Bayesian optimization and Information-based Approaches

José Miguel Hernández-Lobato

joint work with
Michael A. Gelbart, Matt W. Hoffman, Ryan P. Adams and
Zoubin Ghahramani

April 31, 2015 (50% of these slides have been made by Matt)

HARVARD
INTELLIGENT
PROBABILISTIC
SYSTEMS

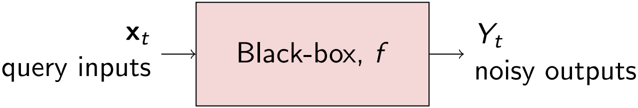
Bayesian optimization

We are interested in solving black-box optimization problems of the form

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

where black-box means:

- We may only be able to observe the function value, **no gradients**.
- Our observations may be **corrupted by noise**.



$\mathbf{x}_t$ query inputs → Black-box, f → Y_t noisy outputs

- One requirement on the noisy outputs: $\mathbb{E}[Y_t|\mathbf{x}_t] = f(\mathbf{x}_t)$.

Black-box queries are **very expensive** (time, economic cost, etc...).

Example (AB testing)

Users visit our website which has different configurations (A and B) and we want to find the **best configuration** (possibly online).

Example (Hyperparameter tuning)

We have some algorithm which relies on **hyperparameters** which we want to optimize with respect to performance.

Example (Design of new molecules)

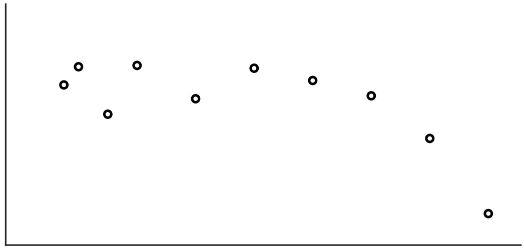
We want to find molecular compounds with optimal **chemical properties**: more efficient solar panels, batteries, drugs, etc...

Bayesian black-box optimization

Bayesian optimization in a nutshell:

1. **Get initial sample.**

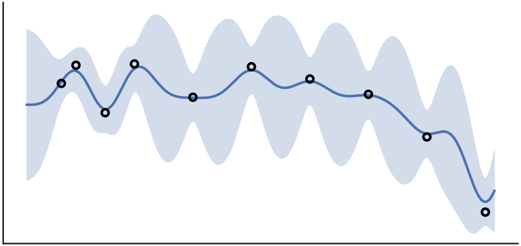
Bayesian black-box optimization



Bayesian optimization in a nutshell:

1. Get initial sample.
2. **Construct a posterior model.**

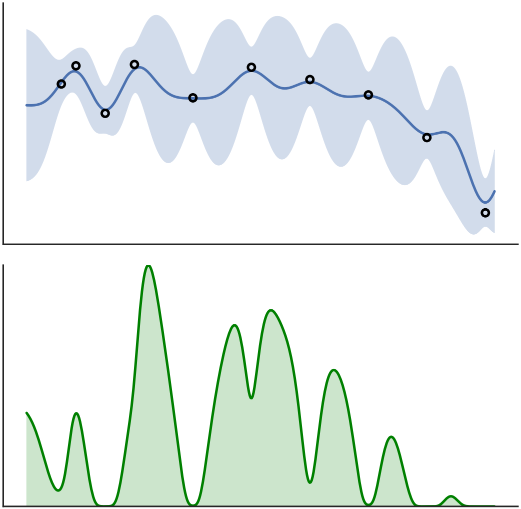
Bayesian black-box optimization



Bayesian optimization in a nutshell:

1. Get initial sample.
2. Construct a posterior model.
3. **Select the exploration strategy. . .**

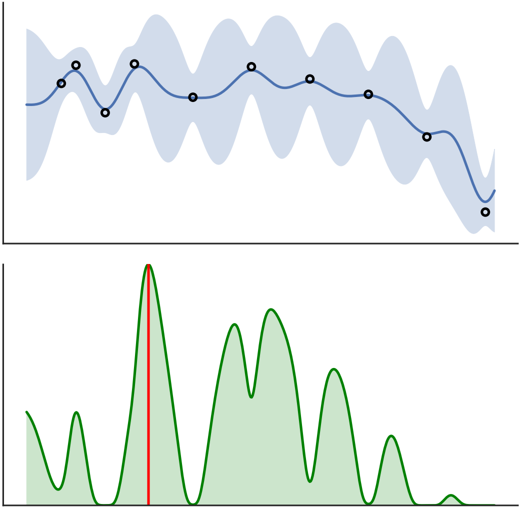
Bayesian black-box optimization



Bayesian optimization in a nutshell:

1. Get initial sample.
2. Construct a posterior model.
3. Select the exploration strategy. . .
4. **. . . and optimize it.**

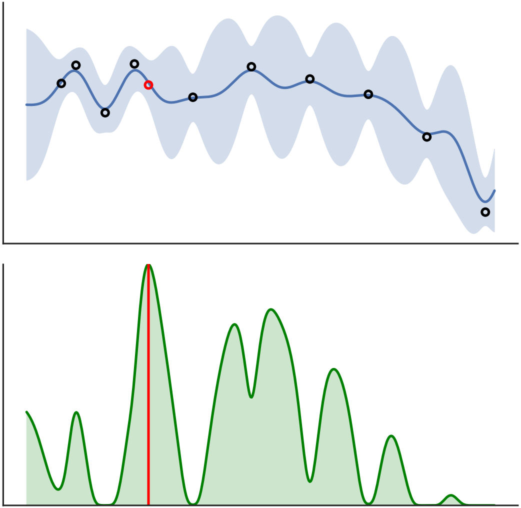
Bayesian black-box optimization



Bayesian optimization in a nutshell:

1. Get initial sample.
2. Construct a posterior model.
3. Select the exploration strategy. . .
4. . . . and optimize it.
5. **Sample new data; update model.**

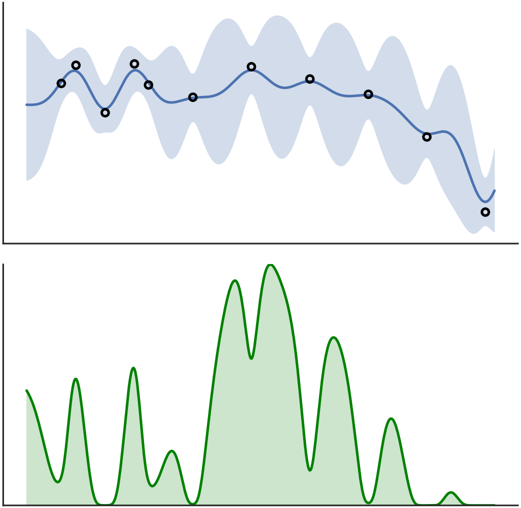
Bayesian black-box optimization



Bayesian optimization in a nutshell:

1. Get initial sample.
2. Construct a posterior model.
3. Select the exploration strategy...
4. ...and optimize it.
5. Sample new data; update model.
6. **Repeat!**

Bayesian black-box optimization



Bayesian optimization in a nutshell:

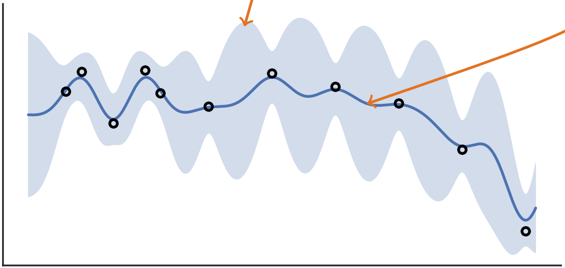
1. Get initial sample.
2. Construct a posterior model.
3. **Select the exploration strategy. . .**
4. . . . and optimize it.
5. Sample new data; update model.
6. Repeat!

Two primary questions to answer are:

- What is the **model** and
- What is the **exploration strategy** given the model?

Modeling

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.

Bayesian neural networks are an alternative when the data size is large.

Snoek et al. [2015]

Modeling: Gaussian processes

A Gaussian process $f \sim \mathrm{GP}(m, k)$ defines a distribution over functions such that any finite collection of evaluations at $\mathbf{x}_{1:n}$ are Normally distributed,

$$\begin{bmatrix} f(\mathbf{x}_1) \\ \vdots \\ f(\mathbf{x}_t) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m(\mathbf{x}_1) \\ \vdots \\ m(\mathbf{x}_t) \end{bmatrix}, \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \dots & k(\mathbf{x}_1, \mathbf{x}_t) \\ \vdots & & \vdots \\ k(\mathbf{x}_t, \mathbf{x}_1) & \dots & k(\mathbf{x}_t, \mathbf{x}_t) \end{bmatrix} \right)$$

If the observations y are the result of Normal noise on f, then

- $P(y_{1:n}, f(\mathbf{x}_{1:n}))$ is jointly Gaussian.
- Conditioning can be done in closed-form.
- The result is a tractable GP posterior distribution.

Rasmussen and Williams [2006]

The exploration strategy: expected improvement

The exploration strategy must explicitly **trade off between exploration and exploitation**.

Should map the model and a query point to expected **future value**. The result is an **acquisition function**.

Common approach: maximize the **Expected Improvement** (EI):

$$\alpha_t(\mathbf{x}) = \mathbb{E}_{f(\mathbf{x})}\Big[\max\Big(0, f(\mathbf{x}) - f(\mathbf{x}_+)\Big)\Big|\mathcal{D}_t\Big]$$

$\mathcal{D}_t$, the observations.

$\mathbf{x}_+$, best value so far.

Intuitively, EI selects the point which gives us the most improvement over our current best solution, in the next round.

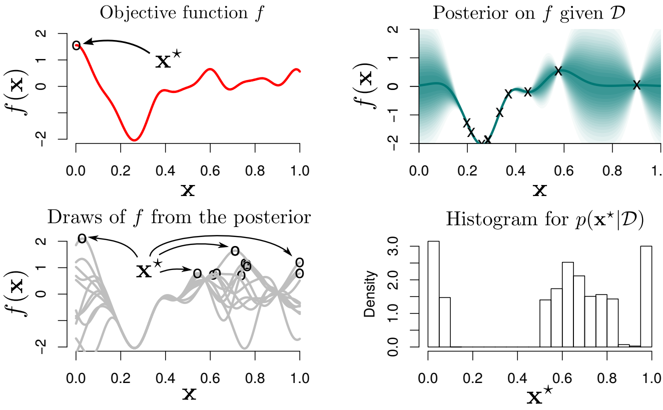
Mockus et al. [1978], Jones et al. [1998]

The exploration strategy: Entropy Search

Entropy search (ES) maximizes the expected reduction in entropy:

$$\alpha_t(\mathbf{x}) = \mathrm{H}\left[\mathbf{x}_\star \middle| \mathcal{D}_t\right] - \mathbb{E}_y\left[\mathrm{H}\left[\mathbf{x}_\star \middle| \mathcal{D}_t \cup \{(\mathbf{x}, y)\}\right] \middle| \mathcal{D}_t, \mathbf{x}\right] \tag{ES}$$

where $\mathbf{x}_\star$ is the unknown **global** optimizer.



Objective function f

Posterior on f given $\mathcal{D}$

Draws of f from the posterior

Histogram for $p(\mathbf{x}^\star|\mathcal{D})$

Villemonteix et al. [2009], Hennig and Schuler [2012]

Predictive Entropy Search

The ES acquisition function is equal to $I(y, \mathbf{x}_\star) = I(\mathbf{x}_\star, y)$.

We can swap y and $\mathbf{x}_\star$ and rewrite the acquisition as

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

which we call **Predictive Entropy Search**.

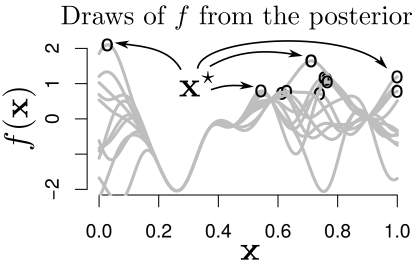
Approximating the PES acquisition function can be done in two steps:

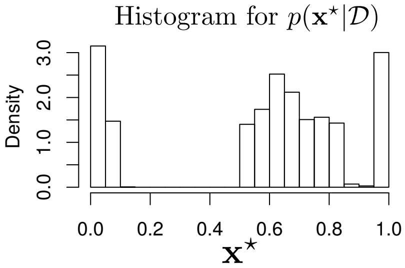
1. Sampling from the distribution over global maximizers $\mathbf{x}_\star$.
2. Estimating the predictive entropy for y conditioned on $\mathbf{x}_\star$.

Hernández-Lobato et al. [2014]

1: sampling the location of the optimum $\mathbf{x}_\star$

To sample $\mathbf{x}_\star$ we need only sample $\tilde{f} \sim p(f|\mathcal{D}_t)$ and return $\arg\max_{\mathbf{x}} \tilde{f}(\mathbf{x})$.





However, $\tilde{f}$ is an infinite dimensional object!

Instead we use $\tilde{f}(\cdot) \approx \phi(\cdot)^\mathsf{T}\boldsymbol{\theta}$ where $\phi(\mathbf{x}) = \sqrt{2\alpha/m}\cos(\mathbf{W}\mathbf{x} + \mathbf{b})$.

Bochner's theorem shows that when $m \to \infty$ the approximation is exact.

Bochner [1959]

2: Approximating the distribution of y given $\mathbf{x}_\star$

Instead of conditioning to $\mathbf{x}_\star$, we use the following simplified constraints:

$$\nabla f(\mathbf{x}_\star) = 0$$
$$\text{upper}[\nabla^2 f(\mathbf{x}_\star)] = 0$$

A

$$\mathbf{d} = \text{diag}[\nabla^2 f(\mathbf{x}_\star)] < 0$$
$$f(\mathbf{x}_\star) > \max_t f(\mathbf{x}_t)$$

B

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

C

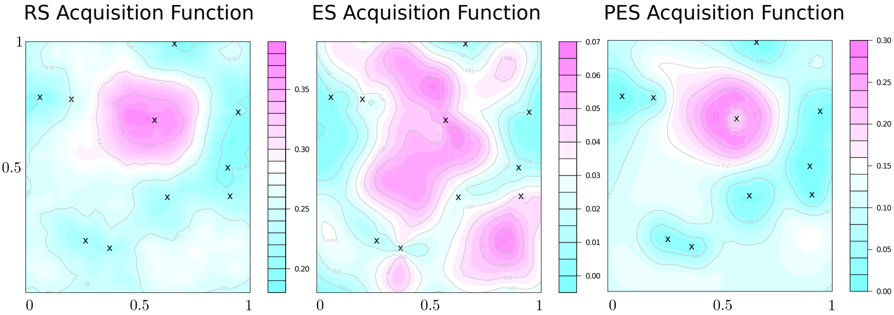
- We can incorporate the equality constraints on the gradient and the Hessian **analytically**.

- To deal with the inequality constraints we use the method **expectation propagation** (EP).

- The result is a Gaussian approximation to $p(y \big| \mathcal{D}_t, \mathbf{x}, \mathbf{x}_\star)$ for which we can easily calculate the entropy.

Accuracy of the PES approximation

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



RS Acquisition Function

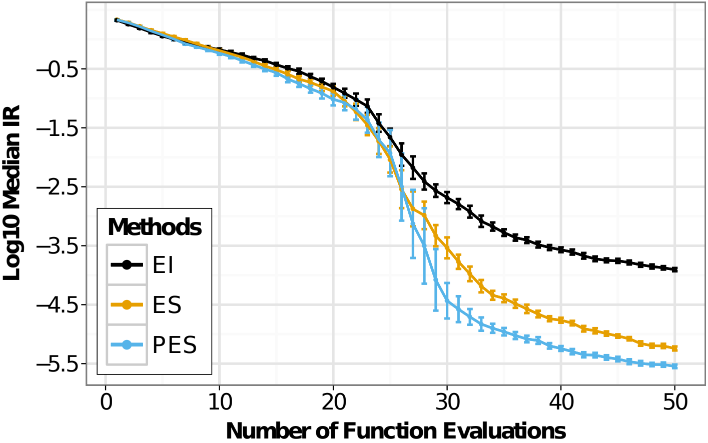
ES Acquisition Function

PES Acquisition Function

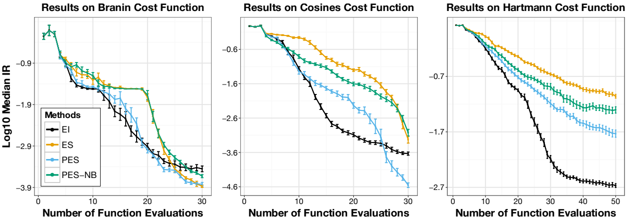
We see that PES provides a much better approximation.

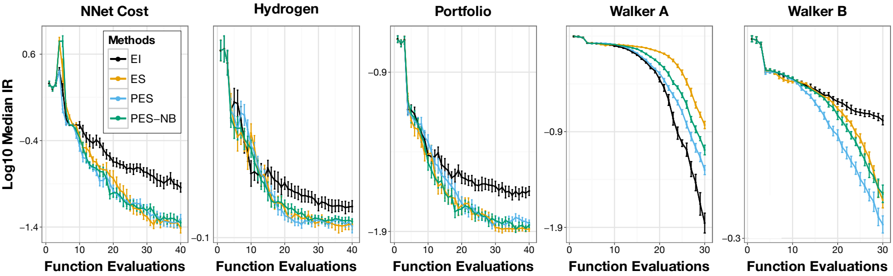
Results on simulated data

Here we show results where the objective function is sampled from a known GP prior.



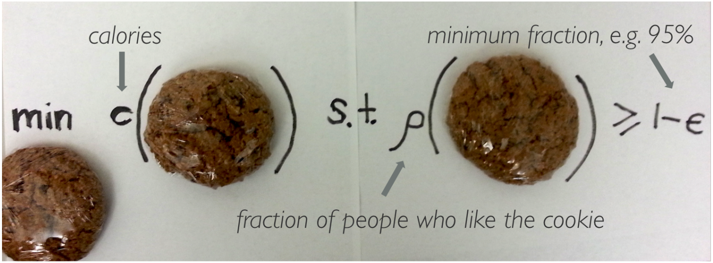
Results on more realistic tasks





Bayesian optimization with unknown constraints

A cookie company wants to create a low-calorie cookie that is just as tasty as the original. This is a **constrained optimization** problem over the parameterized space of cookie recipes:



calories
minimum fraction, e.g. 95%
min c( ) s.t. ρ( ) ≥ 1−ε
fraction of people who like the cookie

More generally, we want to solve

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

where f and $c_1, \ldots, c_K$ are **unknown** and return **noisy** values.

Predictive entropy search with unknown constraints

The PESC acquisition function is

$$\alpha_t(\mathbf{x}) = \mathrm{H}\left[\mathbf{y}\middle|\mathcal{D}_t,\mathbf{x}\right] - \mathbb{E}_{\mathbf{x}_\star}\left[\mathrm{H}\left[\mathbf{y}\middle|\mathcal{D}_t,\mathbf{x},\mathbf{x}_\star\right]\middle|\mathcal{D}_t,\mathbf{x}\right]. \qquad \text{(PESC)}$$

An approximation is obtained in two steps (as in PES):

1. Sampling from the distribution over global maximizers $\mathbf{x}_\star$.
   Sample $\tilde{f} \sim p(f|\mathcal{D}_t), \tilde{c}_1 \sim p(c_1|\mathcal{D}_t), \ldots, \tilde{c}_K \sim p(c_K|\mathcal{D}_t)$ and solve

$$\arg\max_{\mathbf{x}} \tilde{f}(\mathbf{x}) \quad \text{s.t.} \quad \tilde{c}_1(\mathbf{x}) \geq 0, \ldots, \tilde{c}_K(\mathbf{x}) \geq 0\,,$$

2. Estimating the predictive entropy for $\mathbf{y}$ conditioned on $\mathbf{x}_\star$.

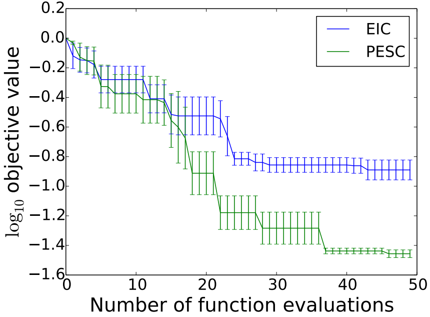
$$p(\mathbf{y}|\mathcal{D}_t,\mathbf{x},\mathbf{x}_\star) \propto \int \delta[y_0 - f(\mathbf{x})]\left[\textstyle\prod_{k=1}^K \delta[y_k - c_k(\mathbf{x})]\right]$$
$$\textstyle\prod_{\mathbf{x}'\neq\mathbf{x}_\star}\left[\left\{\prod_{k=1}^K \Theta[c_k(\mathbf{x}')]\right\}\Theta[f(\mathbf{x}_\star) - f(\mathbf{x}')] + \left\{1 - \prod_{k=1}^K \Theta[c_k(\mathbf{x}')\right\}\right.$$
$$\left[\textstyle\prod_{k=1}^K \Theta[c_k(\mathbf{x}_\star)]\right] p(f, c_1, \ldots, c_K|\mathcal{D}_t)\, df\, dc_1 \ldots dc_k\,.$$

Approximated with a **product of univariate Gaussians** using **EP**.

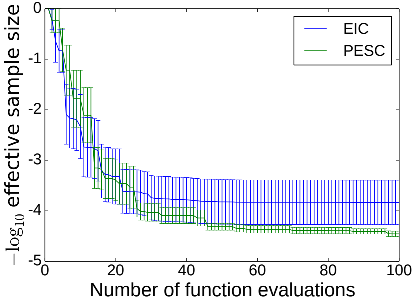
Hernández-Lobato et al. [2015]

Experimental results

Optimizing a neural network validation error on MNIST when constrained to make predictions in under 2ms.



Optimizing the effective sample size of HMC on logistic regression when constrained to pass convergence diagnostics.

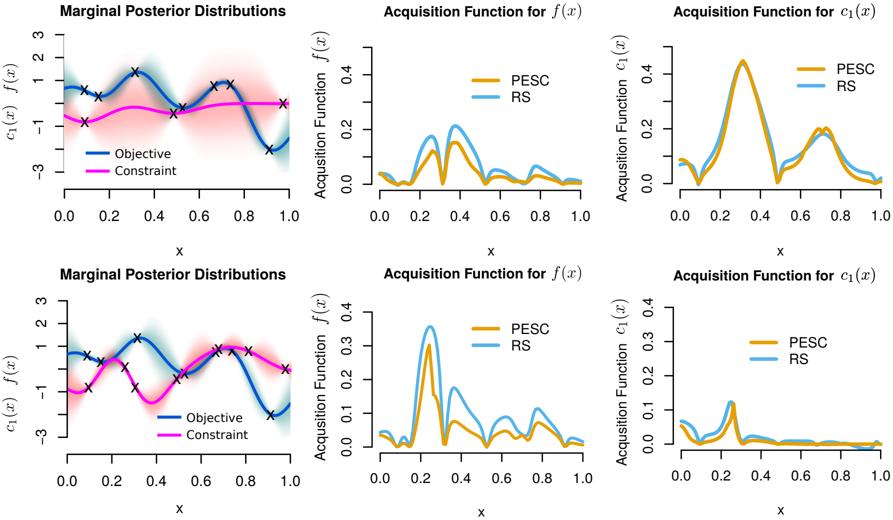


Baseline: **expected improvement with constraints** (EIC):

$$\alpha_t(\mathbf{x}) = \mathbb{E}\Big[\max\Big(0, f(\mathbf{x}) - f(\mathbf{x}_+)\Big)\Big|\mathcal{D}_t\Big]\Big[\prod_{k=1}^{K} p(c_k(\mathbf{x}) \geq 0)\Big]$$

PESC in a decoupled evaluation setting

The PESC acquisition function is **additive** across f and $c_1, \ldots, c_K$.



Marginal Posterior Distributions

Acquisition Function for $f(x)$

Acquisition Function for $c_1(x)$

Marginal Posterior Distributions

Acquisition Function for $f(x)$

Acquisition Function for $c_1(x)$

Thanks!

Thank you for your attention!

References I

S. Bochner. *Lectures on Fourier integrals*. Number 42. Princeton University Press, 1959.

P. Hennig and C. J. Schuler. Entropy search for information-efficient global optimization. *JMLR*, 13, 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 25*. Curran Associates, Inc., 2014.

J. M. Hernández-Lobato, M. A. Gelbart, M. W. Hoffman, R. P. Adams, and Z. Ghahramani. Predictive entropy search for bayesian optimization with unknown constraints. *arXiv:1502.05312*, 2015.

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.

References II

J. Mockus, V. Tiesis, and A. Zilinskas. The application of Bayesian methods for seeking the extremum. *Towards Global Optimization*, 2, 1978.

C. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

J. Snoek, O. Rippel, K. Swersky, R. Kiros, N. Satish, N. Sundaram, M. Patwary, M. Ali, R. P. Adams, et al. Scalable bayesian optimization using deep neural networks. *arXiv preprint arXiv:1502.05700*, 2015.

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.