality with sensor networks: locations of 22 out of the 24 sensors (i.e. excluding sensors #27 (Barajas Pueblo) and #59 (Juan Carlos I)) used to monitor air quality in the Madrid metropolitan area (original figure from [50]).

(ii) Air quality dataset

The air quality dataset[3] contains concentration measurements of pollutants and other particles collected by 24 sensors deployed in the Madrid metropolitan area. Each sensor is uniquely identified by its longitude, latitude and elevation. We focus on finding the locations of highest nitrogen dioxide (NO_2) by sequentially activating the available sensors to identify the region of worst air quality. The parsed data consists of 200 pollution snapshots collected every hour over a 10-day period in March 2018. For each pollution snapshot, we initialize the algorithm by randomly activating $n = 3$ sensors. As in the temperature example, the rewards produced by each sensor are corrupted by small Gaussian noise with $\sigma_n = 10^{-4}$. We use $n_{\mathrm{GMM}} = 2$ for the GMM approximation of the likelihood ratio.

We consider two cases: the partial-context case in which we only use longitude and latitude as the contextual information for each sensor (we summarize the locations for the sensors in figure 6); and the full-context case in which elevation is also accounted for. The partial-context case is worthy of investigation because, for the geographical area considered, the effect of elevation on NO_2 concentration should be quite small. Also, the number of sensors is relatively small (24), so using partial contextual information allows us to reduce the rank of the problem.

For both cases, figure 7*a*,*b* shows that LW-UCB outperforms the other acquisition functions considered. The snapshots shown in figure 7*c*–*h* reinforce the utility of the likelihood ratio to identify regions of high NO_2 concentration (i.e. poor air quality) more efficiently. As in the temperature example, the sparsity of the data does not seem to hamper the ability of LW-UCB to converge to the optimal bandits faster than the other acquisition schemes.

4. Conclusion

We have proposed a novel output-weighted acquisition function (LW-UCB) for sequential decision-making. Our approach leverages the information provided by the GP regression

[3] https://datos.madrid.es/portal/site/egob.

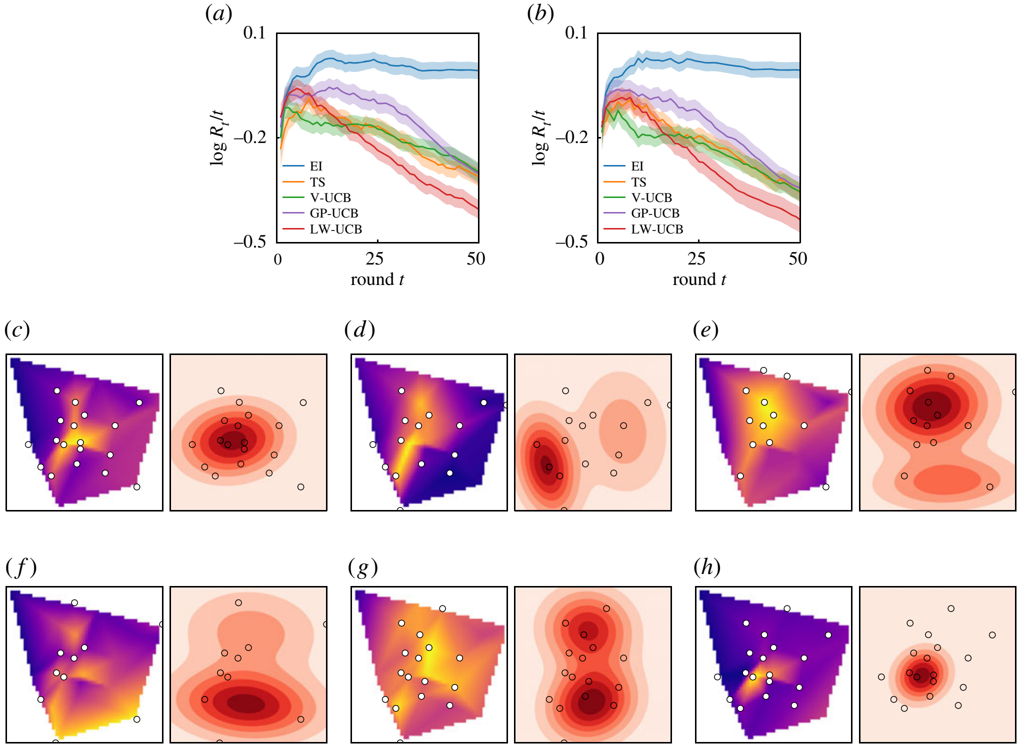


Figure 7. Spatio-temporal monitoring of air quality with sensor networks. Cumulative regret for (*a*) the partial-context case and (*b*) the full-context case; and (*c*–*h*) for the partial-context case and six representative trials of LW-UCB, spatial distribution of NO_2 concentration (left panel) and the likelihood ratio (right panel) learned by the GP model from the activated sensors (circles) after $t = 50$ rounds. (Online version in colour.)

model to regularize uncertainty and favour exploration of abnormally large payoff values. The regularizer takes the form of a sampling weight—the likelihood ratio—and can be efficiently approximated by a Gaussian mixture model. The likelihood ratio provides a principled way to balance exploration and exploitation in multi-armed bandit optimization problems where the goal is to maximize the cumulative reward. The benefits of the proposed method have been systematically established via several benchmark examples which demonstrated the superiority of our method compared to classical acquisition functions (expected improvement, Thompson sampling and two variants of UCB).

Though the proposed LW-UCB criterion yields superior performance in bandit problems, several questions remain open. First, a theoretical analysis of the convergence behaviour of LW-UCB is needed, in the same way that information gain has helped characterize the convergence of GP-UCB [7,8]. The second avenue is to investigate more complex cases with high-dimensional contexts and multi-output GP priors. The latter can be readily accommodated in our JAX implementation which leverages automatic differentiation to allow efficient gradient-based optimization of the LW-UCB criterion for arbitrary GP priors. The third question has to do with extending the proposed approach to other Bayesian inference schemes, e.g. Bayesian linear regression [21], Bayesian neural networks [33] and variational inference [51]. How to choose the number of modes in the Gaussian mixture model is also worthy of investigation. Our analysis here was largely empirical, and there are more systematic ways to optimally select the number of Gaussian mixtures, such as the Silhouette score, the distance between GMMs and the Bayesian information criterion. Fourthly, in the electronic supplementary material, B, we present a fabricated case where the performance of the LW-UCB acquisition function could be biased due to the combination of very high noise and scarce data in the left tail of the payoff function distribution. One way to mitigate this issue is to only consider the right tail of the posterior mean for computing the importance weight. Another interesting question is that of extending the output-weighted approach to stochastic and convex bandit problems [52,53]. Finally, there is the question of how to adapt the proposed framework for use in more general Markov decision processes and reinforcement learning problems [54] where contextual information is typically high-dimensional and rewards are observed after multiple trials rather than instantaneously.

Data accessibility. All code and data accompanying this manuscript are available at https://github.com/PredictiveIntelligenceLab/jax-bandits. The data are provided in electronic supplementary material [55].

Authors' contributions. A.B.: conceptualization, formal analysis, investigation, methodology, software, supervision, validation, visualization, writing—original draft; T.S.: conceptualization, formal analysis, funding acquisition, investigation, methodology, project administration, resources, supervision, writing—review and editing; P.P.: conceptualization, formal analysis, funding acquisition, investigation, methodology, project administration, resources, software, supervision, writing—review and editing.

All authors gave final approval for publication and agreed to be held accountable for the work performed therein.

Conflict of interest declaration. We declare we have no competing interests.

Funding. Y.Y. and P.P. acknowledge support from the DOE grant no. DE-SC0019116, AFOSR grant no. FA9550-20-1-0060 and DOE-ARPA grant no. DE-AR0001201. A.B. and T.S. would like to thank the support from the AFOSR-MURI grant no. FA9550- 21-1-0058 and the ONR grant no. N00014-21-1-2357.

References

1. Li L, Chu W, Langford J, Schapire R. 2010 A contextual-bandit approach to personalized news article recommendation. In *Proc. of the 19th Int. Conf. on World Wide Web, Raleigh, NC, USA, 26–30 April 2010*, pp. 661–670. New York, NY: ACM Press.
2. Kawale J, Bui H, Kveton B, Tran-Thanh L, Chawla S. 2015 Efficient Thompson Sampling for Online Matrix-Factorization Recommendation. In *Advances in Neural Information Processing Systems, Montréal, Canada, 7–12 December 2015*, vol. 28, pp. 1297–1305. Red Hook, NY: Curran Associates, Inc.
3. Dearden R, Friedman N, Russell S. 1998 Bayesian Q-learning. In *AAAI/IAAI, Madison, Wisconsin, US, 26–30 July 1998*, pp. 761–768. Palo Alto, CA: AAAI.
4. Osband I, Blundell C, Pritzel A, Van Roy B. 2016 Deep exploration via bootstrapped DQN. In *Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016*, vol. 29, pp. 4026–4034. Red Hook, NY: Curran Associates, Inc.
5. Azizzadenesheli K, Brunskill E, Anandkumar A. 2018 Efficient exploration through Bayesian deep Q-networks. In *2018 Information Theory and Applications Workshop (ITA), San Diego, CA, 11–16 February 2018*, pp. 1–9. New York, NY: IEEE. See https://ieeexplore.ieee.org/document/8503252.
6. Li C, Bai K, Li J, Wang G, Chen C, Carin L. 2019 Adversarial learning of a sampler based on an unnormalized distribution. In *The 22nd Int. Conf. on Artificial Intelligence and Statistics, Okinawa, Japan, 16–18 April 2019*, pp. 3302–3311. PMLR.
7. Srinivas N, Krause A, Kakade S, Seeger M. 2009 Gaussian process optimization in the bandit setting: no regret and experimental design. (http://arxiv.org/abs/0912.3995)
8. Krause A, Ong CS. 2011 Contextual Gaussian process bandit optimization. In *Advances in Neural Information Processing Systems, Granada, Spain, 12–17 December 2011*, pp. 2447–2455. Red Hook, NY: Curran Associates, Inc.
9. Sacks J, Welch W, Mitchell T, Wynn H. 1989 Design and analysis of computer experiments. *Stat. Sci.* **4**, 409–423. (doi:10.1214/ss/1177012413)
10. Saha U, Thotla S, Maity D. 2008 Optimum design configuration of Savonius rotor through wind tunnel experiments. *J. Wind Eng. Ind. Aerodyn.* **96**, 1359–1375. (doi:10.1016/j.jweia.2008.03.005)
11. Sahli Costabal F, Perdikaris P, Kuhl E, Hurtado D. 2019 Multi-fidelity classification using Gaussian processes: accelerating the prediction of large-scale computational models. *Comput. Methods Appl. Mech. Eng.* **357**, 112602. (doi:10.1016/j.cma.2019.112602)
12. Sahli Costabal F, Yang Y, Perdikaris P, Hurtado D, Kuhl E. 2020 Physics-informed neural networks for cardiac activation mapping. *Front. Phys.* **8**, 42. (doi:10.3389/fphy.2020.00042)

13. Forrester A, Sóbester A, Keane A. 2007 Multi-fidelity optimization via surrogate modelling. *Proc. R. Soc. A* **463**, 3251–3269. (doi:10.1098/rspa.2007.1900)
14. Sarkar S, Mondal S, Joly M, Lynch M, Bopardikar S, Acharya R, Perdikaris P. 2019 Multifidelity and multiscale Bayesian framework for high-dimensional engineering design and calibration. *J. Mech. Des.* **141**, 121001. (doi:10.1115/1.4044598)
15. Shan S, Wang GG. 2010 Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions. *Struct. Multidiscipl. Optim.* **41**, 219–241. (doi:10.1007/s00158-009-0420-2)
16. Perdikaris P, Venturi D, Karniadakis GE. 2016 Multifidelity information fusion algorithms for high-dimensional systems and massive data sets. *SIAM J. Sci. Comput.* **38**, B521–B538. (doi:10.1137/15M1055164)
17. Bouhlel MA, Bartoli N, Otsmane A, Morlier J. 2016 Improving kriging surrogates of high-dimensional design models by Partial Least Squares dimension reduction. *Struct. Multidiscipl. Optim.* **53**, 935–952. (doi:10.1007/s00158-015-1395-9)
18. Wan ZY, Vlachas P, Koumoutsakos P, Sapsis T. 2018 Data-assisted reduced-order modeling of extreme events in complex dynamical systems. *PLoS ONE* **13**, e0197704. (doi:10.1371/journal.pone.0197704)
19. Mohamad M, Sapsis T. 2018 Sequential sampling strategy for extreme event statistics in nonlinear dynamical systems. *Proc. Natl Acad. Sci. USA* **115**, 11 138–11 143. (doi:10.1073/pnas.1813263115)
20. Auer P. 2002 Using confidence bounds for exploitation-exploration trade-offs. *J. Mach. Learn. Res.* **3**, 397–422.
21. Chu W, Li L, Reyzin L, Schapire R. 2011 Contextual bandits with linear payoff functions. In *Proc. of the 14th Int. Conf. on Artificial Intelligence and Statistics, Fort Lauderdale, FL, 11–13 April 2011*, pp. 208–214. PMLR.
22. Schaul T, Quan J, Antonoglou I, Silver D. 2015 Prioritized experience replay. (http://arxiv.org/abs/1511.05952)
23. Agrawal R. 1995 Sample mean based index policies with O(log n) regret for the multi-armed bandit problem. *Adv. Appl. Probab.* **27**, 1054–1078. (doi:10.2307/1427934)
24. Dani V, Hayes T, Kakade S. 2008 Stochastic linear optimization under bandit feedback. In *The 21st Annual Conf. on Learning Theory, Helsinki, Finland, 9–12 July 2008*, pp. 355–366. Omnipress.
25. Thompson W. 1933 On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* **25**, 285–294. (doi:10.1093/biomet/25.3-4.285)
26. Russo D, Van Roy B, Kazerouni A, Osband I, Wen Z. 2017 A tutorial on Thompson sampling. (http://arxiv.org/abs/1707.02038)
27. Chapelle O, Li L. 2011 An empirical evaluation of Thompson sampling. In *Advances in Neural Information Processing Systems, Granada, Spain, 12–17 December 2011*, pp. 2249–2257. Red Hook, NY: Curran Associates, Inc.
28. Kandasamy K, Krishnamurthy A, Schneider J, Póczos B. 2018 Parallelised Bayesian optimisation via Thompson sampling. In *Int. Conf. on Artificial Intelligence and Statistics, Lanzarote, Canary Islands, 9–11 April 2018*, pp. 133–142. PMLR.
29. Kaufmann E, Korda N, Munos R. 2012 Thompson sampling: an asymptotically optimal finite-time analysis. In *Int. Conf. on Algorithmic Learning Theory, Lyon, France, 29–31 October 2012*, pp. 199–213. Berlin; Heidelberg: Springer. See https://link.springer.com/chapter/10.1007/978-3-642-34106-9_18.
30. Leike J, Lattimore T, Orseau L, Hutter M. 2016 Thompson sampling is asymptotically optimal in general environments. (http://arxiv.org/abs/1602.07905)
31. Russo D, Van Roy B. 2016 An information-theoretic analysis of Thompson sampling. *J. Mach. Learn. Res.* **17**, 2442–2471.
32. Graves A. 2011 Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems, Granada, Spain, 12–17 December 2011*, pp. 2348–2356. New York, NY: Curran Associates, Inc.
33. Riquelme C, Tucker G, Snoek J. 2018 Deep Bayesian bandits showdown: an empirical comparison of Bayesian deep networks for Thompson sampling. (http://arxiv.org/abs/1802.09127)
34. Sapsis T. 2020 Output-weighted optimal sampling for Bayesian regression and rare event statistics using few samples. *Proc. R. Soc. A* **476**, 20190834. (doi:10.1098/rspa.2019.0834)
35. Blanchard A, Sapsis T. 2020 Output-weighted importance sampling for Bayesian experimental design and uncertainty quantification. (http://arxiv.org/abs/2006.12394)

36. Blanchard A, Sapsis T. 2021 Bayesian optimization with output-weighted importance sampling. *J. Comput. Phys.* **425**, 109901. (doi:10.1016/j.jcp.2020.109901)
37. Blanchard A, Sapsis T. 2020 Informative path planning for anomaly detection in environment exploration and monitoring. (http://arxiv.org/abs/2005.10040)
38. Vazquez E, Bect J. 2010 Convergence properties of the expected improvement algorithm with fixed mean and covariance functions. *J. Stat. Plann. Inference* **140**, 3088–3095. (doi:10.1016/j.jspi.2010.04.018)
39. Bradbury J et al. 2018 JAX: composable transformations of Python+NumPy programs.
40. Rasmussen CE, Williams C. 2006 *Gaussian processes for machine learning*. Cambridge, MA: MIT Press.
41. Liu D, Nocedal J. 1989 On the limited memory BFGS method for large scale optimization. *Math. Programm.* **45**, 503–528. (doi:10.1007/BF01589116)
42. Agrawal S, Goyal N. 2012 Analysis of Thompson sampling for the multi-armed bandit problem. In *Conf. on Learning Theory, Edinburgh, Scotland, 25–27 June 2012*, pp. 39.1–39.26. PMLR.
43. Murphy KP. 2012 *Machine learning: a probabilistic perspective*. Cambridge, MA: MIT Press.
44. Baydin AG, Pearlmutter BA, Radul AA, Siskind JM. 2015 Automatic differentiation in machine learning: a survey. (http://arxiv.org/abs/1502.05767)
45. Sapsis TP, Blanchard A. 2021 Optimal criteria and their asymptotic form for data selection in data-driven reduced-order modeling with Gaussian process regression. (http://arxiv.org/abs/2112.02636)
46. Azimi J, Fern A, Fern X. 2010 Batch Bayesian optimization via simulation matching. In *Advances in Neural Information Processing Systems, Vancouver, Canada, 6–11 December 2010*, pp. 109–117. Red Hook, NY: Curran Associates, Inc.
47. Qin C, Klabjan D, Russo D. 2017 Improving the expected improvement algorithm. In *Advances in Neural Information Processing Systems, Long Beach, CA, 4–9 December 2017*, vol. 30, pp. 5381–5391. NIPS.
48. Wilson J, Borovitskiy V, Terenin A, Mostowsky P, Deisenroth M. 2020 Efficiently sampling functions from Gaussian process posteriors. In *Int. Conf. on Machine Learning, online conference, 13–18 July 2020*, pp. 10 292–10 302. PMLR. (https://proceedings.mlr.press/v119/wilson20a.html)
49. Cai H, Cen Z, Leng L, Song R. 2021 Periodic-GP: Learning Periodic World with Gaussian Process Bandits. (http://arxiv.org/abs/2105.14422)
50. Lebrusán I, Toutouh J. 2020 Using smart city tools to evaluate the effectiveness of a low emissions zone in Spain: Madrid central. *Smart Cities* **3**, 456–478. (doi:10.3390/smartcities3020025)
51. Hoffman M, Blei D, Wang C, Paisley J. 2013 Stochastic variational inference. *J. Mach. Learn. Res.* **14**, 1303–1347.
52. Lattimore T, Gyorgy A. 2021 Improved regret for zeroth-order stochastic convex bandits. In *Conf. on Learning Theory, online conference, 15–19 August 2021*, pp. 2938–2964. PMLR. (https://proceedings.mlr.press/v134/lattimore21a.html)
53. Flaxman AD, Kalai AT, McMahan HB. 2004 Online convex optimization in the bandit setting: gradient descent without a gradient. (http://arxiv.org/abs/cs/0408007)
54. Sutton R, Barto A. 2018 *Reinforcement learning: an introduction*. Cambridge, MA: MIT Press.
55. Yang Y, Blanchard A, Sapsis T, Perdikaris P. 2022 Output-weighted sampling for multi-armed bandits with extreme payoffs. Figshare. (https://doi.org/10.6084/m9.figshare.c.5953885)