Bayesian and Bandit Optimisation

Matt Hoffman and Rowan McAllister

MLG RCC

24 October 2013

Introduction

Bayesian Optimisation

Optimisation Task

Goal: find maximum of function $f(\cdot)$ over bounded set $\mathcal{X}$:

$$\max_{\mathbf{x}\in\mathcal{X}\subset\mathbb{R}^d} f(\mathbf{x})$$

Challenges:

- unknown f, but $y \sim p(\cdot|\mathbf{x})$ $s.t.$ $f(\mathbf{x}) = \mathbb{E}[y|\mathbf{x}]$
- sampling y is *expensive*,
- samples provides no $\frac{\mathrm{d}f(x)}{\mathrm{d}x}$ information,
- unsure if f is nonlinear, convex.

slides borrow many ideas from [Brochu 2010], [Munos 2012], [Hoffman 2005].

Example Scenarios

Examples where sampling objective function $f(\cdot)$ is expensive:

- sequential decision problems requiring many long-horizon simulations - e.g. adversarial games and reinforcement learning,
- drug trials,
- active user modelling: avoid asking the human an unreasonable amount of questions

Idea

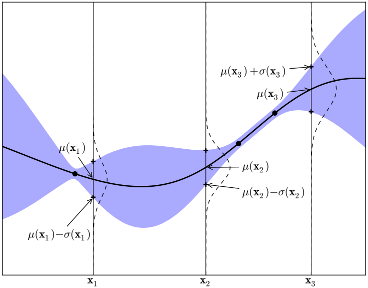
- Keep track of what we expect $f(\mathbf{x})$ to be $\forall\,\mathbf{x}$,
- and how certain we are of $f(\mathbf{x})\,\forall\,\mathbf{x}$.
- Use this to guide a strategic sampling strategy, given a small 'sample-budget'.

Bayesian Belief Monitoring

Combine prior belief of plausible functions $p(f)$, with evidence from previous samples $\mathcal{D}_{1:t}$, to maintain a posterior belief:

$$
\begin{aligned}
p(f|\mathcal{D}_{1:t}) &\propto p(f)p(\mathcal{D}_{1:t}|f), \\
&= p(f|\mathcal{D}_{1:t-1})p(\mathcal{D}_t|f)
\end{aligned}
$$

The posterior is used to decide where to sample next: $\mathbf{x}_{t+1}$.



[Brochu 2010]

Introduction

Bandits

Likelihood / Generating Distribution

Discrete version of problem. Consider K slot machines:

$\cdots$

$\mathcal{X}$:	1	2		K
$p(y\|\theta)$:	$Ber(\theta_1)$	$Ber(\theta_2)$	$\cdots$	$Ber(\theta_K)$
$f(\cdot)$	θ_1	θ_2		θ_K

- You have T tokens.
- The i'th machine returns \$0 or \$1 based on a fixed yet unknown probability $\theta_i \in [0, 1]$.
- Objective: maximise winnings.

Bernoulli Bandit example

1

2

As you play, you record what you see. Your current record is:

	wins	losses
Bandit 1:	1	2
Bandit 2:	10	9

Q: Which machine would you play next if you have 1 token remaining?

Bernoulli Bandit example

1 2

As you play, you record what you see. Your current record is:

	wins	losses
Bandit 1:	1	2
Bandit 2:	10	9

Q: Which machine would you play next if you have 1 token remaining?

Q: How about if you have 100 tokens remaining?

Bandits: how to solve?

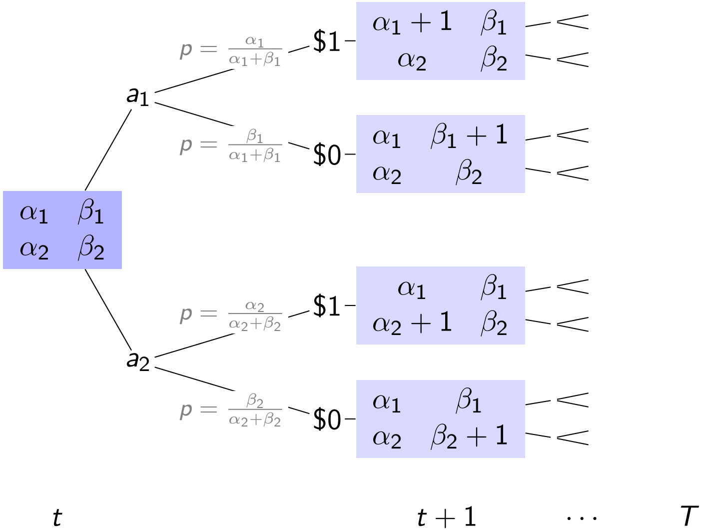
Treat as a sequential decision making problem in an MDP environment:

- Action space $\mathcal{X}$ = bandit choice $\{1, ..., K\}$
- State space $\mathcal{S}$ = sufficient statistics of record (*information states*)
- horizon = T tokens
- objective = $\mathbb{E}[\sum_{t=1}^{T} \gamma^{t-1} r_t | s_t]$

Solve with a probabilistic model of each bandit then simulate potential futures using:

- Dynamic Programming
- Tree search

Bernoulli Bandits: example tree solution



Bernoulli Bandits: tree solution pros/cons

Pro:

- Computes 'Bayes-optimal' action-values / acquisition function $\rightarrow$ optimal policy w.r.t beliefs and horizon.

Con:

- Computation time: $\mathcal{O}((2K)^T)$.

This is a general solution. When points are independent, we can compute faster!

Gittins Index

Gittins Index: Introduction

What is it?

- Gittins Index (like tree solution) is the Bayes-optimal acquisition function for the bandit problem.
- Difference: exploits independence of bandits' unknown expected return f to compute same policy faster!

for $t = 1 : T$

1. compute Gittins Index of all K bandits,
2. sample / play bandit of greatest Index,
3. observe reward-outcomes and update belief-posterior.

Gittins Index: Intuition

Consider a Bernoulli bandit of unknown p, and a deterministic bandit that returns known reward $\$\nu$ each play.

$p(\theta_i) \sim Beta(\alpha_i, \beta_i)$

$\$\nu$

Sampling policy solution: perhaps use a tree like before.

Q: Which value of ν makes us indifferent to deciding next-bandit?

Gittins Index: Intuition

Consider a Bernoulli bandit of unknown p, and a deterministic bandit that returns known reward $\$\nu$ each play.

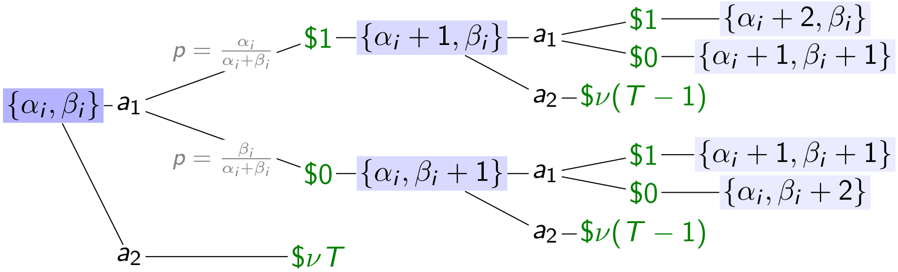
$p(\theta_i) \sim Beta(\alpha_i, \beta_i)$ $\$\nu$

Sampling policy solution: perhaps use a tree like before.

Q: Which value of ν makes us indifferent to deciding next-bandit?

...that is our Gittins Index for bandit i.

Computing Gittins Indices (1)



Computing Gittins Indices (2)

Value of stochastic bandit:

$$V(i,\nu) = \sup_{\tau>1}\{\mathbb{E}[\sum_{t=1}^{\tau-1}\gamma^{t-1}r_t|\alpha_i,\beta_i] + \nu\sum_{t=\tau}^{T}\gamma^{t-1}\}$$

Advantage of stochastic bandit:

$$\begin{aligned} D(i,\nu) &= V(i,\nu) - \nu\cdot\frac{1-\gamma^T}{1-\gamma}, \\ &= \sup_{\tau>1}\{\mathbb{E}[\sum_{t=1}^{\tau-1}\gamma^{t-1}(r_t-\nu)|\alpha_i,\beta_i]\}, \end{aligned}$$

Solving Gittins Index ν_i:

$$\nu_i = \{\nu : D(i,\nu) = 0\}$$

using dynamic programming, computational complexity only $\mathcal{O}(KT^2)$.

Heuristic Acquisition Functions

Bayesian Optimisation Algorithm

Define the *acquisition function* $u(\mathbf{x})$, the 'utility of sampling' at point $\mathbf{x}$. It trades off information gain vs. high-expectation of f.

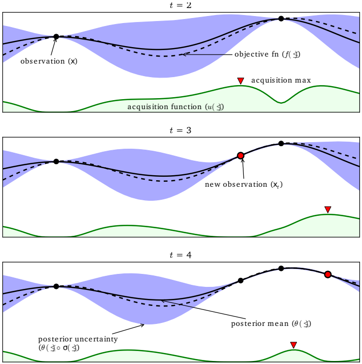
for $t = 1 : T$

1. find test-point of greatest sampling utility: $\mathbf{x}_t = \arg\max_{\mathbf{x} \in \mathcal{X}} u(\mathbf{x}|\mathcal{D}_{1:t-1})$,
2. sample y_t
3. update belief: $P(f|\mathcal{D}_{1:t}) \propto P(f|\mathcal{D}_{1:t-1})P(\mathcal{D}_t|f)$.

What's a Good Acquisition Function $u(\mathbf{x})$?

Want $u(\mathbf{x})$ to:

- consider mean $\mu(\mathbf{x})$ *and* uncertainty $\sigma(\mathbf{x})$ of posterior belief,
- monotonically increase with μ (exploitatory value),
- monotonically increase with σ (exploratory value),
- trade off exploration and exploitation gains somehow



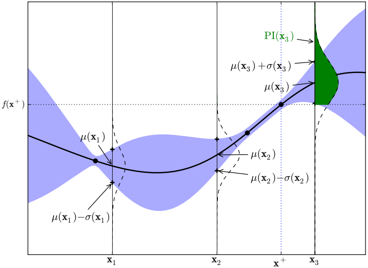
Probability of Improvement (PI)

$$PI(\mathbf{x}) \quad = \quad P(f(\mathbf{x}) > f(\mathbf{x}^+) + \xi | \mathcal{D}_{1:t})$$

where $\mathbf{x}^+ = \arg\max_{\mathbf{x} \in \mathbf{x}_{1:t}} \mathbb{E}[f(\mathbf{x})]$.

$\xi \geq 0$ determines minimum improvements on $\mathbb{E}[f(\mathbf{x}^+)]$ we'd consider.

Low $\xi \rightarrow$ exploitative, high $\xi \rightarrow$ explorative.



[Brochu 2010]

Expected Improvement (EI)

EI takes into account probability *and* amount of improvement.

$$
\begin{aligned}
EI(\mathbf{x}) &= \mathbb{E}[max\{0, f(\mathbf{x}) - f(\mathbf{x}^+) - \xi\} | \mathcal{D}_{1:t}] \\
&= \int_{f(\mathbf{x}^+)}^{\infty} (y - f(\mathbf{x}^+) - \xi) p(f(\mathbf{x}) = y | \mathcal{D}_{1:t}) \mathrm{d}y
\end{aligned}
$$

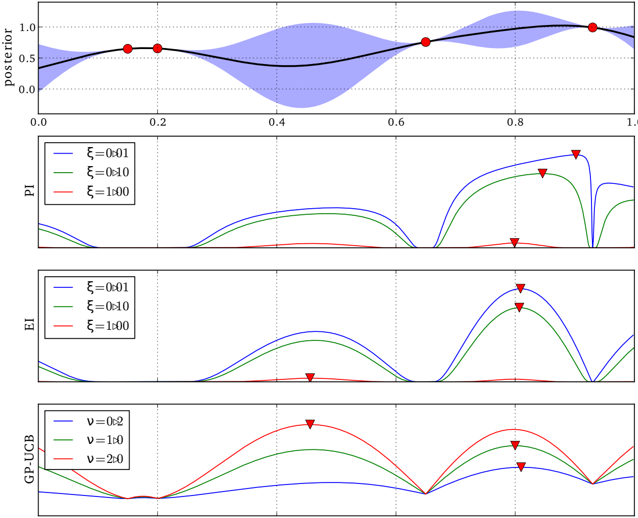
GP Upper Confidence Bound (GP-UCB)

$$UCB(\mathbf{x}) = \mu(\mathbf{x}) + \kappa\sigma(\mathbf{x})$$

$GP\text{-}UCB(\mathbf{x})$:

$$\kappa_t = \mathcal{O}(\sqrt{\log t})$$

Acquisition Function Comparisons



[Brochu 2010]

Extra Slides

Notation

d	dimensionality of $\mathcal{X}$
T	time horizon
$\mathbf{x}^+$	$\arg\max_{\mathbf{x}\in\mathbf{x}_{1:t}} f(\mathbf{x})$
$\mathbf{x}^*$	$\arg\max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$
$\mu(\mathbf{x})$	mean of $p(f\|\mathcal{D}_{1:t})$ at $\mathbf{x}$
$\sigma^2(\mathbf{x})$	variance of $p(f\|\mathcal{D}_{1:t})$ at $\mathbf{x}$
$\phi(\cdot)$	normal probability density function
$\Phi(\cdot)$	normal cumulative distribution function

Performance Measures

Simple Regret

$r_T = f(\mathbf{x}^*) - f(\mathbf{x}_T)$

Cumulative Regret

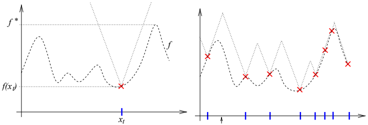
$R_T = \sum_{t=1}^{T} f(\mathbf{x}^*) - f(\mathbf{x}_t)$

where $\mathbf{x}^* = \arg\max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$ (unknown to agent)

Lipschitz-Continuity

f is Lipschitz-continuous, $\forall\, \mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}$, $\exists\, C < \infty$ s.t.:

$$||f(\mathbf{x}_i) - f(\mathbf{x}_j)|| \leq C||\mathbf{x}_i - \mathbf{x}_j||$$



[Munos 2012]

Expected Improvement (EI) - details

EI takes into account probability *and* amount of improvement.

$$
\begin{aligned}
EI(\mathbf{x}) &= \mathbb{E}[max\{0, f(\mathbf{x}) - f(\mathbf{x}^+)\}|\mathcal{D}_{1:t}] \\
&= \int_{f(\mathbf{x}^+)}^{\infty} (y - f(\mathbf{x}^+)p(f(\mathbf{x}) = y|\mathcal{D}_{1:t})\mathrm{d}y \\
&= \begin{cases} (\mu(\mathbf{x}) - f(\mathbf{x}^+))\Phi(z) + \sigma(\mathbf{x})\sigma(z), & \text{if } \sigma(\mathbf{x}) > 0 \\ 0, & \text{if } \sigma(\mathbf{x}) = 0 \end{cases} \\
z &= \frac{\mu(\mathbf{x}) - f(\mathbf{x}^+)}{\sigma(\mathbf{x})}
\end{aligned}
$$

GP Upper Confidence Bound (GP-UCB) - details

$$UCB(\mathbf{x}) = \mu(\mathbf{x}) + \kappa\sigma(\mathbf{x})$$

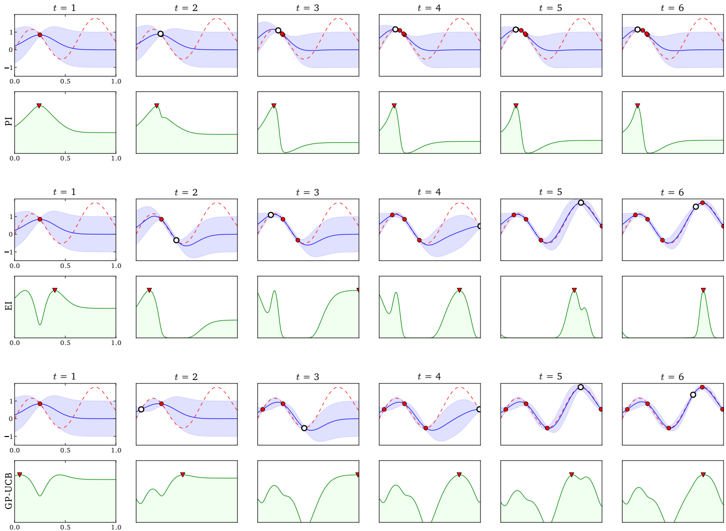
$GP\text{-}UCB(\mathbf{x})$: set $\kappa_t^2 = 2\log(t^2 2\pi^2/(3\delta)) + 2d\log(t^2 dbr\sqrt{\log(4da/\delta)})$,
then $P(R_T \leq \sqrt{8T\kappa_T\gamma_T/\log(1+\sigma^{-2})} + 2 \;\; \forall T \geq 1) \geq 1 - \delta$.
[Srinivas 2010]

where:

- δ is user selected $\in (0, 1)$,
- r is the length of space $\mathcal{X}$ in each dimension.
- γ_T is maximum information gain on $f(\cdot)$ by time T,
- a and b define probabilistic derivative bounds on GP-samples of f: $P(\max_{x\in\mathcal{X}} |\partial f/\partial x_j| > L) \leq ae^{-(L/b)^2}$.

$GP\text{-}UCB(\mathbf{x})$ achieves *no-regret*: $\lim_{n\to\infty} R_T/T = 0$ with high probability.

Acquisition Function Comparisons (2)



[Brochu 2010]