Learning To Optimize Via Posterior Sampling

Dan Russo and Benjamin Van Roy
Stanford University

Motivation

➢*The Challenge*: Mathematical optimization is often used to guide decision making in complicated systems, but its effective use requires the ability to evaluate the performance impact of changes to the system. This is hard!
➢*Exciting Potential*: Modern information technology gives system designers an unprecedented opportunity to cheaply test potential improvements to the system.
➢*Goal*: Design methods that learn to attain near optimal performance through efficient experimentation. Inherent tradeoff between exploration and exploitation.

An Example: Whenever a customer visits an ecommerce website a set of items that are available for purchase is displayed. The company can choose how to price these products, which deals appear, and the order in which they are displayed. How can it learn to do this near optimally?

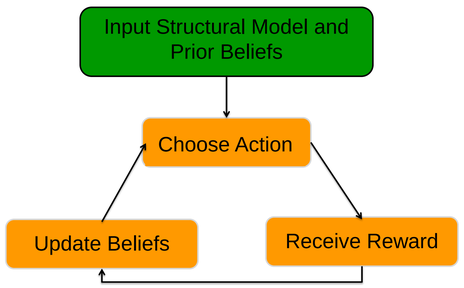
Mathematical Formulation

A Multiarmed Bandit Problem with Correlated Arms:
➢The goal is to choose an actions $A_t \in \mathcal{A}$ to maximize online performance $\sum_1^T f_\theta(A_t)$.
➢ Model uncertainty is captured by prior distribution over $\theta \in \Theta$.
➢ When an action A_t is chosen, a reward is observed:

$$R_t := f_\theta(A_t) + noise$$

➢Hope to bound performance relative to an algorithm that always chooses the optimal action:

$$\text{BayesRisk}(T) := \mathbb{E}\sum_{t=1}^{T}[f_\theta(A^*) - f_\theta(A_t)]$$



Main Contributions

➢ Study a promising, but poorly understood, posterior sampling (PS) algorithm for selecting actions.
➢Show PS satisfies a risk decomposition similar to that of upper confidence bound (UCB) algorithms. This connects the two types of algorithms, providing insight into why PS works well, and its potential advantages.
➢Use the risk decomposition to establish theoretical guarantees:
- Convert existing analysis of specific UCB algorithms to give guarantees for PS in important special cases
- A general bound that depends on a new measure of the complexity of a problem instance: the Margin Dimension.

Posterior Sampling Algorithm

For t=1,2,….
1. **Sample θ_t from posterior using MCMC**
2. **Select $A_t \in argmax_a f_{\theta_t}(a)$ using an optimization algorithm.**
3. **Observe Reward**

➢ Also known as Thompson Sampling, and Randomized Probability Matching.
➢ Algorithm is simple, often computationally efficient, and has been observed to have great empirical performance.
➢ Appealing Heuristic: "Sample an action according to the probability the action is optimal".

Our Question:
How do we think about this algorithm? Why does this work? Can we provide general theoretical guarantees? How do these guarantees depend on the problem instance?

Upper Confidence Bound (UCB) Algorithms

For t=1,2,….
1. **Set $U_t(a)$ to be the largest value of $f_\theta(a)$ that is statistically plausible given observed data.**
2. **Play $A_t \in argmax_a U_t(a)$**
3. **Observe Reward**

➢ Guiding Principle: "Optimism in the face of uncertainty."
- Encourages the selection of poorly understood actions.

➢There is a very large literature on these algorithms, and they can work extremely well.
➢Drawback: We need to construct the upper confidence bound $U_t(a)$. This choice dramatically affects performance and computational tractability.

Example: Linear Programming with Uncertain Objective

- Want to solve LP: $\max_{a\in\mathcal{A}} \theta^T a$
- $\mathcal{A} \subset \mathbb{R}^d$ is a polyhedron expressed in terms of linear inequalities. (e.g. $\mathcal{A} = \{a:\ a^T b_i \le c_i\ i = 1,..,n\}$)
- Multivariate Gaussian Prior: $\theta \sim N(\mu_0, \Sigma_0)$.

Assume noise is Gaussian, so posterior is multivariate Gaussian $\theta \sim N(\mu_t, \Sigma_t)$ and (μ_t, Σ_t) are given in closed form.

Predicted Reward *Uncertainty Bonus*

Posterior Sampling
1. Sample $\theta_t \sim N(\mu_t, \Sigma_t)$
2. Choose $A_t \in \text{argmax}_{a\in\mathcal{A}}\{\theta^T a\}$

Action Selection: Solve an LP!

UCB Algorithm
$U_t(a) = \mu_t^T a + \beta\|a\|_{\Sigma_t^{-1}}$
Action Selection: Solve: $\text{argmax}_{a\in\mathcal{A}}\{\mu_t^T a + \beta\|a\|_{\Sigma_t^{-1}}\}$
This is NP Hard

Risk Decompositions

UCB Risk Decomposition

If f_θ has range $[0, R_{max}]$ the Bayes Risk of a UCB algorithm executed with upper confidence indices $(U_t: t \ge 1)$ is bounded by:

$$\mathbb{E}\sum_{t=1}^{T}[U_t(A_t) - f_\theta(A_t)] + R_{max}\sum_{t=1}^{T}\mathbb{P}(f_\theta(A^*) > U_t(A^*))$$

Posterior Sampling Risk Decomposition

New Proposition: For *all* upper confidence indices $(U_t: t \ge 1)$ the Bayes Risk of PS is bounded by:

$$\mathbb{E}\sum_{t=1}^{T}[U_t(A_t) - f_\theta(A_t)] + R_{max}\sum_{t=1}^{T}\mathbb{P}(f_\theta(A^*) > U_t(A^*))$$

Interpretation:
➢ *"Performance can only be bad if you're learning a lot."* Risk is bounded by the uncertainty about the actions the algorithm selects. We expect to learn a lot by sampling an action if we're really uncertain about its value.
➢ A close theoretical connection between UCB algorithms and Posterior Sampling.
➢ Crucial advantage of Posterior Sampling: UCB decomposition depends on the upper confidence bounds that are explicitly constructed and used. Posterior Sampling bound depends on the best possible choice of confidence bounds.

New Theoretical Guarantees

Because of the Risk Decomposition, existing analysis that provides bounds for specific UCB algorithms immediately gives new bounds on the Bayes Risk of PS

Model	Bayes Risk Bound: (Up to Log-Factors)
Any Finite Action Space $\lvert\mathcal{A}\rvert = K$	$\sqrt{KT}$
Linear Model: $f_\theta(a) = \phi(a)^T\theta$, for known feature vector $\phi(a) \in \mathbb{R}^d$.	$d\sqrt{T}$
Generalized Linear Model: $f_\theta(a) = g(\phi(a)^T\theta)$, , for known feature vector $\phi(a) \in \mathbb{R}^d$ and function g	$d\sqrt{T}$
Sparse Linear Model $f_\theta(a) = \phi(a)^T\theta$, and we expect θ is sparse.	$\mathbb{E}\sqrt{d\lVert\theta\rVert_0 T}$
Gaussian Process $\{f_\theta(a)\lvert a \in \mathcal{A}\}$ sampled from a GP.	$\sqrt{\gamma_T[\log\lvert\mathcal{A}\rvert]T}$ • γ_T:= Maximum T period information gain about $\{f_\theta(a)\lvert a \in \mathcal{A}\}$ • Extensions to infinite $\mathcal{A}$

Bound for a general class of functions

➢Goal: Give a unified analysis of many problems, and provide a bound that depends on the complexity of the class of functions $\mathcal{F} = \{f_\theta: \theta \in \Theta\}$.

"Theorem": *Approx. Less than* ↓

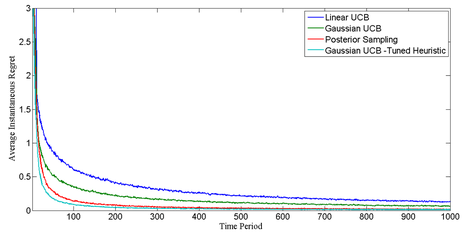
$$\text{BayesRisk}(T) \lesssim \sqrt{\text{Dim}_K(\mathcal{F})\text{Dim}_M(\mathcal{F}, T^{-1})T}$$

$\text{Dim}_K(\mathcal{F})$ = Kolmogorov Dimension. Roughly captures sensitivity to statistical over-fitting.
$\text{Dim}_M(\mathcal{F})$ = Margin Dimension - A new notion that measures the degree of dependence among rewards generated by different actions.

When $\{f_\theta: \theta \in \Theta\}$ is a class of linear or generalized linear models this matches the best bounds available for a UCB algorithm. Hence, this generalizes results on linear and generalized linear bandits.

What is Margin Dimension?
Def: $\text{Dim}_M(\mathcal{F}, \epsilon)$ is the length of the longest sequence in $\mathcal{A}$ such that each action is ϵ −independent of its predecessors.
Def: An action $a \in \mathcal{A}$ is ϵ −dependent on $\{a_1, .., a_n\}$ with respect to $\mathcal{F}$ if any two functions $f, \tilde{f} \in \mathcal{F}$ satisfying $\sqrt{\sum_1^n (f - \tilde{f})^2(a_i)} \le \epsilon$ satisfy $\lvert f(a) - \tilde{f}(a)\rvert \le \epsilon$

Simulation With Gaussian Linear Model



➢ $f_\theta(a) = \phi(a)^T\theta$, for a known feature vector $\phi(a) \in \mathbb{R}^d$
➢Gaussian Prior ($\theta \sim N(\mu_0, \Sigma_0)$) and Gaussian noise.
➢Simulation trial with 100 actions with randomly drawn feature vectors and $d = 10$.
➢ Results averaged across 5000 trials.

Conclusion: posterior sampling outperforms the best UCB algorithms in the literature ([1] and [12]), but in this simple setting a UCB algorithm that is tuned to the time horizon outperforms them all.

Some References

1. Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved Algorithms for Linear Stochastic Bandits. In NIPS, pages 2312–2320, 2011.
2. Y. Abbasi-yadkori, D. Pal, and C. Szepesvari. Online to-confidence-set conversions and application to sparse stochastic bandits. In Artificial Intelligence and Statistics, 2012.
3. P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite time analysis of the multiarmed bandit problem. Machine Learning, 47(2-3):235–256, 2002.
4. Varsha Dani, Thomas P. Hayes, and Sham M. Kakade. Stochastic Linear Optimization under Bandit Feedback. In COLT, pages 355–366, 2008.
5. Sarah Filippi, Olivier Cappé, Aurélien Garivier, and Csaba Szepesvári. Parametric Bandits: The Generalized Linear Case. In NIPS, pages 586–594, 2010.
6. E. Kaufmann, N. Korda, and R. Munos. Thompson sampling: An optimal finite time analysis. arXiv preprint arXiv:1205.4217, 2012.
7. L. Li and O. Chapelle.. An empirical evaluation of Thompson sampling. In Neural Information Processing Systems (NIPS), 2011.
8. L. Li and O. Chapelle. Open problem: Regret bounds for Thompson sampling. In Proceedings of the 25th Annual Conference on Learning Theory (COLT), 2012.
9. Paat Rusmevichientong and John N. Tsitsiklis. Linearly parameterized bandits. Mathematics of Operations Research, 35(2):395–411, 2010.
10. N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In Proc. International Conference on Machine Learning (ICML), 2010.