Gaussian Process Bandits: An Experimental Design Approach

Niranjan Srinivas
California Institute of Technology
niranjan@caltech.edu

Andreas Krause
California Institute of Technology
krausea@caltech.edu

Sham M Kakade
Toyota Technological Institute
sham@tti-c.org

Matthias Seeger
Saarland University
mseeger@mmci.uni-saarland.de

Abstract

We consider the online setting of optimizing an unknown function, sampled from a Gaussian Proccess, over a given bounded decision set so that that our cumulative regret is low. Our analysis of an upper confidence algorithm provides sublinear regret bounds for popular classes of kernels by exploiting a surprising connection to optimal experimental design– in particular, our rates do no explicitly depend on the dimensionality of the decision space, which need not even be a vector space.

1 Introduction

In many real-world problems, one needs to optimize a noisy function which is expensive to evaluate. Recent examples of interest include choosing the advertisements in sponsored search to maximize profit in a click-through model [1] and learning optimal control strategies for robots [2]. This problem can be naturally cast as a bandit optimization problem [3] with a large (possibly infinite) number of arms.

Gaussian Process Optimization (GPO) is a natural framework for studying these problems. The function to be optimized is assumed to reside in the Reproducing Kernel Hilbert Space (RKHS) associated with a suitable kernel function. This framework is powerful, yet flexible; we can have kernels over objects with structure, such as vectors, strings and graphs. The Upper Confidence Bound (UCB) algorithm [4, 5] has been used successfully as a heuristic in several applications. However, so far we are unaware of any theoretical performance analyses.

We analyze the UCB algorithm for GPO and prove sublinear regret bounds for popular classes of kernels by exploiting a surprising link between bandit optimization and optimal experimental design.

2 Analysis of the Gaussian Bandit UCB

A Gaussian process (c.f., [6]) is a collection of random variables such that every finite subset shares a joint Gaussian distribution. A GP $f(x) \sim GP(m(x), k(x,x'))$ is completely specified by its mean function $m(x) = \mathbb{E}[f(x)]$ and its covariance function $k(x,x') = \mathbb{E}[(f(x)-m(x))(f(x')-m(x'))]$. The smoothness we assume for our function is encoded by the covariance function. Some popular covariance functions include: the squared exponential kernel $k(x,x') = \exp(-\frac{|x-x'|^2}{2l^2})$ where l is the length-scale parameter, and the Matern class of covariance functions, given by $k(x,x') = \frac{2^{1-\nu}}{\Gamma(\nu)}(\frac{\sqrt{2\nu}|x-x'|}{l})^\nu K_\nu(\frac{\sqrt{2\nu}|x-x'|}{l})$ where ν is a parameter that controls smoothness and K_ν is a modified Bessel function.

Our objective while sampling is to maximize the sum of the 'rewards' we get, i.e., to choose our sampling points in order to maximize the sum of the function values obtained. If the decision set D is continuous, we discretize it into a set $V = V_T$. In our analysis, we assume that the decision region

```
**Algorithm 2.1:** GAUSSIAN BANDIT UCB$(D, \delta)$

$V$ : Discretization of decision space using $n$ points
Prior : $\hat{\mu}_1 \equiv 0$ , $\hat{\Sigma}_1 = \Sigma$
for $t \leftarrow 1$ to $T$
    Choose $x_t = \mathrm{argmax}_D(\hat{\mu}_t(i) + \sqrt{\beta_t}\hat{\Sigma}_t(i,i))$
    Sample $y_t = f(x_t) + \epsilon_t$
    Bayesian update:
    $\hat{\mu}_{t+1} = K(X_*, X)[K(X,X) + \sigma^2 I]^{-1} y_t$
    $\hat{\Sigma}_{t+1} = K(X_*, X_*) - K(X_*, X)[K(X,X) + \sigma^2 I]^{-1} K(X, X_*)$
```

Figure 1: The Gaussian bandit UCB algorithm. Hereby, $\beta_t = 4(\log(\frac{2T^{\tau+1}}{\delta}))^2$, X_* is the set of test points, i.e., all points in V. X is the set of training points x_t, and y are the function values y_t.

is bounded. For bounded decision spaces in R^d and many kernels, a polynomial increase (for fixed d) of $n = |V_T| = \mathcal{O}(T^\tau)$ for some $\tau > 0$ suffices to ensure that the discretization error shrinks at a rate of $1/\sqrt{T}$.

Sampling to maximize sum of the rewards is analogous to the classical multi-armed bandit problem, where we have one arm for each possible decision. In bandit problems, we have the exploration-exploitation trade-off; between the need to explore various possible options and the desire to constantly choose the empirically best option. The Upper Confidence Bound (UCB) algorithm chooses the arm by maximizing an upper confidence index. For each arm, the index is obtained by adding the current estimate of the mean and a measure of uncertainty for that estimate. So, if an arm lies in an unexplored region, its index is likely to be high even if its initial mean estimate is low. In this way, the UCB algorithm implicitly negotiates the exploration-exploitation trade-off.

Algorithms for bandit problems are usually analyzed in terms of their ‘cumulative regret’. For a particular choice of arm x_t, the instantaneous regret is $r_t = \mathbb{E}(f(x_*) - f(x_t))$ where $\mathbb{E}(f(x))$ is maximized at x_*, and the expectation is with respect to the function f being drawn from the Gaussian process prior. Essentially, r_t it is the expected loss of reward we incur due to our lack of knowledge of the best arm to play. The cumulative regret R_T at time T is the sum of instantaneous regrets: $R_T = \sum_{t=1}^{T} r_t$. In a multi-armed bandit problem, the quest is to find a no-regret algorithm for choosing the arms to play; an algorithm $\mathcal{A}$ is said to be *no-regret* if $R_{\mathcal{A}}(T) \sim o(T)$.

For the K-armed bandit problem, the UCB algorithm is no-regret; however, $R_T = O(\sqrt{KT})$ and therefore the bound is vacuous for infinitely many arms (or if $K = \Omega(T)$). We provide a no-regret bound for infinitely armed bandits in a GP framework by replacing K in the bound by the maximum possible information gain due to sampling, thus connecting GP bandit optimization and optimal experimental design. In the finite arms case, the regret grows polynomially with the number of arms because each is independent. However, the GP setting imposes smoothness through the covariance structure, and therefore playing one ‘arm’ gives more information about other ‘arms’ in its neighbourhood – therefore the information gain grows sublinearly during the sampling process.

Let V represent the discretization of the decision set. We observe points $y_s = x_s^T F_V + \epsilon_s$, where F is the unknown reward function, ϵ_i is a Gaussian white noise process with variance σ^2 and x_s is an indicator vector which refers to the particular member s of V we choose to observe. We can think of picking the set of vectors x_i in terms of choosing the matrix A with x_i as the columns. Then, we observe $Y_A = A^T F_V + \epsilon$, where $\epsilon \sim \mathbb{N}(0, \sigma^2 I)$.

We define the information gain to be

$$\gamma_T = \max_A I(F_V; Y_A) = \max_A (H(Y_A) - H(Y_A|F_V)) \qquad (1)$$

where $I(\cdot,\cdot)$ stands for mutual information.

Our main result is the following regret bound in terms of maximal information gain. This connects the GP bandit problem with the problem of experimental design, where points of measurement need to be chosen in order to maximize the information gain.

Theorem 1 *Let* $0 < \delta < 1$. *If we run the Gaussian Bandit UCB algorithm with discretization* V, $|V| = T^{\tau}$ *and parameter* δ, *the cumulative regret* R_T *after* T *plays is bounded as:*

$$Prob(\forall T, R_T \leq \sqrt{8T\beta_T\gamma_T}) \geq 1 - \delta$$

where $\beta_T = 4(\log(\frac{2T^{\tau+1}}{\delta}))^2$.

In order to bound γ_T, we exploit the fact that information gain is *submodular*, i.e., the information gain on sampling from one point is larger when our number of earlier sample points is low than when it is high [7]. That is, $I(\mathcal{A} \cup \{s\}) - I(\mathcal{A}) \geq I(\mathcal{A}' \cup \{s\}) - I(\mathcal{A}')$ whenever $\mathcal{A} \subseteq \mathcal{A}'$ and $s \notin \mathcal{A}$. Submodularity implies a performance guarantee on the 'greedy' algorithm for maximizing the information gain.

Which inputs does the greedy algorithm pick? If we allow selecting arbitrary vectors of unit norm (instead of point evaluations), maximizing the RHS of (1) reduces to

$$\max_A H(Y_A) = \max_A A^T \Sigma A.$$

When $A = v$ is just a vector, i.e., we sample just one point at a time, we have

$$\max_A H(Y_A) = \max_{\|v\|_2 \leq 1} v^T \Sigma v \tag{2}$$

We know that (2) is maximized by selecting the eigenvector v of Σ with maximal absolute eigenvalue. This insight shows that the worst case bound occurs when the UCB algorithm is allowed to sample *eigenvectors* of Σ. Further, the submodularity of information gain allows us to choose the sampling points greedily - picking eigenvectors with maximal eigenvalues maximizes information gain. Formally:

Theorem 2 *The maximal information gain* γ_T *is bounded as follows:*

$$\gamma_T \leq (1 - \frac{1}{e})^{-1} \max_{m_1,\ldots,m_n} \sum_{t=1}^{n} \log(1 + \frac{m_t \lambda_t}{\sigma^2})$$

Thus, in order to bound the maximal information gain, we need to understand the spectral properties of the kernel matrices involved, and how they affect the optimal allocation $m_1, \ldots, m_n$. If we know the decay of the eigenvalues λ_i, we can bound the optimal allocation $m_1, \ldots, m_n$ using a fractional relaxation. We present the regret bounds for three popular classes of kernels below by exploiting their known spectral properties.

Consider finite-dimensional Bayesian linear regression, with Lipschitz-continuous basis functions $\phi(x)^T = (\phi_1(x), \phi_2(x), ..., \phi_q(x))$. We show that $\gamma_T = \mathcal{O}(\log(T))$. So, from Theorem 1, since $\beta_T = \mathcal{O}(\log^2(T))$,

$$R_T = \mathcal{O}(\sqrt{T \log^3(T)})$$

For the squared-exponential kernel with fixed lengthscale (i.e., independent of the discretization), we exploit the exponential decay of the eigenvalues to show that $\gamma_T = \mathcal{O}(\log^2(T))$ and therefore, reasoning as before,

$$R_T = \mathcal{O}(\sqrt{T \log^4(T)})$$

If the eigenvalues decay in a power law fashion, say with index α, we show that $\gamma_T = \mathcal{O}(T^{\frac{\tau+1}{\alpha}})$ and reasoning as earlier,

$$R_T = \mathcal{O}(T^{\frac{1}{2} + \frac{\tau+1}{2\alpha}}) \tag{3}$$

From (3), the average regret vanishes asymptotically if $\alpha > \tau+1$ - this is true for all Sacks-Ylvisaker kernels of order $r > \tau - 1$ (such as the Matern class, with smoothness parameter $\nu = r + \frac{1}{2}$, provided $\nu > 2$).

Therefore, we have sublinear regret bounds for Bayesian linear regression, the squared exponential kernel and a large class of Sacks-Ylvisaker kernels. We believe that the above analysis presents a natural framework to analyze Gaussian process optimization for a variety of kernels.

References

[1] Sandeep Pandey and Christopher Olston. Handling advertisements of unknown quality in search advertising. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 1065–1072. MIT Press, Cambridge, MA, 2007.

[2] Daniel Lizotte, Tao Wang, Michael Bowling, and Dale Schuurmans. Automatic gait optimization with gaussian process regression. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 944–949, 2007.

[3] H. Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58:527–535, 1952.

[4] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2-3):235–256, 2002.

[5] Varsha Dani, Thomas P. Hayes, and Sham M. Kakade. Stochastic linear optimization under bandit feedback. In *In submission*, 2008.

[6] C. E. Rasmussen and C. K.I. Williams. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. The MIT Press, 2006.

[7] A. Krause and C. Guestrin. Near-optimal nonmyopic value of information in graphical models. In *Proc. of Uncertainty in Artificial Intelligence (UAI)*, 2005.