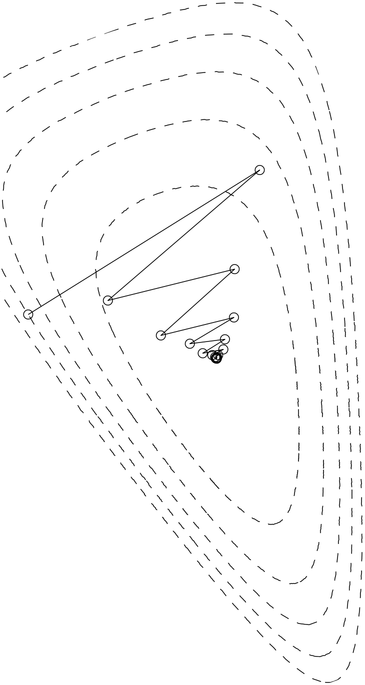


Introduction to Optimization

Global & Bayesian Optimization

Multi-armed bandits, exploration vs. exploitation, navigation through belief space, upper confidence bound (UCB), global optimization = infinite bandits, Gaussian Processes, probability of improvement, expected improvement, UCB

Marc Toussaint
University of Stuttgart
Summer 2015

Global Optimization

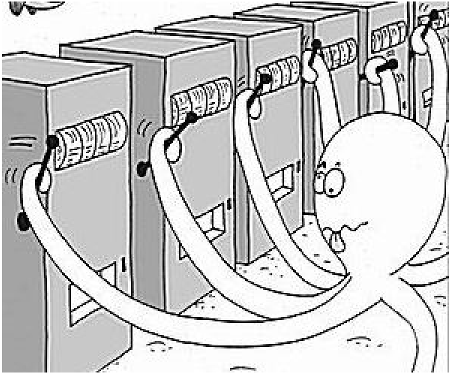
- Is there an optimal way to optimize (in the Blackbox case)?
- Is there a way to find the *global* optimum instead of only local?

Outline

- Play a game
- Multi-armed bandits
  - Belief state & belief planning
  - Upper Confidence Bound (UCB)
- Optimization as infinite bandits
  - GPs as belief state
- Standard heuristics:
  - Upper Confidence Bound (GP-UCB)
  - Maximal Probability of Improvement (MPI)
  - Expected Improvement (EI)

Bandits

Bandits



- There are n machines.
- Each machine i returns a reward $y \sim P(y; \theta_i)$
  The machine's parameter θ_i is unknown

Bandits

- Let $a_t \in \{1, .., n\}$ be the choice of machine at time t
  Let $y_t \in \mathbb{R}$ be the outcome with mean $\langle y_{a_t} \rangle$

- A policy or strategy maps all the history to a new choice:

$$\pi : \ [(a_1, y_1), (a_2, y_2), ..., (a_{t\text{-}1}, y_{t\text{-}1})] \mapsto a_t$$

- Problem: Find a policy π that

$$\max \left\langle \sum_{t=1}^T y_t \right\rangle$$

  or

$$\max \langle y_T \rangle$$

  or other objectives like discounted infinite horizon $\max \left\langle \sum_{t=1}^\infty \gamma^t y_t \right\rangle$

Exploration, Exploitation

- "Two effects" of choosing a machine:
  - You collect more data about the machine $\rightarrow$ knowledge
  - You collect reward

- For example
  - Exploration: Choose the next action a_t to $\min \langle H(b_t) \rangle$
  - Exploitation: Choose the next action a_t to $\max \langle y_t \rangle$

The Belief State

- "Knowledge" can be represented in two ways:
  - as the full history

$$h_t \; = \; [(a_1, y_1), (a_2, y_2), ..., (a_{t\text{-}1}, y_{t\text{-}1})]$$

  - as the **belief**

$$b_t(\theta) = P(\theta | h_t)$$

    where θ are the unknown parameters $\theta = (\theta_1, .., \theta_n)$ of all machines

- In the bandit case:
  - The belief factorizes $b_t(\theta) = P(\theta|h_t) = \prod_i b_t(\theta_i|h_t)$
    e.g. for Gaussian bandits with constant noise, $\theta_i = \mu_i$

$$b_t(\mu_i|h_t) = \mathcal{N}(\mu_i|\hat y_i, \hat s_i)$$

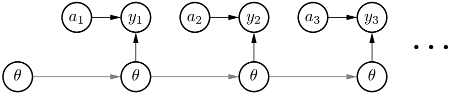
    e.g. for binary bandits, $\theta_i = p_i$, with prior $\text{Beta}(p_i|\alpha, \beta)$:

$$b_t(p_i|h_t) = \text{Beta}(p_i|\alpha + a_{i,t}, \beta + b_{i,t})$$

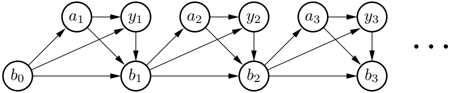
$$a_{i,t} = \sum_{s=1}^{t-1} [a_s = i][y_s = 0] \; , \quad b_{i,t} = \sum_{s=1}^{t-1} [a_s = i][y_s = 1]$$

The Belief MDP

- The process can be modelled as



  or as Belief MDP



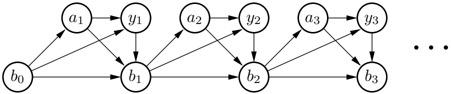
$$P(b'|y,a,b) = \begin{cases} 1 & \text{if } b' = b'_{[b,a,y]} \\ 0 & \text{otherwise} \end{cases} \quad , \quad P(y|a,b) = \int_{\theta_a} b(\theta_a)\ P(y|\theta_a)$$

- The Belief MDP describes a *different* process: the interaction between the information available to the agent (b_t or h_t) and its actions, where *the agent uses his current belief to anticipate outcomes,* $P(y|a,b)$*.*
- The belief (or history h_t) is all the information the agent has avaiable; $P(y|a,b)$ the “best” possible anticipation of observations. If it acts optimally in the Belief MDP, it acts optimally in the original problem.

  Optimality in the Belief MDP $\Rightarrow$ *optimality in the original problem*

Optimal policies via Belief Planning

- The Belief MDP:



$$P(b'|y,a,b) = \begin{cases} 1 & \text{if } b' = b'_{[b,a,y]} \\ 0 & \text{otherwise} \end{cases} , \quad P(y|a,b) = \int_{\theta_a} b(\theta_a)\ P(y|\theta_a)$$

- Belief Planning: Dynamic Programming on the value function

$$\begin{aligned} \forall_b :\ V_{t\text{-}1}(b) &= \max_\pi \left\langle \textstyle\sum_{t=t}^T y_t \right\rangle \\ &= \max_\pi \Big[\langle y_t \rangle + \left\langle \textstyle\sum_{t=t+1}^T y_t \right\rangle \Big] \\ &= \max_{a_t} \int_{y_t} P(y_t|a_t,b)\ \Big[y_t + V_t(b'_{[b,a_t,y_t]}) \Big] \end{aligned}$$

Optimal policies

- The value function assigns a value (maximal achievable return) to a state of knowledge
- The optimal policy is greedy w.r.t. the value function (in the sense of the $\max_{a_t}$ above)
- Computationally heavy: b_t is a probability distribution, V_t a function over probability distributions

- The term $\int_{y_t} P(y_t|a_t, b_{t\text{-}1}) \left[y_t + V_t(b_{t\text{-}1}[a_t, y_t])\right]$ is related to the *Gittins Index*: it can be computed for each bandit separately.

Example exercise

- Consider 3 binary bandits for $T = 10$.
  - The belief is 3 Beta distributions $\text{Beta}(p_i | \alpha + a_i, \beta + b_i) \quad \rightarrow \quad$ 6 integers
  - $T = 10 \quad \rightarrow \quad$ each integer ≤ 10
  - $V_t(b_t)$ is a function over $\{0, .., 10\}^6$

- Given a prior $\alpha = \beta = 1$,
  a) compute the optimal value function and policy for the final reward and the average reward problems,
  b) compare with the UCB policy.

Greedy heuristic: Upper Confidence Bound (UCB)

Initializaiton: Play each machine once
repeat
    Play the machine i that maximizes $\hat{y}_i + \beta\sqrt{\frac{2\ln n}{n_i}}$
until

$\hat{y}_i$ is the average reward of machine i so far
n_i is how often machine i has been played so far
$n = \sum_i n_i$ is the number of rounds so far
β is often chosen as $\beta = 1$

See *Finite-time analysis of the multiarmed bandit problem*, Auer, Cesa-Bianchi & Fischer, Machine learning, 2002.

UCB algorithms

- UCB algorithms determine a **confidence interval** such that

$$\hat{y}_i - \sigma_i < \langle y_i \rangle < \hat{y}_i + \sigma_i$$

  with high probability.
  UCB chooses the upper bound of this confidence interval

- *Optimism in the face of uncertainty*

- Strong bounds on the regret (sub-optimality) of UCB (e.g. Auer et al.)

Conclusions

- The bandit problem is an archetype for
  - Sequential decision making
  - Decisions that influence knowledge as well as rewards/states
  - Exploration/exploitation

- The same aspects are inherent also in global optimization, active learning & RL

- Belief Planning in principle gives the optimal solution

- Greedy Heuristics (UCB) are computationally much more efficient and guarantee bounded regret

Further reading

- ICML 2011 Tutorial *Introduction to Bandits: Algorithms and Theory*, Jean-Yves Audibert, Rémi Munos
- *Finite-time analysis of the multiarmed bandit problem*, Auer, Cesa-Bianchi & Fischer, Machine learning, 2002.
- *On the Gittins Index for Multiarmed Bandits*, Richard Weber, Annals of Applied Probability, 1992.
  Optimal Value function is submodular.

Global Optimization

Global Optimization

- Let $x \in \mathbb{R}^n$, $f:\ \mathbb{R}^n \to \mathbb{R}$,   find

$$\min_x\ f(x)$$

(I neglect constraints $g(x) \le 0$ and $h(x) = 0$ here – but could be included.)

- Blackbox optimization: find optimium by sampling values $y_t = f(x_t)$
  No access to ∇f or $\nabla^2 f$
  Observations may be noisy $y \sim \mathcal{N}(y \,|\, f(x_t), \sigma)$

Global Optimization $=$ infinite bandits

- In global optimization $f(x)$ defines a reward for every $x \in \mathbb{R}^n$
  – Instead of a finite number of actions a_t we now have x_t

- Optimal Optimization could be defined as: find $\pi : h_t \mapsto x_t$ that

$$\min \left\langle \sum_{t=1}^{T} f(x_t) \right\rangle$$

or

$$\min \langle f(x_T) \rangle$$

Gaussian Processes as belief

- The unknown "world property" is the function $\theta = f$
- Given a Gaussian Process prior $GP(f|\mu, C)$ over f and a history

$$D_t \;=\; [(x_1, y_1), (x_2, y_2), ..., (x_{t\text{-}1}, y_{t\text{-}1})]$$

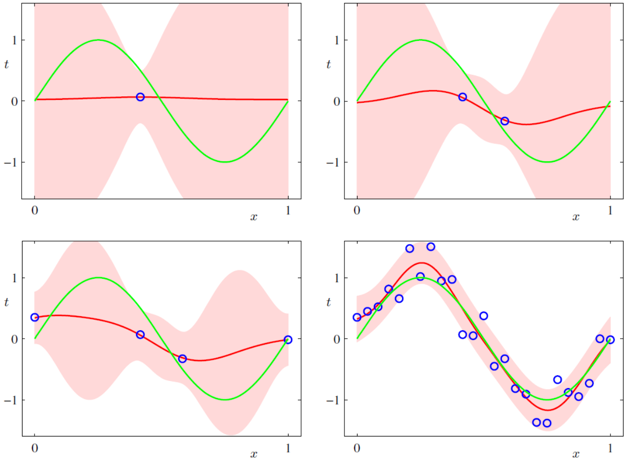
the belief is

$$b_t(f) = P(f \mid D_t) = \mathsf{GP}(f|D_t, \mu, C)$$

$$\mathsf{Mean}(f(x)) = \hat f(x) = \boldsymbol{\kappa}(x)(\boldsymbol{K} + \sigma^2\mathbf{I})^{\text{-}1}\boldsymbol{y} \qquad \textit{response surface}$$

$$\mathsf{Var}(f(x)) = \hat\sigma(x) = k(x,x) - \boldsymbol{\kappa}(x)(\boldsymbol{K} + \sigma^2\mathbf{I}_n)^{\text{-}1}\boldsymbol{\kappa}(x) \qquad \textit{confidence interval}$$

- Side notes:
  - Don't forget that $\mathsf{Var}(y^*|x^*, D) = \sigma^2 + \mathsf{Var}(f(x^*)|D)$
  - We can also handle discrete-valued functions f using GP classification


t
x
0
1
−1

Optimal optimization via belief planning

- As for bandits it holds

$$\begin{aligned} V_{t\text{-}1}(b_{t\text{-}1}) &= \max_\pi \Big\langle \textstyle\sum_{t=t}^T y_t \Big\rangle \\ &= \max_{x_t} \textstyle\int_{y_t} P(y_t|x_t, b_{t\text{-}1}) \ \Big[y_t + V_t(b_{t\text{-}1}[x_t, y_t])\Big] \end{aligned}$$

  $V_{t\text{-}1}(b_{t\text{-}1})$ is a function over the GP-belief!
  If we could compute $V_{t\text{-}1}(b_{t\text{-}1})$ we "optimally optimize"

- I don't know of a minimalistic case where this might be feasible

Conclusions

- Optimization as a problem of
  - Computation of the belief
  - Belief planning

- Crucial in all of this: **the prior** $P(f)$
  - GP prior: smoothness; but also limited: only local correlations!
    No "discovery" of non-local/structural correlations through the space
  - The latter would require different priors, e.g. over different function classes

Heuristics

1-step heuristics based on GPs

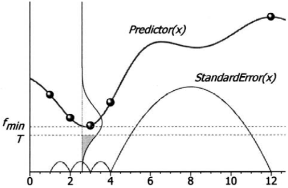


Figure 14. Using kriging, we can estimate the probability that sampling at a given point will 'improve' our solution, in the sense of yielding a value that is equal or better than some target T.

from Jones (2001)

- Maximize Probability of Improvement (MPI)

$$x_t = \underset{x}{\operatorname{argmax}} \int_{-\infty}^{y^*} \mathcal{N}(y|\hat{f}(x), \hat{\sigma}(x))$$

- Maximize Expected Improvement (EI)

$$x_t = \underset{x}{\operatorname{argmax}} \int_{-\infty}^{y^*} \mathcal{N}(y|\hat{f}(x), \hat{\sigma}(x))\ (y^* - y)$$

- Maximize UCB

$$x_t = \underset{x}{\operatorname{argmax}}\ \hat{f}(x) + \beta_t \hat{\sigma}(x)$$

(Often, $\beta_t = 1$ is chosen. UCB theory allows for better choices. See Srinivas et al. citation below.)

Each step requires solving an optimization problem

- Note: each argmax on the previous slide is an optimization problem
- As $\hat{f}, \hat{\sigma}$ are given analytically, we have gradients and Hessians. BUT: multi-modal problem.
- In practice:
  - Many restarts of gradient/2nd-order optimization runs
  - Restarts from a grid; from many random points

- We put a lot of effort into carefully selecting just the next query point

From: *Information-theoretic regret bounds for gaussian process optimization in the bandit setting* Srinivas, Krause, Kakade & Seeger, Information Theory, 2012.

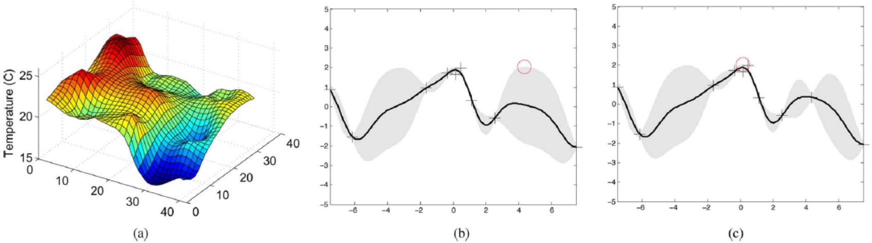


Fig. 2. (a) Example of temperature data collected by a network of 46 sensors at Intel Research Berkeley. (b) and (c) Two iterations of the GP-UCB algorithm. The dark curve indicates the current posterior mean, while the gray bands represent the upper and lower confidence bounds which contain the function with high probability. The "+" mark indicates points that have been sampled before, while the "o" mark shows the point chosen by the GP-UCB algorithm to sample next. It samples points that are either (b) uncertain or have (c) high posterior mean.

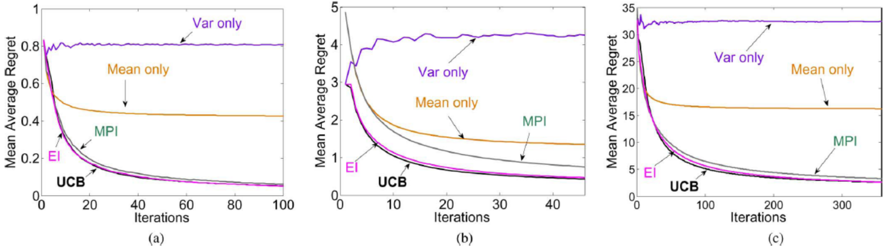


Fig. 6. Mean average regret: GP-UCB and various heuristics on (a) synthetic and (b, c) sensor network data.

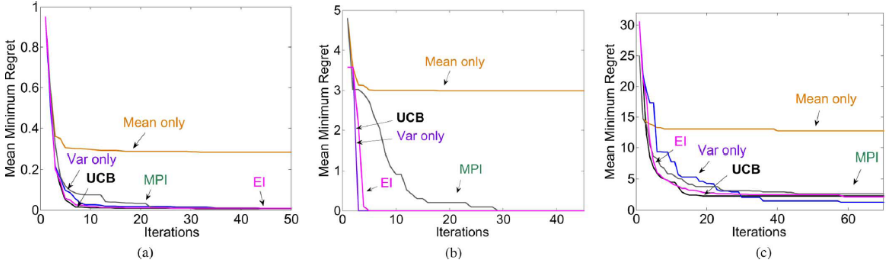


Fig. 7. Mean minimum regret: GP-UCB and various heuristics on (a) synthetic, and (b, c) sensor network data.

Pitfall of this approach

- A real issue, in my view, is the choice of kernel (i.e. prior $P(f)$)
  - 'small' kernel: almost exhaustive search
  - 'wide' kernel: miss local optima
  - adapting/choosing kernel online (with CV): might fail
  - real f might be non-stationary
  - non RBF kernels? Too strong prior, strange extrapolation

- Assuming that we have the right prior $P(f)$ is really a strong assumption

Further reading

- Classically, such methods are known as *Kriging*

- *Information-theoretic regret bounds for gaussian process optimization in the bandit setting* Srinivas, Krause, Kakade & Seeger, Information Theory, 2012.

- *Efficient global optimization of expensive black-box functions.* Jones, Schonlau, & Welch, Journal of Global Optimization, 1998.
- *A taxonomy of global optimization methods based on response surfaces* Jones, Journal of Global Optimization, 2001.
- *Explicit local models: Towards optimal optimization algorithms*, Poland, Technical Report No. IDSIA-09-04, 2004.

Entropy Search

slides by Philipp Hennig

P. Hennig & C. Schuler: *Entropy Search for Information-Efficient Global Optimization*, JMLR 13 (2012).

Predictive Entropy Search

- Hernández-Lobato, Hoffman & Ghahraman: *Predictive Entropy Search for Efficient Global Optimization of Black-box Functions*, NIPS 2014.
- Also for constraints!
- Code: `https://github.com/HIPS/Spearmint/`