Bayesian optimization for automatic machine learning

Matthew W. Hoffman

based off work with J. M. Hernández-Lobato, M. Gelbart, B. Shahriari, and others!

University of Cambridge

July 11, 2015

UNIVERSITY OF CAMBRIDGE

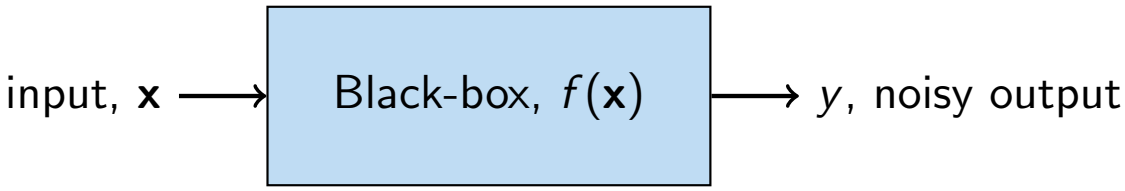
Black-box optimization

I'm interested in solving black-box optimization problems of the form

$$\mathbf{x}_\star = \arg\max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$$

where black-box means:

- we may only be able to observe the function value, i.e. **no gradients**
- our observations may be **corrupted by noise**



- optimization involves designing a sequential strategy which maps collected data to the next query point

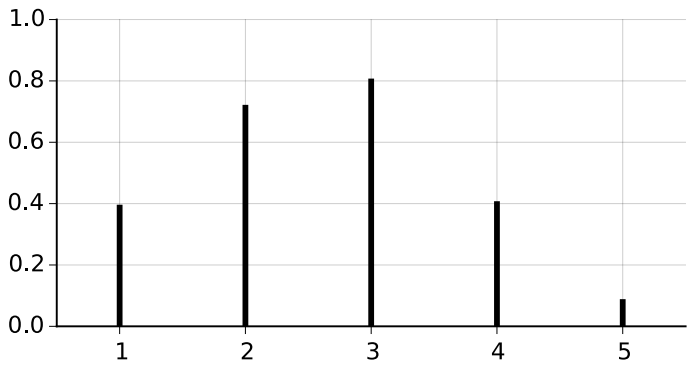
Example (AB testing)

Users visit our website which has different configurations (A and B) and we want to find the **best configuration** to optimize clicks, revenue, etc.

Example (Hyperparameter tuning)

A Machine Learning algorithm may rely on hard-to-tune **hyperparameters** which we want to optimize wrt. some test-set accuracy.

Note that I haven't said the word **Bayesian** yet. . .

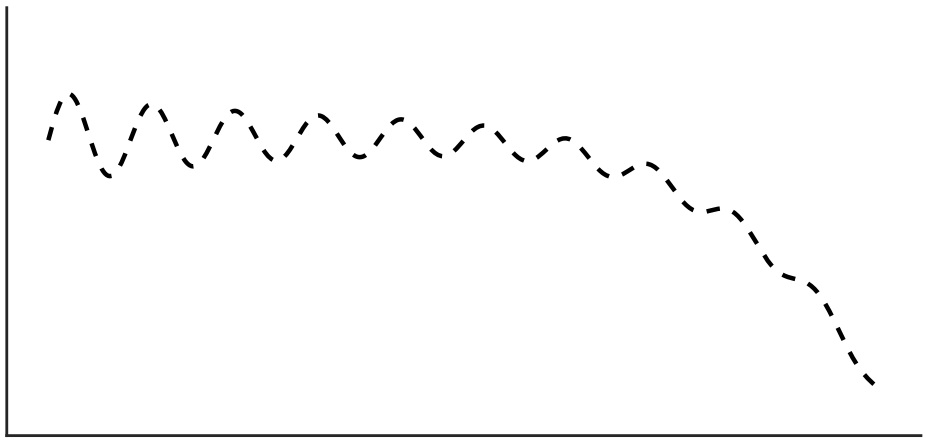


Consider a function defined over finite indices with Bernoulli observations given by $f(i)$. This is a classic **bandit problem**.

Often bandit settings involve **cumulative rewards** but there is a growing deal of literature on **best arm identification**

- UCBE [Audibert and Bubeck, 2010]
- UGap [Gabillon et al., 2012]
- BayesGap **[Hoffman et al., 2014]**
- in linear bandits [Soare et al., 2014]
- explicitly for optimization as in SOO [Munos, 2011]
- and many others [Kaufmann et al., 2014]

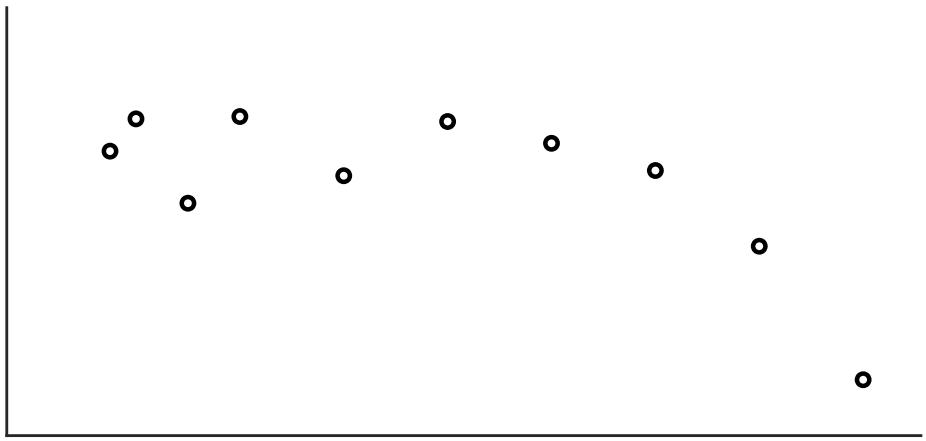
Bayesian black-box optimization



Bayesian optimization in a nutshell:

Močkus et al. [1978], Jones et al. [1998], Jones [2001]

Bayesian black-box optimization

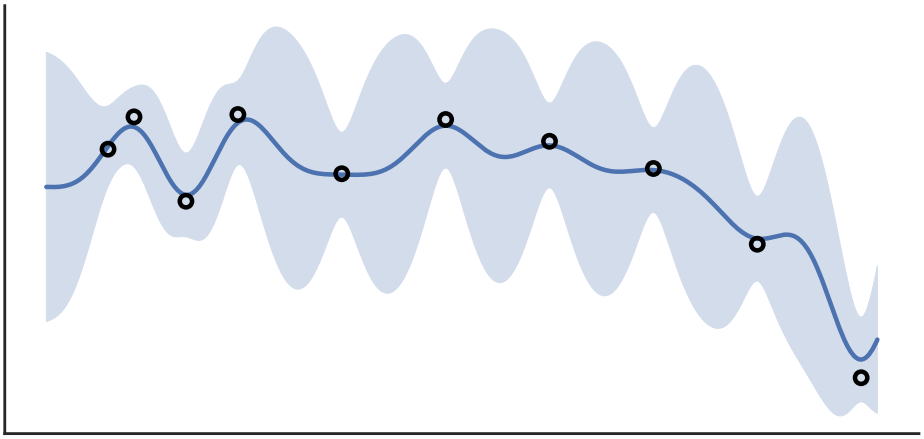


Bayesian optimization in a nutshell:

1. **initial sample**

Mockus et al. [1978], Jones et al. [1998], Jones [2001]

Bayesian black-box optimization

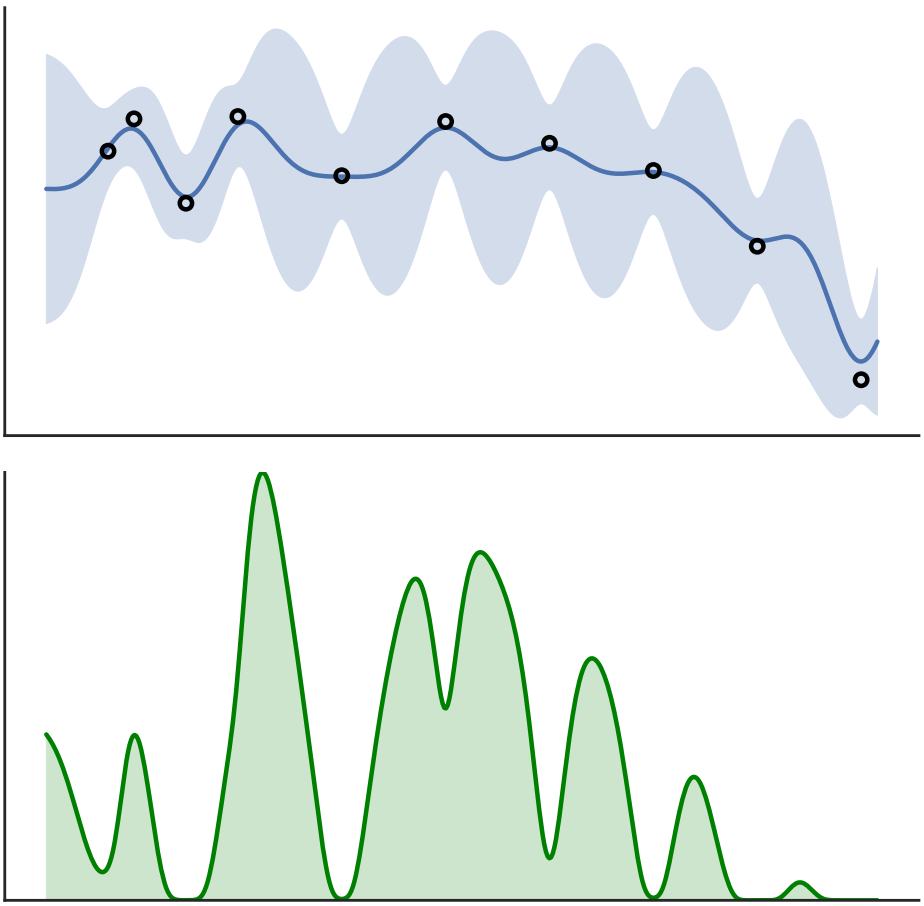


Bayesian optimization in a nutshell:

1. initial sample
2. **construct a posterior model**

Močkus et al. [1978], Jones et al. [1998], Jones [2001]

Bayesian black-box optimization

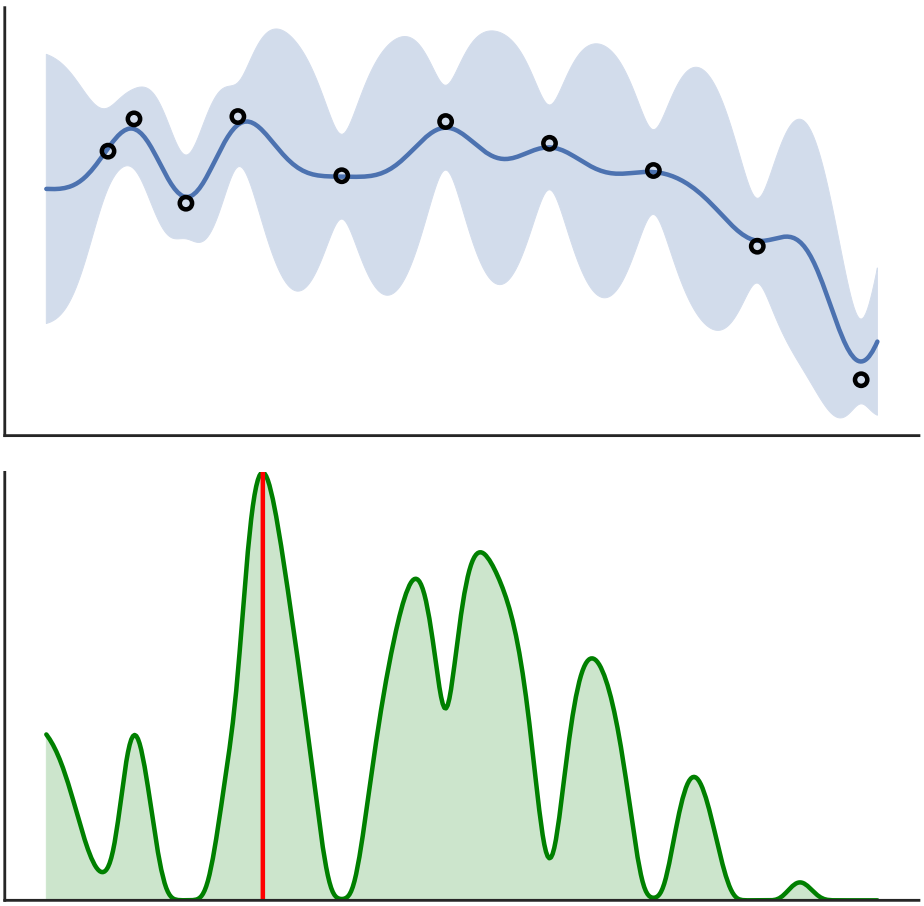


Bayesian optimization in a nutshell:

1. initial sample
2. construct a posterior model
3. **get the exploration strategy $\alpha(\mathbf{x})$**

Močkus et al. [1978], Jones et al. [1998], Jones [2001]

Bayesian black-box optimization

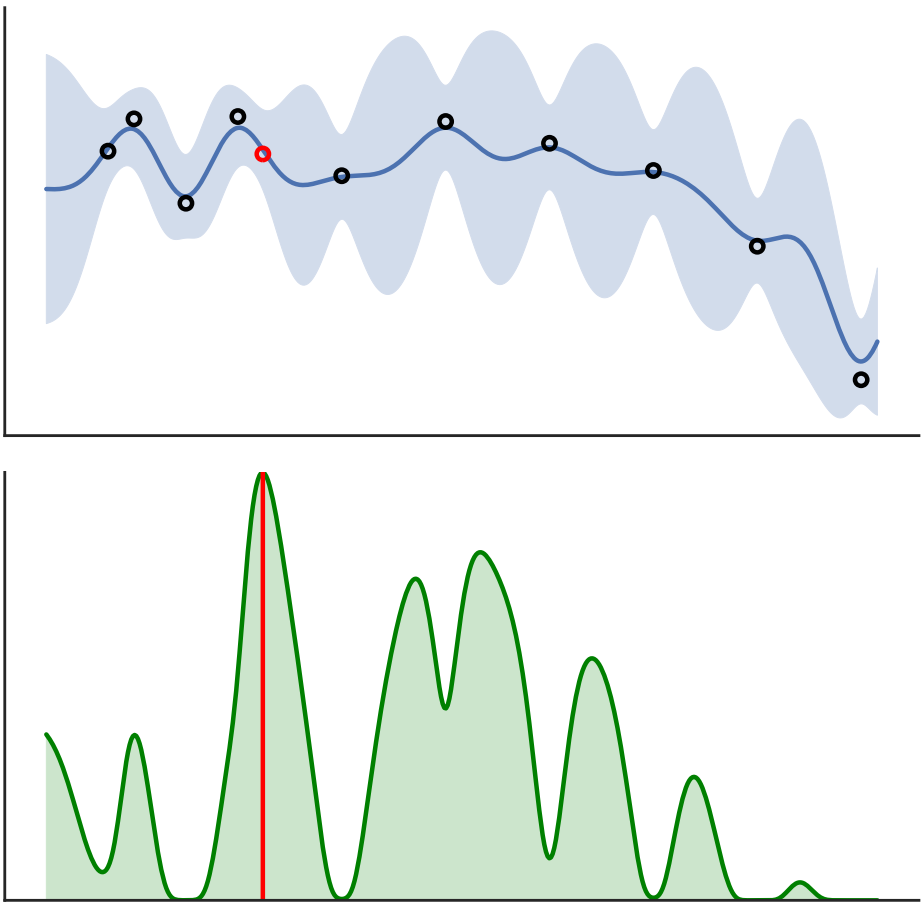


Bayesian optimization in a nutshell:

1. initial sample
2. construct a posterior model
3. get the exploration strategy $\alpha(\mathbf{x})$
4. **optimize it!**
   $\mathbf{x}_{\mathbf{next}} = \arg\max \alpha(\mathbf{x})$

Močkus et al. [1978], Jones et al. [1998], Jones [2001]

Bayesian black-box optimization

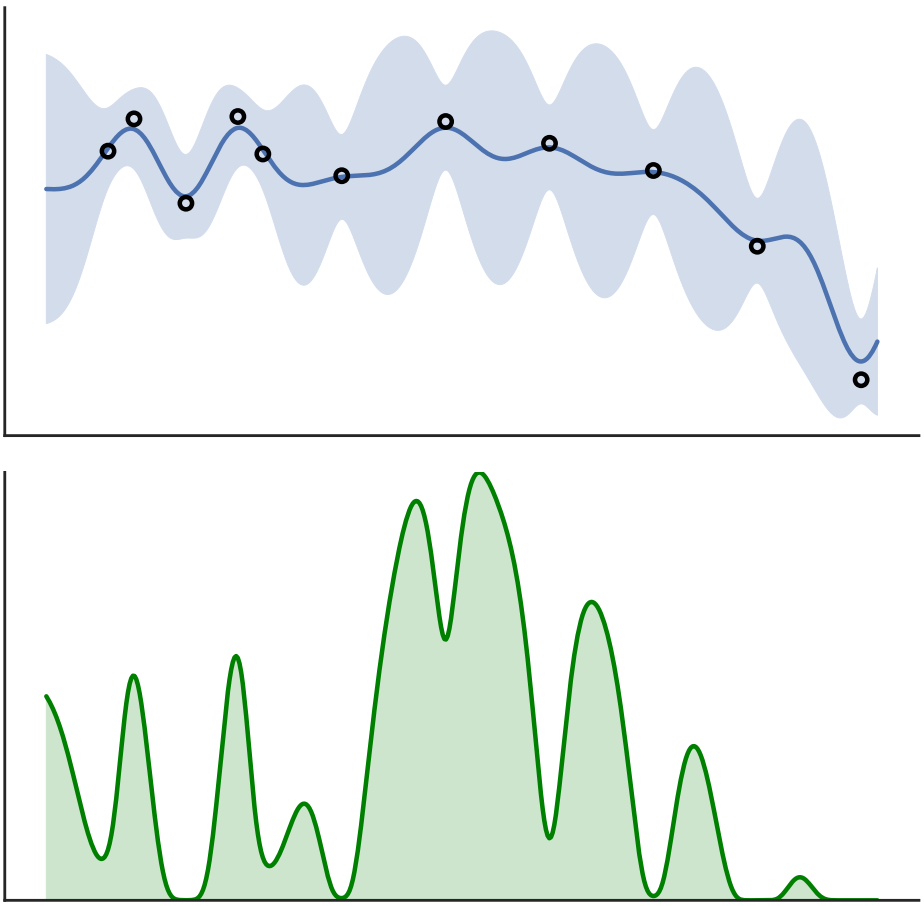


Bayesian optimization in a nutshell:

1. initial sample
2. construct a posterior model
3. get the exploration strategy $\alpha(\mathbf{x})$
4. optimize it! $\mathbf{x}_{\text{next}} = \arg\max \alpha(\mathbf{x})$
5. **sample new data; update model**

Močkus et al. [1978], Jones et al. [1998], Jones [2001]

Bayesian black-box optimization



Bayesian optimization in a nutshell:

1. initial sample
2. construct a posterior model
3. **get the exploration strategy $\alpha(\mathbf{x})$**
4. optimize it! $\mathbf{x}_{\text{next}} = \arg\max \alpha(\mathbf{x})$
5. sample new data; update model
6. **repeat!**

Močkus et al. [1978], Jones et al. [1998], Jones [2001]

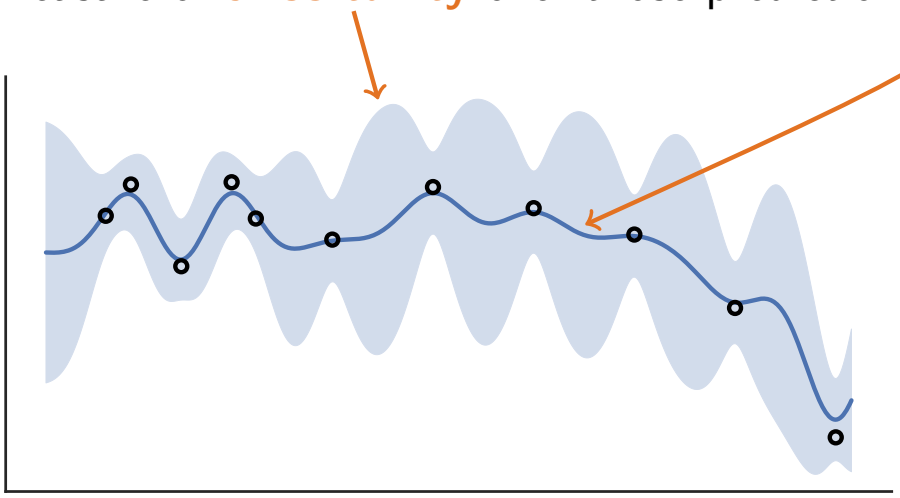
Two primary questions to answer are:

- what is my **model** and
- what is my **exploration strategy** given that model?

Modeling

Gaussian processes

We want a model that can both **make predictions** and maintain a measure of **uncertainty** over those predictions.



Gaussian processes provide a flexible prior for modeling continuous functions of this form.

Rasmussen and Williams [2006]

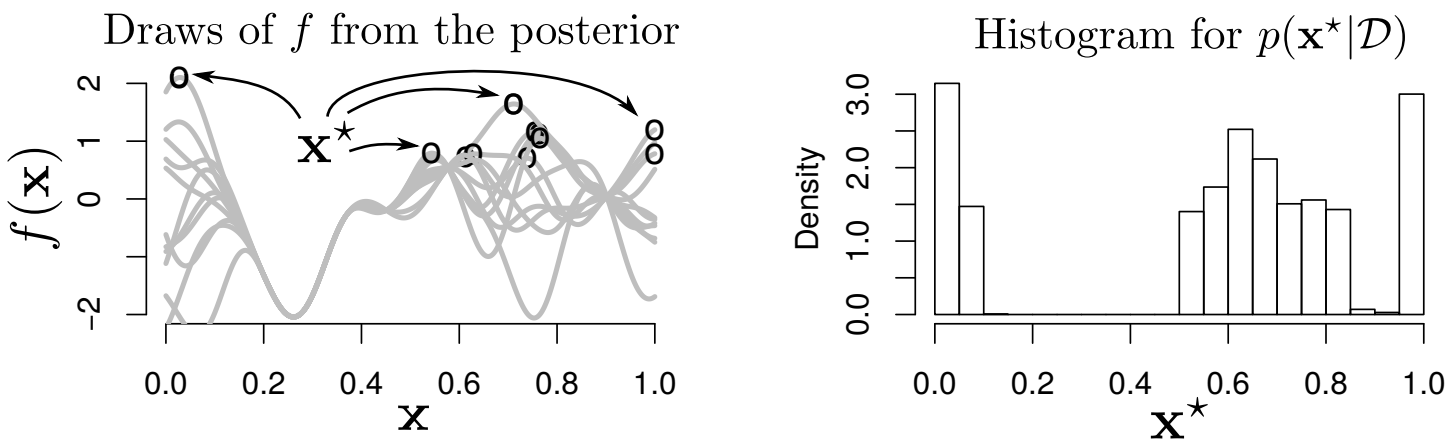
Exploration strategies

The simplest acquisition function

Thompson sampling is perhaps the simplest acquisition function to implement and uses a random acquisition function:

$$\alpha \sim p(f|\mathcal{D})$$

We can also view this as a random strategy sampling $\mathbf{x}_{\text{next}}$ from $p(\mathbf{x}_\star|\mathcal{D})$



Thompson [1933]

Of course for GPs f is an infinite-dimensional object so sampling and optimizing it is not quite as simple.

- we could **lazily evalauate** f but the complexity of this grows with the number function evaluations necessary to optimize it.
- Instead we will approximate $f(\cdot) \approx \phi(\cdot)^\mathsf{T}\boldsymbol{\theta}$ with **random features**

$$\phi(\mathbf{x}) = \cos(\mathbf{W}^\mathsf{T}\mathbf{x} + \mathbf{b})$$

- $p(\mathbf{W}, \mathbf{b})$ depends on the kernel of the GP
- and $\boldsymbol{\theta}$ is determined simply by Bayesian linear regression

Rahimi and Recht [2007], **Shahriari et al. [2014]**, **Hernández-Lobato et al. [2014]**

There are many other exploration strategies

- Expected Improvement
- Probability of Improvement
- UCB, etc.

but intuitively they all try and greedily **gain information** about the maximum

Predictive Entropy Search

A common strategy in **active learning** is to select points maximizing the expected reduction in posterior entropy.

In our setting this corresponds to minimizing the entropy of the unknown maximizer $\mathbf{x}_\star$:

$$\begin{aligned} \alpha(\mathbf{x}) &= \mathrm{H}\left[\mathbf{x}_\star \middle| \mathcal{D}\right] - \mathbb{E}_{y_\mathbf{x}}\left[\mathrm{H}\left[\mathbf{x}_\star \middle| \mathcal{D} \cup \{y_\mathbf{x}\}\right] \middle| \mathcal{D}\right] && \text{(ES)} \\ &= \textbf{mutual information} \\ &= \mathrm{H}\left[y_\mathbf{x} \middle| \mathcal{D}\right] - \mathbb{E}_{\mathbf{x}_\star}\left[\mathrm{H}\left[y_\mathbf{x} \middle| \mathcal{D}, \mathbf{x}_\star\right] \middle| \mathcal{D}\right] && \text{(PES)} \end{aligned}$$

The first quantity is difficult to approximate, but the second only concerns predictive distributions; we call this **Predictive Entropy Search**.

Villemonteix et al. [2009], Hennig and Schuler [2012], **Hernández-Lobato et al. [2014]**

Computing the PES acquisition function

We can write the acquisition function as,

$$\alpha(\mathbf{x}) \approx \mathrm{H}\left[y_{\mathbf{x}} \middle| \mathcal{D}\right] - \frac{1}{M}\sum_i \mathrm{H}\left[y_{\mathbf{x}} \middle| \mathcal{D}, \mathbf{x}_\star^i\right] \qquad \mathbf{x}_\star^i \sim p(\cdot|\mathcal{D})$$

under Gaussian assumptions (and eliminating constants) this is

$$\approx \log v(\mathbf{x}|\mathcal{D}) - \frac{1}{M}\sum_i \log v(\mathbf{x}|\mathcal{D}, \mathbf{x}_\star^i)$$

This can be done as follows:

1. sampling $\mathbf{x}_\star$ is just Thompson sampling!
2. we then need to approximate $p(y_{\mathbf{x}}|\mathcal{D}, \mathbf{x}_\star^i)$ with a Gaussian

Approximating the conditional

The fact that $\mathbf{x}_\star$ is a global maximizer can be approximated with the following constraints:

$$f(\mathbf{x}_\star) > \max_t f(\mathbf{x}_t) \quad f(\mathbf{x}_\star) > f(\mathbf{x})$$

The distribution,

$$p\big(f(\mathbf{x}_\star) \big| \text{ A } \big) \approx \mathcal{N}(\mathbf{m}_1, \mathbf{V}_1)$$

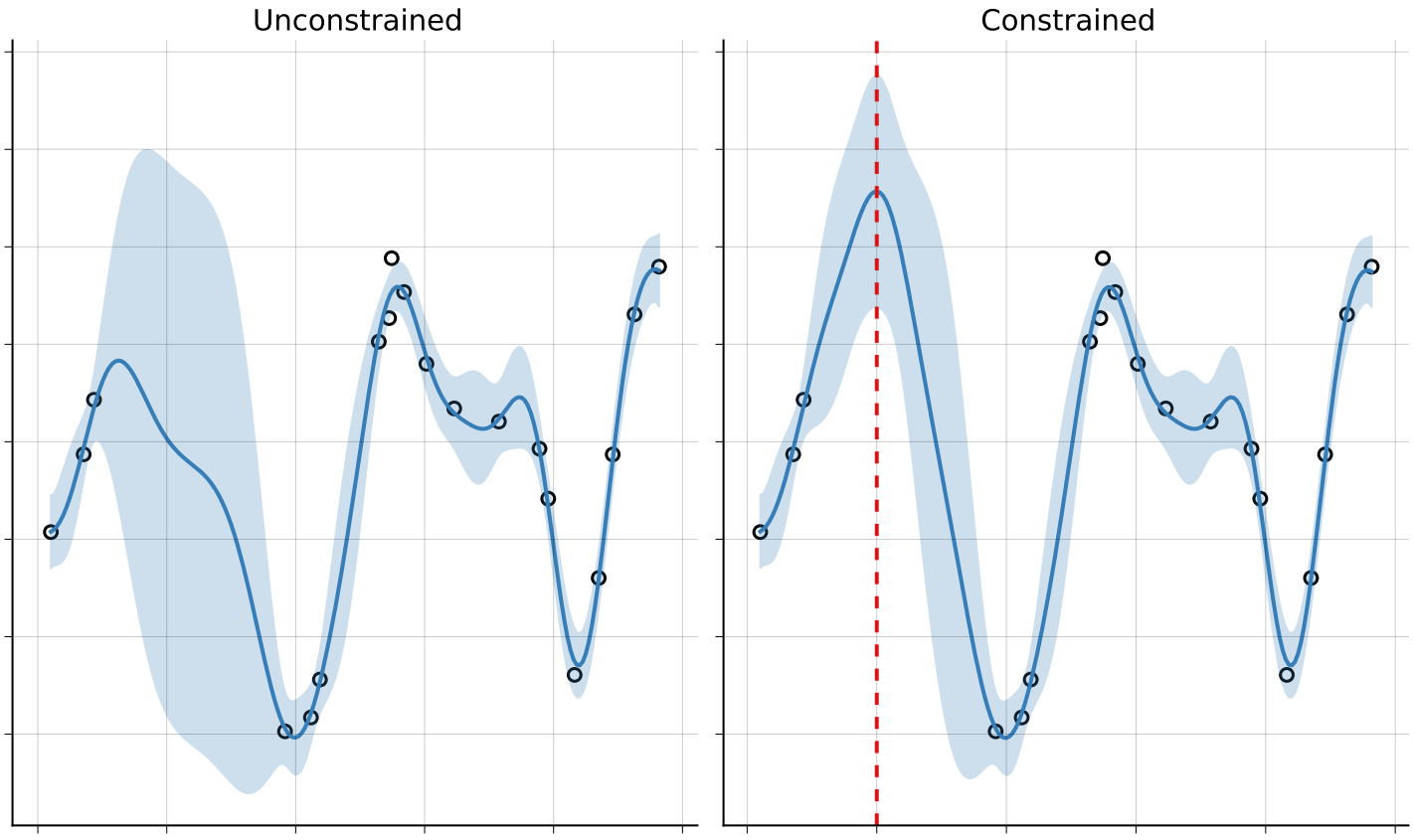
can be approximated using EP. From there in closed-form we can approximate for any $\mathbf{x}$,

$$p\big(f(\mathbf{x}), f(\mathbf{x}_\star) \big| \text{ A } \big)$$

and finally, with one moment-matching step we can approximate,

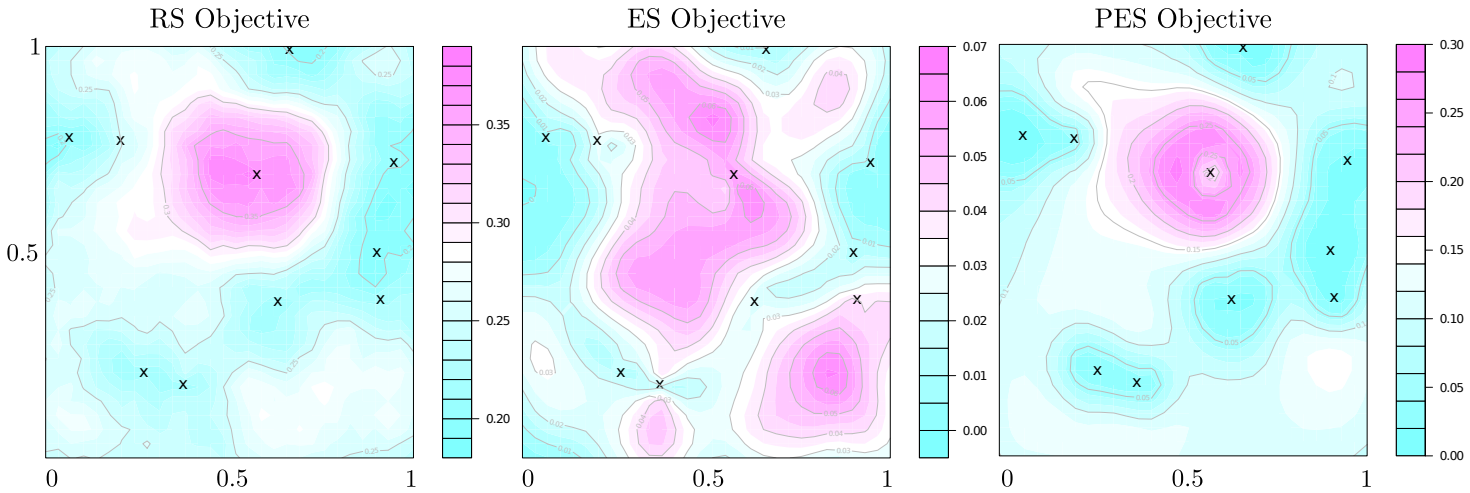
$$p\big(f(\mathbf{x}) \big| \text{ A }, \text{ B } \big) \approx \mathcal{N}(m, v)$$

Minka [2001]


Unconstrained
Constrained

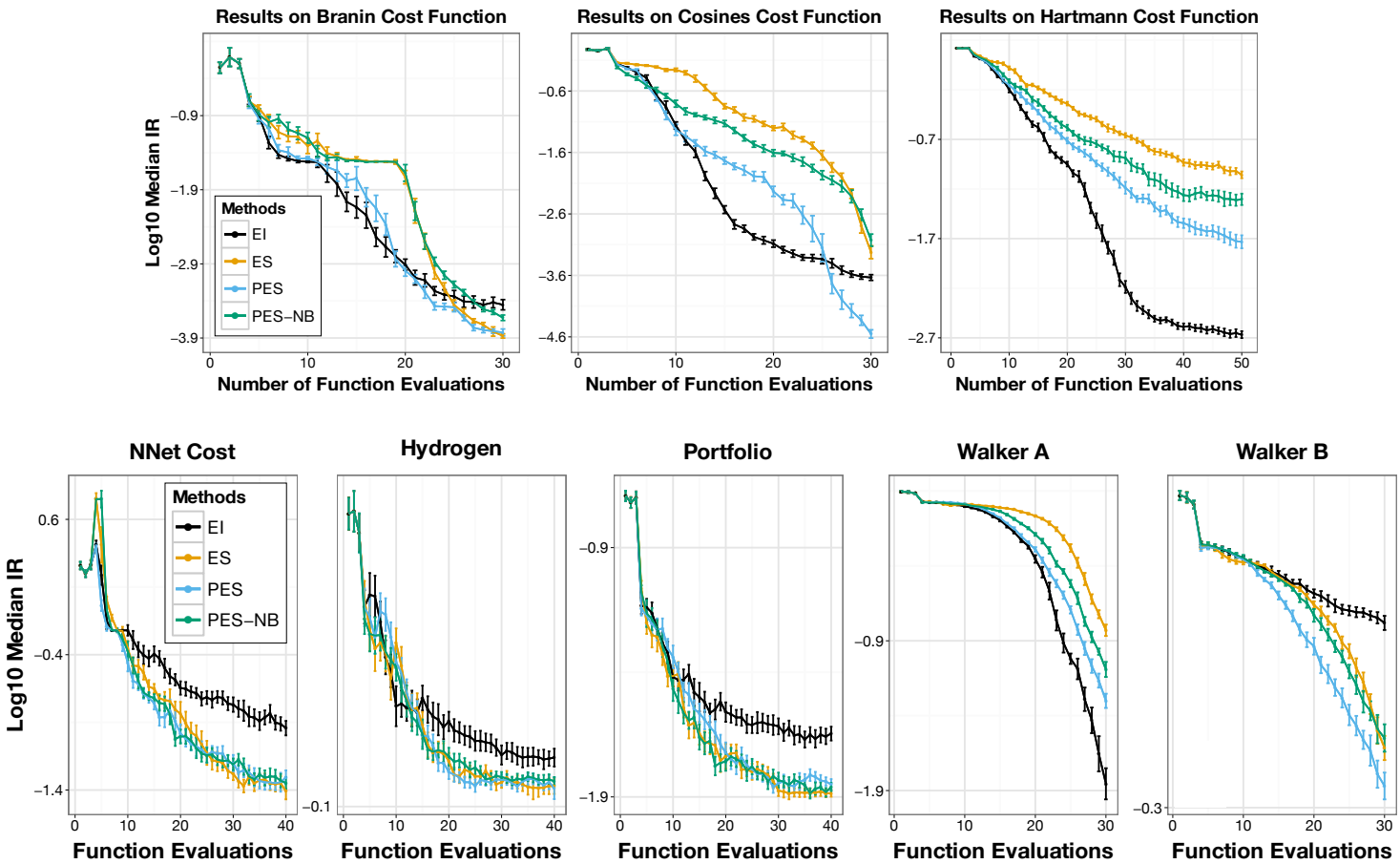
Accuracy of the PES approximation

The following compares a fine-grained random sampling (RS) scheme to compute the ground truth objective with ES and PES.



We see PES provides a much better approximation.

Results on real-world tasks



Portfolios of meta-algorithms

Of course each of these acquisition functions can be seen as a heuristic for the intractable optimal solution

So we can consider mixing over strategies in order to correct for any sub-optimality

- **[Hoffman et al., 2011]**
- **[Shahriari et al., 2014]**, uses a similar entropy-based strategy to PES

An extension to constrained black-box problems

This framework also easily allows us to tackle problems with constraints

$$\max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) \quad \text{s.t.} \quad c_1(\mathbf{x}) \geq 0, \ldots, c_K(\mathbf{x}) \geq 0$$

where $f, c_1, \ldots, c_k$ are all black-boxes.

- we will model each function with a GP prior
- can write the same acquisition function

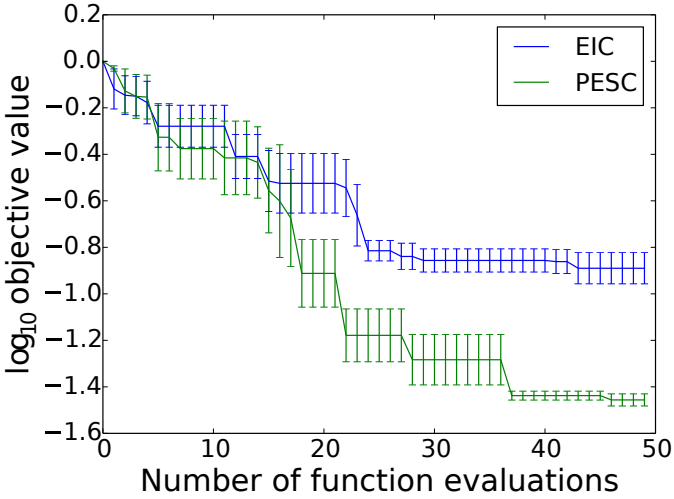
$$\alpha(\mathbf{x}) = \mathrm{H}\left[\mathbf{y}_{\mathbf{x}} \middle| \mathcal{D}\right] - \mathbb{E}_{\mathbf{x}_\star}\left[\mathrm{H}\left[\mathbf{y}_{\mathbf{x}} \middle| \mathcal{D}, \mathbf{x}_\star\right] \middle| \mathcal{D}\right]$$

  except $\mathbf{y}$ now contains both function and constraints

Hernández-Lobato et al. [2015]

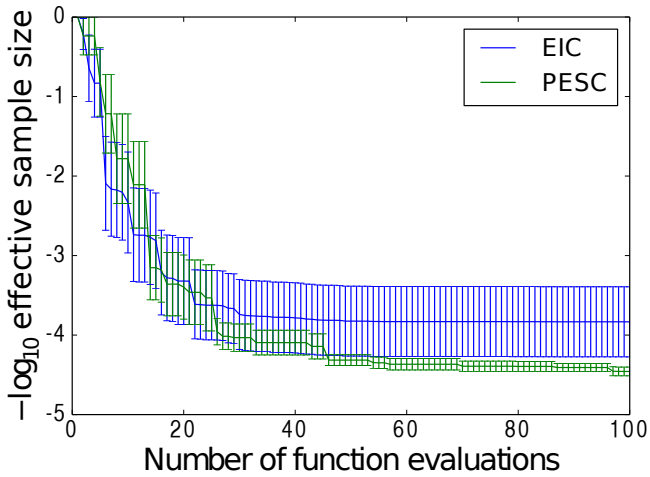
Tuning a fast neural network

Tune the hyperparameters of a neural network subject to the constraint that prediction time must not exceed 2 ms



Tuning Hamiltonian MCMC

Optimize the effective sample size of HMC subject to convergence diagnostic constraints



So what are the **problems** with PES?

PES with non-conjugate likelihoods

When introducing the PES approximations I included the constraint

$$f(\mathbf{x}_\star) > \max_t f(\mathbf{x}_t)$$

But we never actually observe $f(\mathbf{x}_t)$. Instead this is incorporated as a **soft constraint**

$$f(\mathbf{x}_\star) > \max_t y_t + \mathcal{N}(0, \sigma^2)$$

but this explicitly requires a **Gaussian likelihood**

PES with disjoint input spaces

Consider optimizing over a space

$$\mathcal{X} = \cup_{i=1}^{n} \mathcal{X}_d$$

of disjoint discrete/continuous spaces with potentially differing dimensionalities.

- each of these spaces could be the parameters of a different learning algorithm
- but the entropy $\mathrm{H}[\mathbf{x}_\star | \mathcal{D}]$ is **not well-defined** in this setting

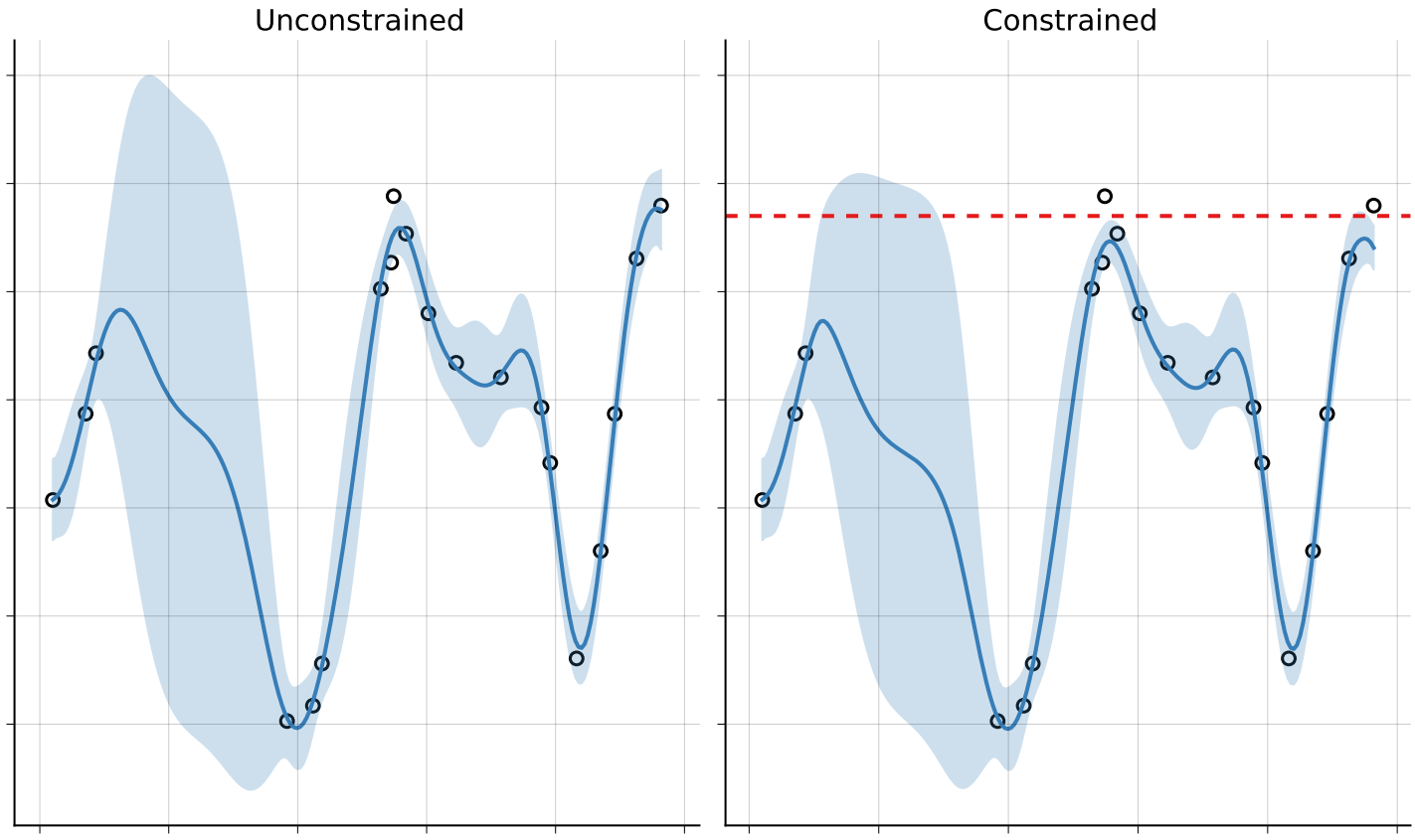
A potential solution: output-space PES

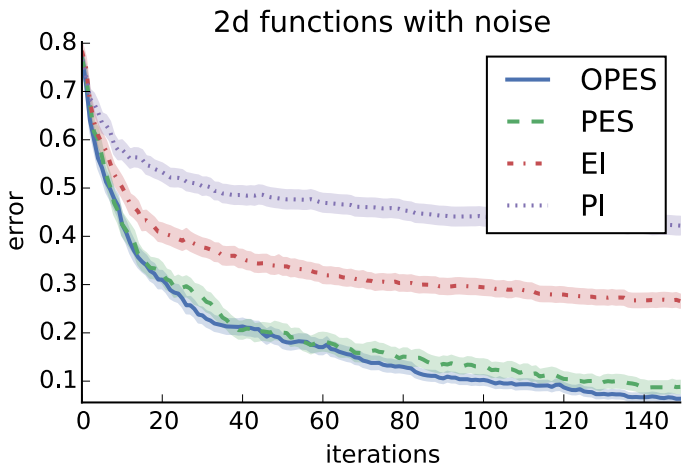
The main problem here is the fact that we are conditioning on or taking the entropy of $\mathbf{x}_\star$

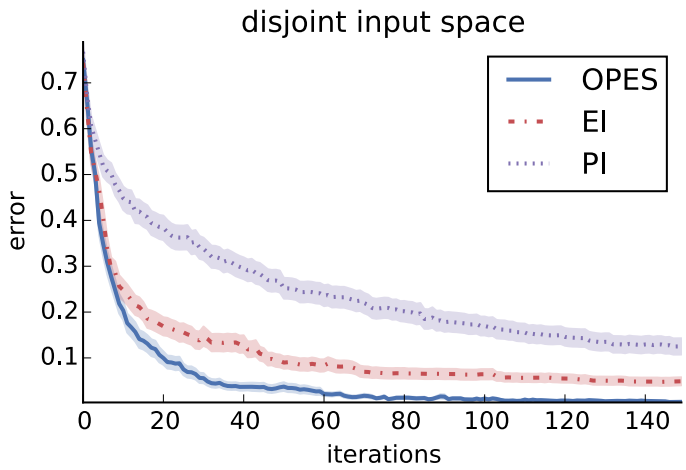
So let's stop doing that:

$$\begin{aligned}\alpha(\mathbf{x}) &= \mathsf{H}\big[\, f_\star \,\big|\mathcal{D}\big] - \mathbb{E}_{y_\mathbf{x}}\Big[\mathsf{H}\big[\, f_\star \,\big|\mathcal{D} \cup \{y_\mathbf{x}\}\big]\Big|\mathcal{D}\Big] \\ &\vdots \\ &= \mathsf{H}\big[y_\mathbf{x}\big|\mathcal{D}\big] - \mathbb{E}_{f_\star}\Big[\mathsf{H}\big[y_\mathbf{x}\big|\mathcal{D}, f_\star\big]\Big|\mathcal{D}\Big]\end{aligned}$$

which I'm calling **output-space PES**


Unconstrained
Constrained





Preliminary results indicate this can be as effective as PES and applicable where PES is not

PyBO as it stands now

I was quite glib before when I mentioned my GP model...

```
# base GP model
m = make_gp(sn, sf, ell)

# set priors
m.params['like.sn2'].set_prior('lognormal', 0, 10)
m.params['kern.rho'].set_prior('lognormal', 0, 100)
m.params['kern.ell'].set_prior('lognormal', 0, 10)
m.params['mean.bias'].set_prior('normal', 0, 20)

# marginalize hypers
m = MCMC(m)

# do some bayesopt...
```

https://github.com/mwhoffman/pybo

Modular Bayesian optimization

But what we're moving towards:

```
# PI
m.get_tail(X, fplus)

# EI
m.get_improvement(X, fplus)

# OPES
sum(m.get_entropy(X)
    - m.condition_fstar(fplus).get_entropy(X)
    for i in xrange(100))
```

References I

J.-Y. Audibert and S. Bubeck. Best arm identification in multi-armed bandits. In *Conference on Learning Theory*, pages 13–p, 2010.

V. Gabillon, M. Ghavamzadeh, and A. Lazaric. Best arm identification: A unified approach to fixed budget and fixed confidence. In *Advances in Neural Information Processing Systems*, 2012.

P. Hennig and C. J. Schuler. Entropy search for information-efficient global optimization. *the Journal of Machine Learning Research*, 13: 1809–1837, 2012.

J. M. Hernández-Lobato, M. W. Hoffman, and Z. Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *Advances in Neural Information Processing Systems*, 2014.

References II

J. M. Hernández-Lobato, M. Gelbart, M. W. Hoffman, R. P. Adams, and Z. Ghahramani. Predictive entropy search for Bayesian optimization with unknown constraints. In *the International Conference on Machine Learning*, 2015.

M. W. Hoffman, E. Brochu, and N. de Freitas. Portfolio allocation for Bayesian optimization. In *Uncertainty in Artificial Intelligence*, pages 327–336, 2011.

M. W. Hoffman, B. Shahriari, and N. de Freitas. On correlation and budget constraints in model-based bandit optimization with application to automatic machine learning. In *the International Conference on Artificial Intelligence and Statistics*, pages 365–374, 2014.

D. R. Jones. A taxonomy of global optimization methods based on response surfaces. *Journal of global optimization*, 21(4):345–383, 2001.

References III

D. R. Jones, M. Schonlau, and W. J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492, 1998.

E. Kaufmann, O. Cappé, and A. Garivier. On the complexity of best arm identification in multi-armed bandit models. *arXiv preprint arXiv:1407.4443*, 2014.

T. P. Minka. *A family of algorithms for approximate Bayesian inference*. PhD thesis, Massachusetts Institute of Technology, 2001.

J. Močkus, V. Tiesis, and A. Žilinskas. The application of Bayesian methods for seeking the extremum. In L. Dixon and G. Szego, editors, *Toward Global Optimization*, volume 2. Elsevier, 1978.

R. Munos. Optimistic optimization of deterministic functions without the knowledge of its smoothness. In *Advances in neural information processing systems*, 2011.

References IV

A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, pages 1177–1184, 2007.

C. E. Rasmussen and C. K. Williams. *Gaussian processes for machine learning*. The MIT Press, 2006.

B. Shahriari, Z. Wang, M. W. Hoffman, A. Bouchard-Côté, and N. de Freitas. An entropy search portfolio for Bayesian optimization. In *NIPS Workshop on Bayesian Optimization*, 2014.

M. Soare, A. Lazaric, and R. Munos. Best-arm identification in linear bandits. In *Advances in Neural Information Processing Systems*, pages 828–836, 2014.

W. R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285–294, 1933.

References V

J. Villemonteix, E. Vazquez, and E. Walter. An informational approach to the global optimization of expensive-to-evaluate functions. *Journal of Global Optimization*, 44(4):509–534, 2009.