Introduction to Bayesian Optimization

Javier González

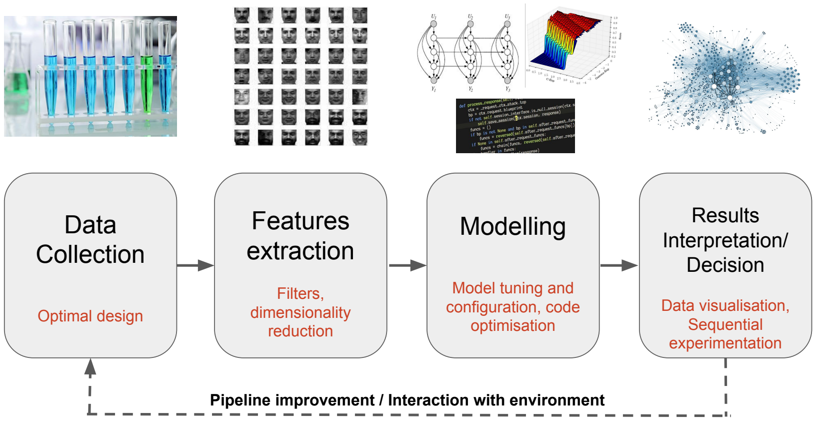
Masterclass, 7-February, 2107 @Lancaster University

Section I: Introduction to Bayesian Optimization

- What is BayesOpt and why it works?
- Relevant things to know.

Data Science pipeline/Autonomous System

Challenges and needs for automation



Data Collection
Optimal design
Features extraction
Filters, dimensionality reduction
Modelling
Model tuning and configuration, code optimisation
Results Interpretation/ Decision
Data visualisation, Sequential experimentation
Pipeline improvement / Interaction with environment

Experimental Design - Uncertainty Quantification

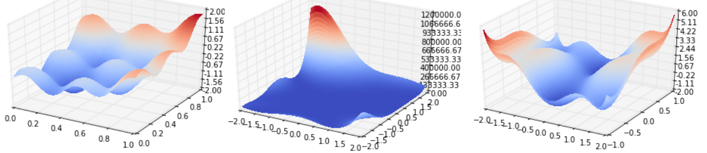
Can we automate/simplify the process of designing complex experiments?

Emulator - Simulator - Physical system

Global optimization

Consider a 'well behaved' function $f : \mathcal{X} \to \mathbb{R}$ where $\mathcal{X} \subseteq \mathbb{R}^D$ is a bounded domain.

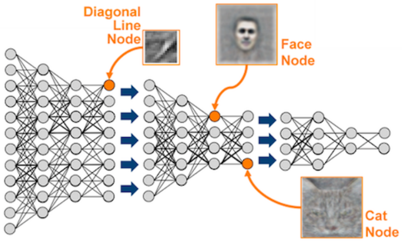
$$x_M = \arg\min_{x \in \mathcal{X}} f(x).$$



- f is explicitly unknown and multimodal.
- Evaluations of f may be perturbed.
- Evaluations of f are expensive.

Expensive functions, who doesn't have one?

Parameter tuning in ML algorithms.

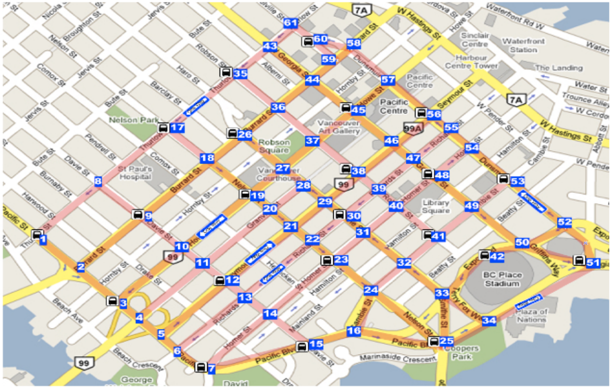


- Number of layers/units per layer
- Weight penalties
- Learning rates, etc.

Figure source: http://theanalyticsstore.com/deep-learning

Expensive functions, who doesn't have one?

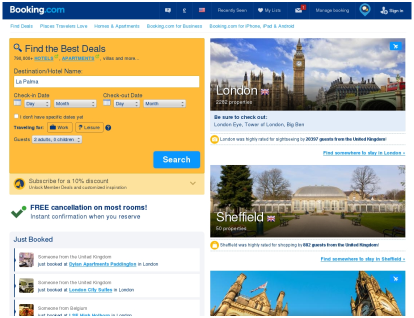
Active Path Finding in Middle Level



Optimise the location of a sequence of waypoints in a map to navigate from a location to a destination.

Expensive functions, who doesn’t have one?

Tuning websites with A/B testing



FREE cancellation on most rooms!
Instant confirmation when you reserve

Just Booked

Someone from the United Kingdom
just booked at Dylan Apartments Paddington in London

Someone from the United Kingdom
just booked at London City Suites in London

Someone from Belgium
just booked at LSE High Holborn in London

Be sure to check out:
London Eye, Tower of London, Big Ben

London was highly rated for sightseeing by **20397 guests from the United Kingdom**!

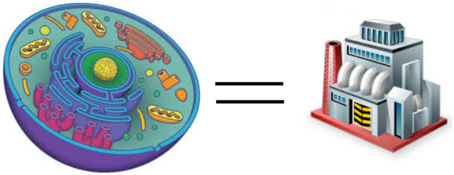
Sheffield was highly rated for shopping by **882 guests from the United Kingdom**!

Optimize the web design to maximize sign-ups, downloads, purchases, etc.

Expensive functions, who doesn't have one?

[González, Lonworth, James and Lawrence, NIPS workshops 2014, 2015]

Design of experiments: gene optimization



- Use mammalian cells to make protein products.
- Control the ability of the cell-factory to use synthetic DNA.

Optimize genes (ATTGGTUGA...) to best enable the cell-factory to operate most efficiently.

Expensive functions, who doesn't have one?

Many other problems:

- Robotics, control, reinforcement learning.
- Scheduling, planning
- compilers, hardware, software?
- Intractable likelihoods.

What to do?

Option 1: Use previous knowledge

To select the parameters at hand. Perhaps not very scientific but still in use...

What to do?

Option 2: Grid search?

If f is L-Lipschitz continuous and we are in a noise-free domain to guarantee that we propose some $x_{M,n}$ such that

$$f(x_M) - f(x_{M,n}) \leq \epsilon$$

we need to evaluate f on a D-dimensional unit hypercube:

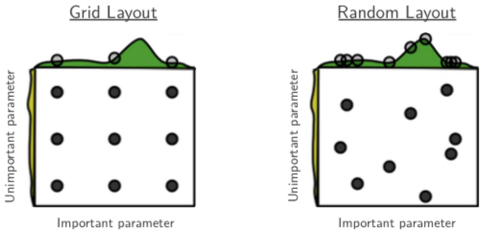
$$(L/\epsilon)^D evaluations!$$

Example: $(10/0.01)^5 = 10e14$...
... but function evaluations are very expensive!

What to do?

Option 3: Random search?

We can sample the space uniformly [Bergstra and Bengio 2012]



Better than grid search in various senses but still expensive to guarantee good coverage.

What to do?

Key question:

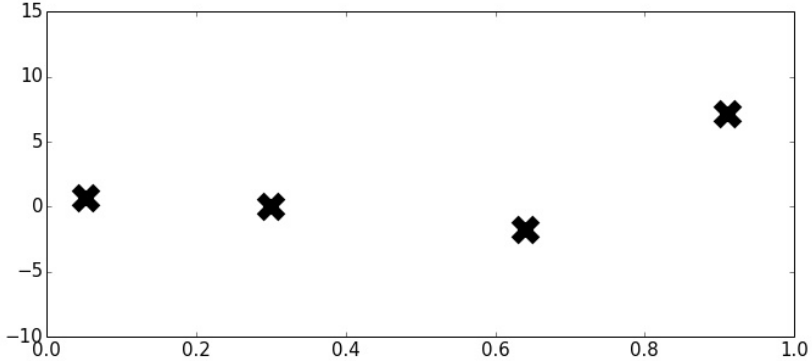
Can we do better?

Problem (the audience is encouraged to participate!)

- Find the optimum of some function f in the interval [0,1].
- f is L-Lipchitz continuous and differentiable.
- Evaluations of f are exact and we have 4 of them!

Situation

We have a few function evaluations

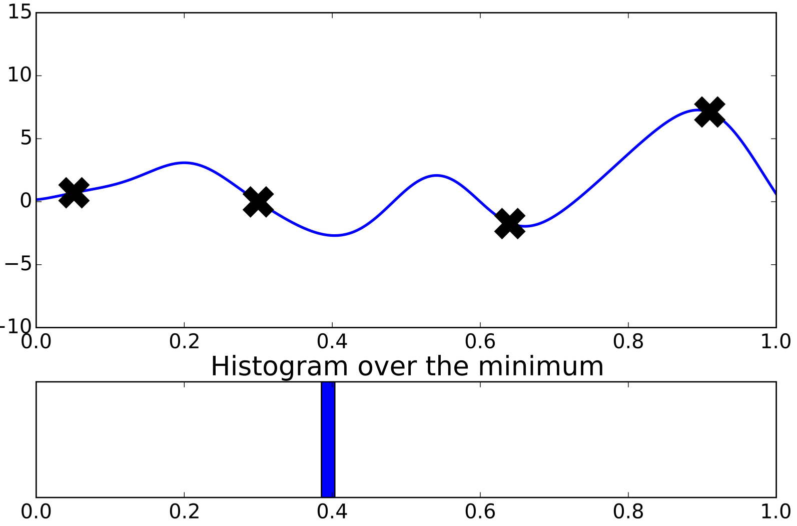


Where is the minimum of f?

Where should the take the next evaluation?

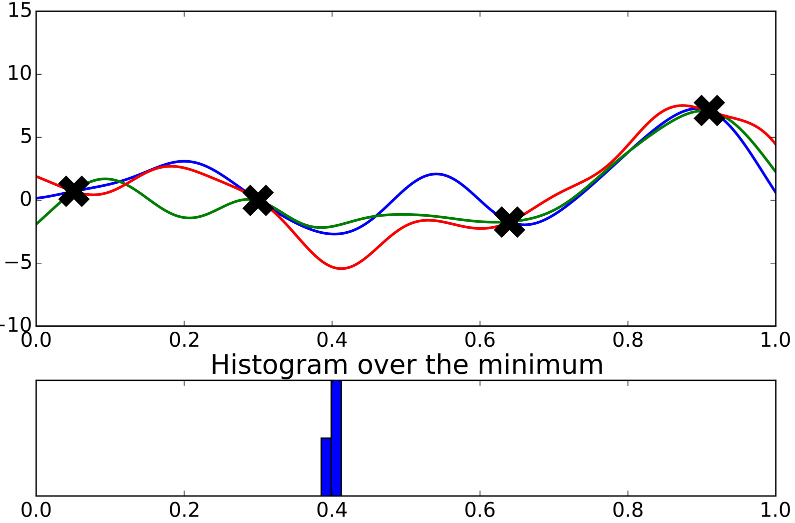
Intuitive solution

One curve



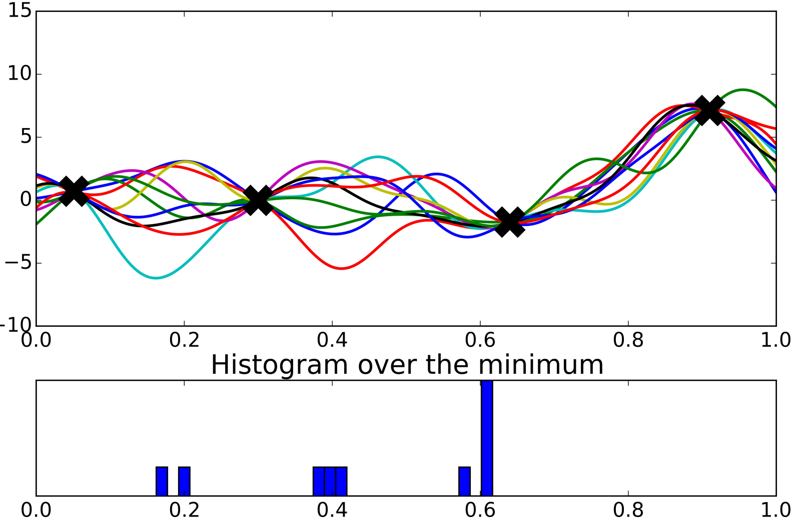
Intuitive solution

Three curves



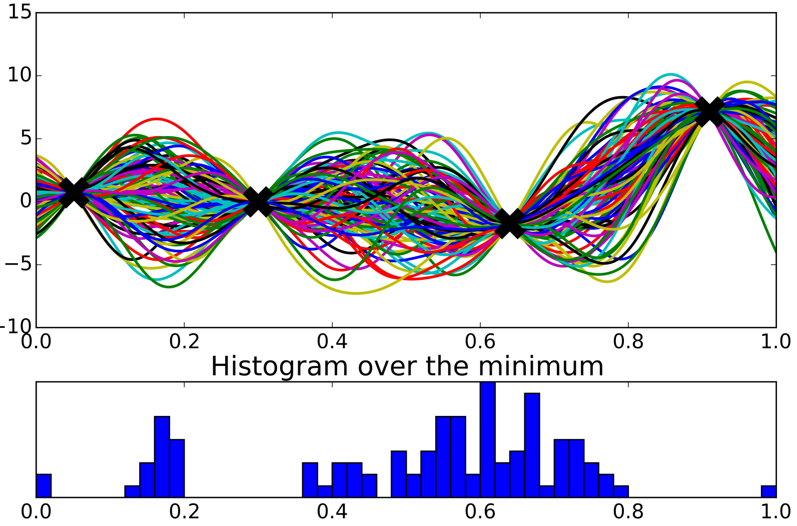
Intuitive solution

Ten curves



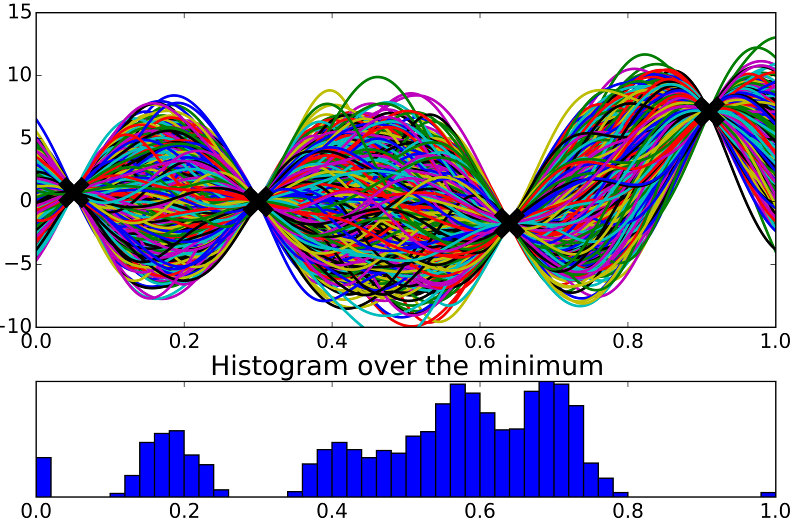
Intuitive solution

Hundred curves



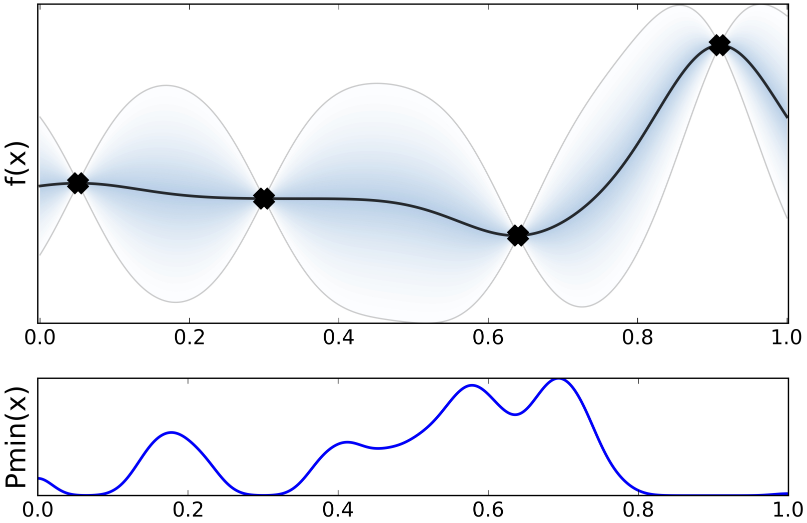
Intuitive solution

Many curves



Intuitive solution

Infinite curves



General idea: surrogate modelling

1. Use a surrogate model of f to carry out the optimization.

2. Define an utility function to collect new data points satisfying some optimality criterion: *optimization* as *decision*.

3. Study *decision* problems as *inference* using the surrogate model: use a probabilistic model able to calibrate both, epistemic and aleatoric uncertainty.

Uncertainty Quantification

Utility functions

The utility should represent our design goal:.

1. Active Learning and experimental design: Maximize the differential entropy of the posterior distribution $p(f|X, y)$ (D-optimality in experimental design).

2. Minimize the loss in a sequence $x_1, \ldots, x_n$

$$r_N = \sum_{n=1}^{N} f(x_n) - N f(x_M)$$

(1) does to a lot exploration whereas (2) encourages exploitation about the minimum of f.

Bayesian Optimisation

[Mockus, 1978]

Methodology to perform global optimisation of multimodal black-box functions.

1. Choose some *prior measure* over the space of possible objectives f.

2. Combine prior and the likelihood to get a *posterior measure* over the objective given some observations.

3. Use the posterior to decide where to take the next evaluation according to some *acquisition/loss function*.

4. Augment the data.

Iterate between 2 and 4 until the evaluation budget is over.

Surrogate model: Gaussian process

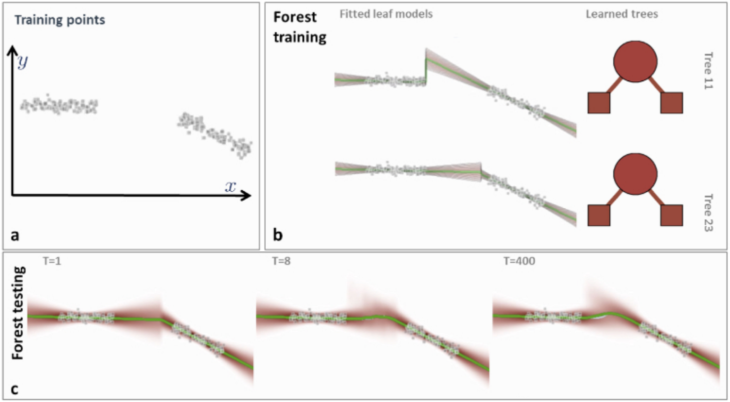
Default Choice: Gaussian processes [Rasmunsen and Williams, 2006]

Infinite-dimensional probability density, such that each linear finite-dimensional restriction is multivariate Gaussian.

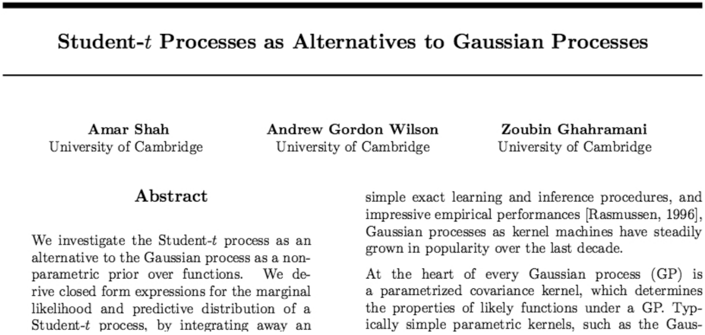
- Model $f(x) \sim \mathcal{GP}(\mu(x), k(x, x'))$ is determined by the *mean function* $m(x)$ and *covariance function* $k(x, x'; \theta)$.
- Posterior mean $\mu(x; \theta, \mathcal{D})$ and variance $\sigma(x; \theta, \mathcal{D})$ can be *computed explicitly* given a dataset $\mathcal{D}$.

Other models are also possible: Random Forrest

[Criminisi et al, 2011]



Other models are also possible: t-Student processes



Student-t Processes as Alternatives to Gaussian Processes

Amar Shah
University of Cambridge

Andrew Gordon Wilson
University of Cambridge

Zoubin Ghahramani
University of Cambridge

Abstract

We investigate the Student-t process as an alternative to the Gaussian process as a non-parametric prior over functions. We derive closed form expressions for the marginal likelihood and predictive distribution of a Student-t process, by integrating away an

simple exact learning and inference procedures, and impressive empirical performances [Rasmussen, 1996], Gaussian processes as kernel machines have steadily grown in popularity over the last decade.

At the heart of every Gaussian process (GP) is a parametrized covariance kernel, which determines the properties of likely functions under a GP. Typically simple parametric kernels, such as the Gaus-

Exploration vs. exploitation

Welcome to El Jalapeño Mexican Grill
Choose Your Favorite Fillings!
Lightly seasoned white or brown rice, organic black or pinto beans, green peppers, onions, mild/hot/fire or Verde Salsa, crunchy lettuce, shredded cheese, sour cream.
Add fresh chips, jalapeño cheese, salsa & fountain drink to ANY main item for $.
Burrito
HOUSE BURRITO
wrapped in 12" Tortilla
Favorite Fillings + Veggie, Chicken or Beef $.
Favorite Fillings + Steak $.
BURRITO PLATE
excludes 12" Tortilla
Favorite Fillings + Veggie, Chicken or Beef $.
Favorite Fillings + Steak $.
Quesadilla
HOUSE QUESADILLA
12" flour tortilla w/ El Jalapeño's special sauce
Cheese Only $.
Beef or chicken $.
Steak $.
Tacos
HOUSE TACOS
3 crispy corn or soft flour tortillas
Favorite Fillings + Veggie, Chicken or Beef $.
Favorite Fillings + Steak $.
CRISPY TACO SALAD
(crunchy tortilla bowl)
Favorite Fillings + Veggie, Chicken or Beef $.
Favorite Fillings + Steak $.
Los Niños
Each meal includes juice & Los Niños size chips & salsa
QUESADILLA (10" flour tortilla)
Los Niños choice of cheese, chicken or steak $.
TACOS
Favorite Fillings + Los Niños choice of beef, chicken or steak, lettuce & cheese in 2 crispy corn or soft tacos $.
House Specials
TRIPLE BEEF STEAK FAJITA BURRITO
Favorite Fillings + 3 scoops of beef steak, fire-roja sauce $.
CHALUPA
2 fried flat bread & 1 crispy or soft taco
Veggie, chicken, or Beef $.
Steak $.
JALAPEÑO NACHOS
Veggie, chicken or Beef $.
Steak $.
Extras
Add a taco (crispy or soft) $.
Fresh chips, jalapeño cheese, & Salsa $.
Add guacamole $.
Extra salsa $.
Extra jalapeño cheese $.
Chips only $.

Bayesian optimization explains human active search

[Borji and Itti, 2013]

Exploration vs. exploitation

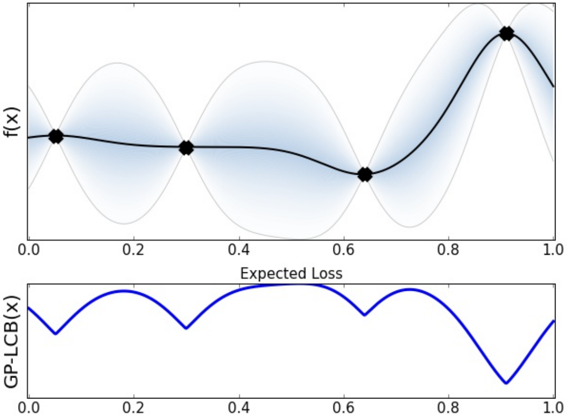
Picture source: http://peakdistrictcycleways.co.uk

GP Upper (lower) Confidence Band

[Srinivas et al., 2010]

Direct balance between exploration and exploitation:

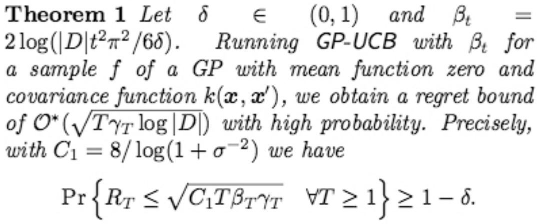
$$\alpha_{LCB}(\mathbf{x};\theta,\mathcal{D}) = -\mu(\mathbf{x};\theta,\mathcal{D}) + \beta_t\sigma(\mathbf{x};\theta,\mathcal{D})$$



GP Upper (lower) Confidence Band

[Srinivas et al., 2010]

- In noiseless cases, it is a lower bound of the function to minimize.
- This allows to computer a bound on how close we are to the minimum.
- Optimal choices available for the 'regularization parameter'.



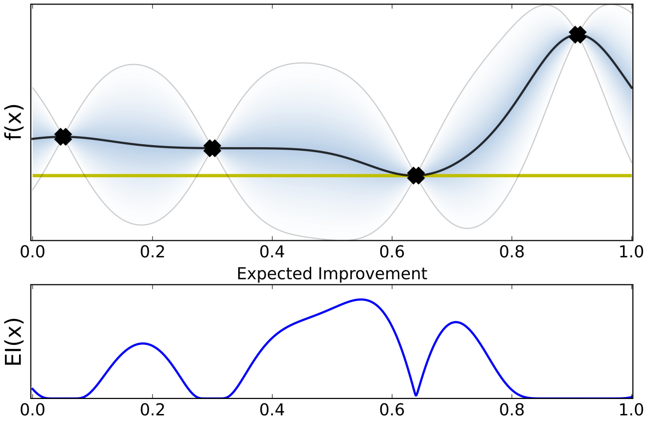
Theorem 1 *Let* $\delta \in (0,1)$ *and* $\beta_t = 2\log(|D|t^2\pi^2/6\delta)$. *Running GP-UCB with* β_t *for a sample* f *of a GP with mean function zero and covariance function* $k(\boldsymbol{x}, \boldsymbol{x}')$, *we obtain a regret bound of* $\mathcal{O}^*(\sqrt{T\gamma_T \log|D|})$ *with high probability. Precisely, with* $C_1 = 8/\log(1+\sigma^{-2})$ *we have*

$$\Pr\left\{R_T \leq \sqrt{C_1 T \beta_T \gamma_T} \quad \forall T \geq 1\right\} \geq 1 - \delta.$$

Expected Improvement

[Jones et al., 1998]

$$\alpha_{EI}(\mathbf{x};\theta,\mathcal{D}) = \int_y \max(0, y_{best} - y)p(y|\mathbf{x};\theta,\mathcal{D})dy$$



Expected Improvement

[Jones et al., 1998]

- Perhaps the most used acquisition.
- Explicit for available for Gaussian posteriors.
- It is too greedy in some problems. It is possible to make more explorative adding a 'explorative' parameter

$$\alpha_{EI}(\mathbf{x};\theta,\mathcal{D}) = \sigma(\mathbf{x};\theta,\mathcal{D})(\gamma(x)\Phi(\gamma(x))) + \mathcal{N}(\gamma(x);0,1).$$

where

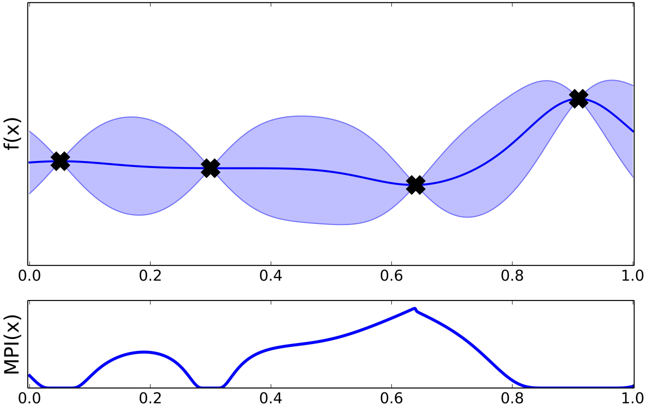
$$\gamma(x) = \frac{f(x_{best}) - \mu(\mathbf{x};\theta,\mathcal{D}) + \psi}{\sigma(\mathbf{x};\theta,\mathcal{D})}.$$

Maximum Probability of Improvement

[Hushner, 1964]

$$\gamma(\mathbf{x}) = \sigma(\mathbf{x}; \theta, \mathcal{D})^{-1}(\mu(\mathbf{x}; \theta, \mathcal{D}) - y_{best})$$

$$\alpha_{MPI}(\mathbf{x}; \theta, \mathcal{D}) = p(f(\mathbf{x}) < y_{best}) = \Phi(\gamma(\mathbf{x}))$$



Maximum Probability of Improvement

[Hushner, 1964]

- First used acquisition: very intuitive.
- Less used in practice.
- Explicit for available for Gaussian posteriors.

$$\alpha_{MPI}(\mathbf{x}; \theta, \mathcal{D}) = \Phi(\gamma(x))).$$

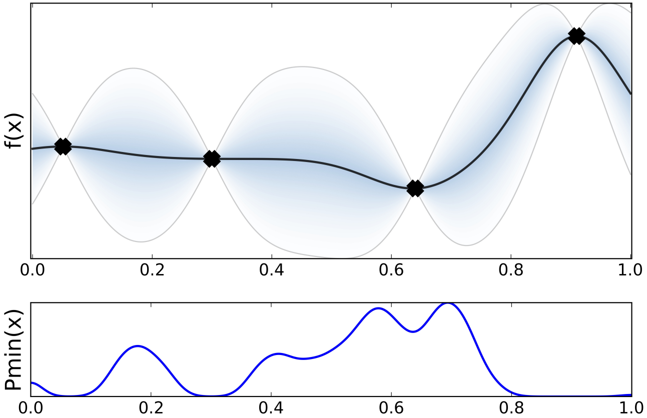
where

$$\gamma(x) = \frac{f(x_{best}) - \mu(\mathbf{x}; \theta, \mathcal{D}) + \psi}{\sigma(\mathbf{x}; \theta, \mathcal{D})}.$$

Information-theoretic approaches

[Hennig and Schuler, 2013; Hernández-Lobato et al., 2014]

$$\alpha_{ES}(\mathbf{x};\theta,\mathcal{D}) = H[p(x_{min}|\mathcal{D})] - \mathbb{E}_{p(y|\mathcal{D},\mathbf{x})}[H[p(x_{min}|\mathcal{D}\cup\{\mathbf{x},y\})]]$$



Information-theoretic approaches

Uses the distribution of the minimum

$$p_{min}(x) \equiv p[x = \arg\min f(x)] = \int_{f:I\to\Re} p(f) \prod_{\substack{\tilde{x}\in I \\ \tilde{x}\neq x}} \theta[f(\tilde{x}) - f(x)] df$$

where θ is the Heaviside's step function. No closed form!

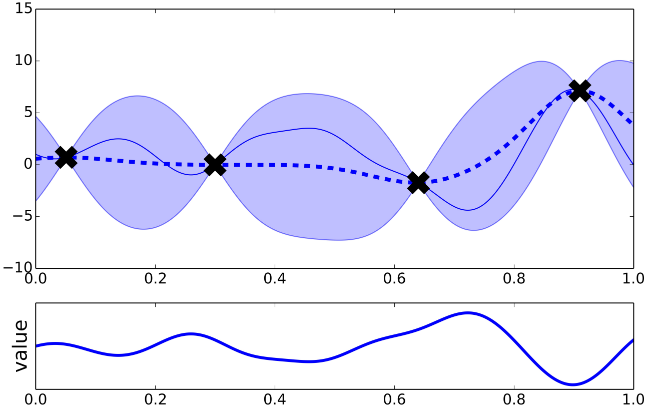
Use Thomson sampling to approximate the distribution.
Generate many sample paths from the GP, optimize them to take samples from $p_{min}(x)$.

Thomson sampling

Probability matching

$$\alpha_{THOMSON}(\mathbf{x}; \theta, \mathcal{D}) = g(\mathbf{x})$$

$g(\mathbf{x})$ is sampled form $\mathcal{GP}(\mu(x), k(x, x'))$



Thompson sampling

Probability matching [Rahimi and B. Recht, 2007]

- It is easy to generate posterior samples of a GP at a finite set of locations.
- More difficult is to generate ‘continuous’ samples.

Possible using the Bochner’s lemma: existence of the Fourier dual of k, $s(\omega)$ which is equal to the spectral density of k

$$k(x, x') = \nu \mathbb{E}_{\omega} \left[e^{-i\omega^T (x-x')} \right] = 2\nu \mathbb{E}_{\omega,b} \left[\cos(\omega x^T + b) \cos(\omega x^T + b) \right]$$

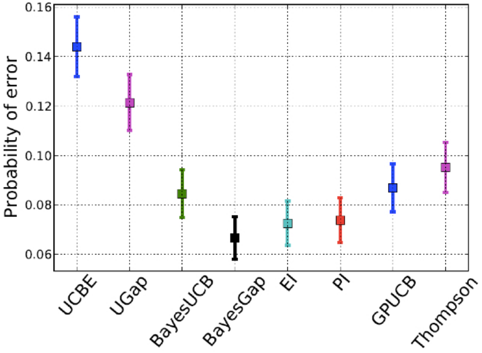
With sampling and this lemma (taking $p(w) = s(\omega)/\nu$ and $b \sim \mathcal{U}[0, 2\pi]$) we can construct a feature based approximation for sample paths of the GP.

$$k(x, x') \approx \frac{\nu}{m} \sum_{i=1}^{m} e^{-i\omega^{(i)T} x} e^{-i\omega^{(i)T} x'}$$

The choice of utility matters

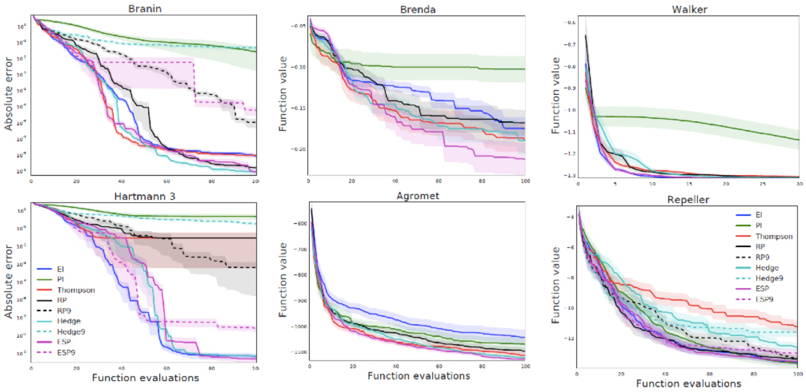
[Hoffman, Shahriari and de Freitas, 2013]

The choice of the utility may change a lot the result of the optimisation.



The choice of utility in practice

[Hoffman, Shahriari and de Freitas, 2013]



The best utility depends on the problem and the level of exploration/exploitation required.

Illustration of BO

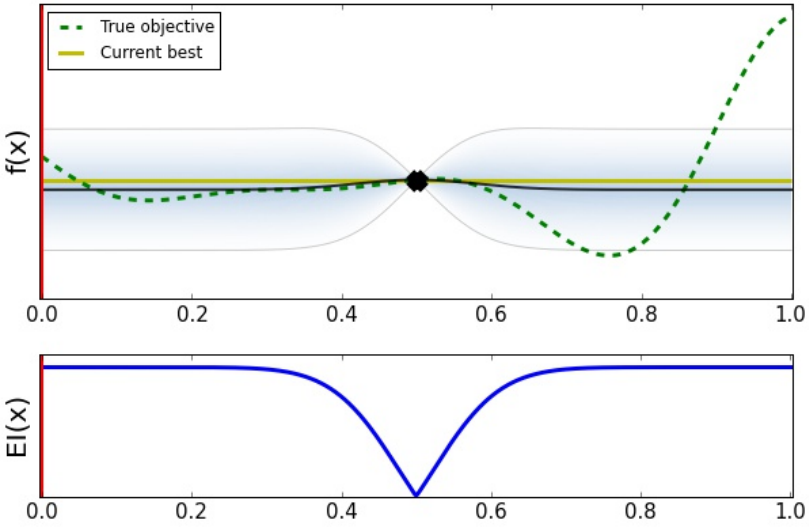


Illustration of BO

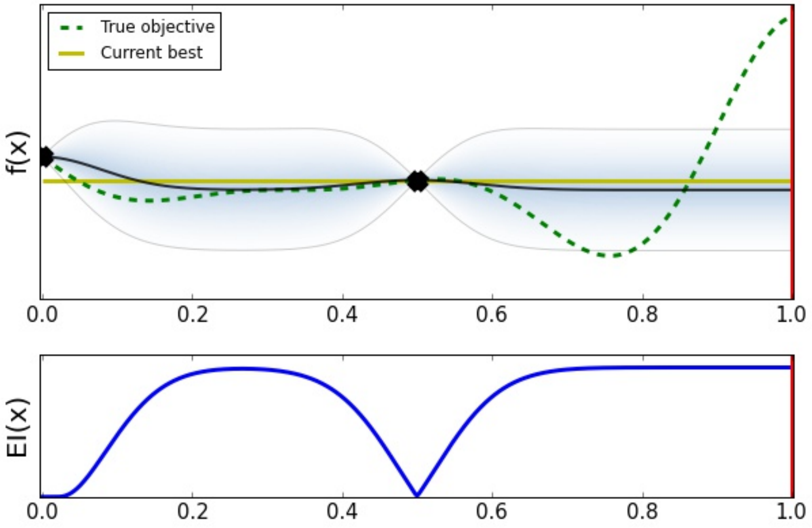


Illustration of BO

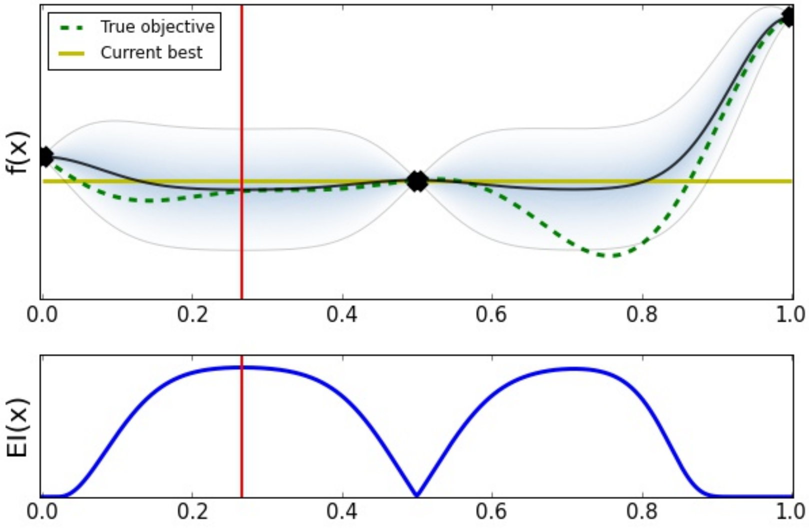


Illustration of BO

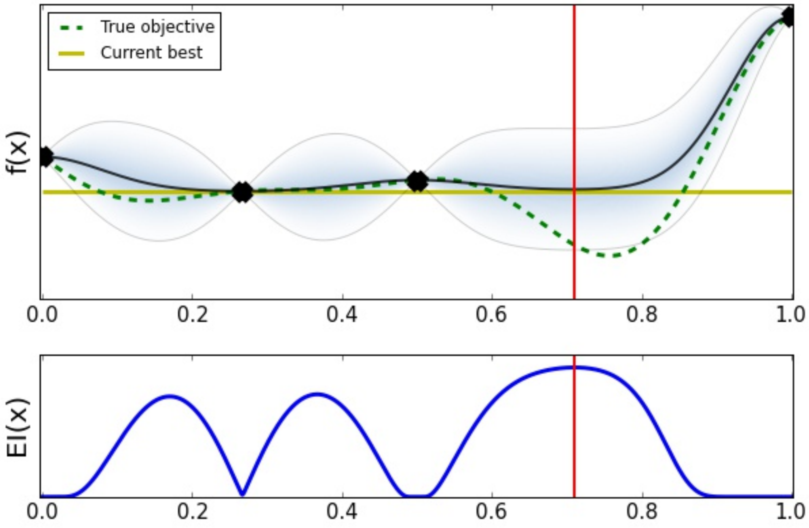


Illustration of BO

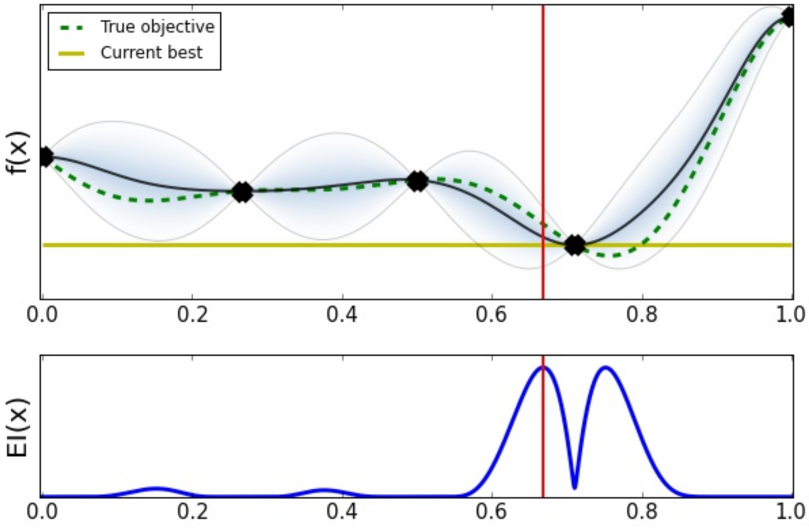


Illustration of BO

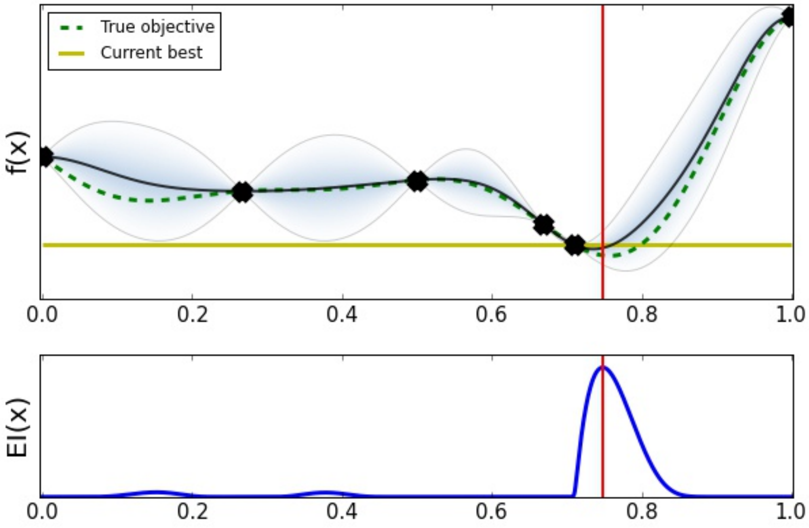


Illustration of BO

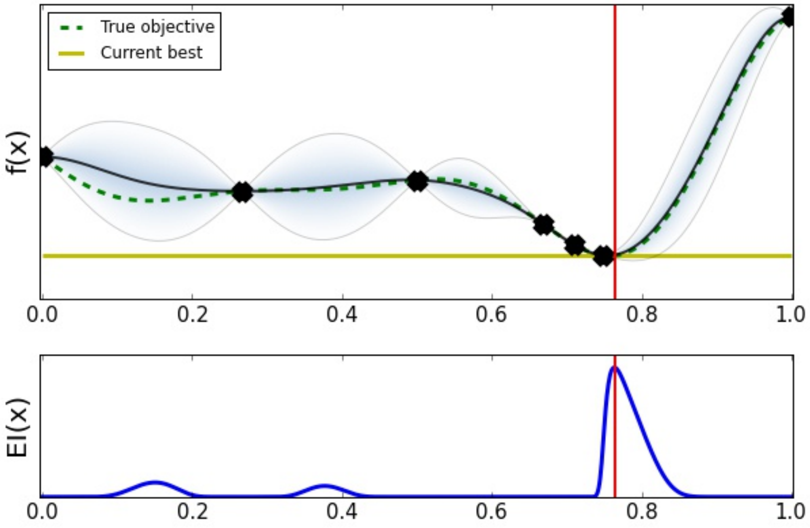
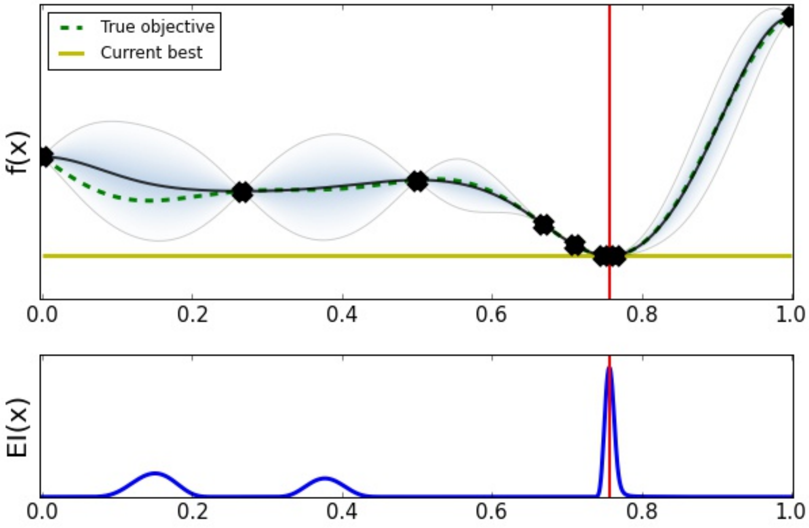


Illustration of BO



Bayesian Optimization

As a 'mapping' between two problems

BO is an strategy to transform the problem

$$x_M = \arg\min_{x \in \mathcal{X}} f(x)$$
$$\textit{solvable}!$$

into a series of problems:

$$x_{n+1} = \arg\max_{x \in \mathcal{X}} \alpha(x; \mathcal{D}_n, \mathcal{M}_n)$$
$$\textit{solvable}!$$

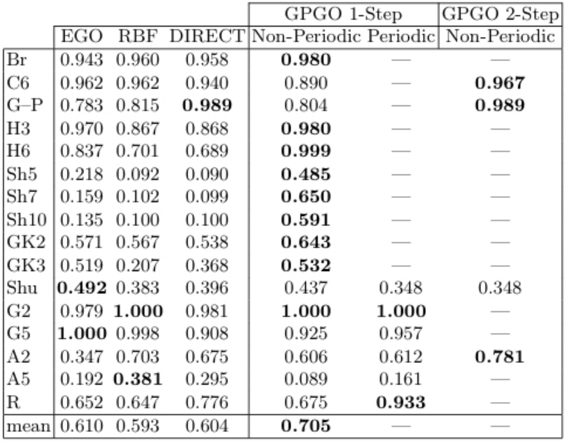
where now:

- $\alpha(x)$ is inexpensive to evaluate.
- The gradients of $\alpha(x)$ are typically available.
- Still need to find x_{n+1}.

BO vs other methods

[Osborne et al, 2009]

Bayesian optimization works better in practice!



				GPGO 1-Step		GPGO 2-Step
	EGO	RBF	DIRECT	Non-Periodic	Periodic	Non-Periodic
Br	0.943	0.960	0.958	**0.980**	—	—
C6	0.962	0.962	0.940	0.890	—	**0.967**
G–P	0.783	0.815	**0.989**	0.804	—	**0.989**
H3	0.970	0.867	0.868	**0.980**	—	—
H6	0.837	0.701	0.689	**0.999**	—	—
Sh5	0.218	0.092	0.090	**0.485**	—	—
Sh7	0.159	0.102	0.099	**0.650**	—	—
Sh10	0.135	0.100	0.100	**0.591**	—	—
GK2	0.571	0.567	0.538	**0.643**	—	—
GK3	0.519	0.207	0.368	**0.532**	—	—
Shu	**0.492**	0.383	0.396	0.437	0.348	0.348
G2	0.979	**1.000**	0.981	**1.000**	**1.000**	—
G5	**1.000**	0.998	0.908	0.925	0.957	—
A2	0.347	0.703	0.675	0.606	0.612	**0.781**
A5	0.192	**0.381**	0.295	0.089	0.161	—
R	0.652	0.647	0.776	0.675	**0.933**	—
mean	0.610	0.593	0.604	**0.705**	—	—

Recap

- Bayesian optimization is a way of encoding our beliefs about a property of a function (the minimum)
- Two key elements: the model and the acquisition function.
- Many choices in both cases, especially in terms of the acquisition function used.
- The key is to find a good balance between exploration and exploitation.

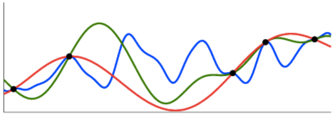
Main issues

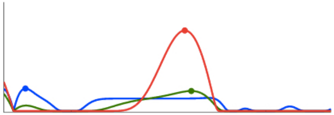
- What to do with the hyper-parameters of the model?
- How to select points to initialize the model?
- How to optimize the acquisition function?

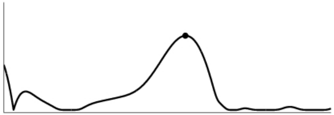
BO independent of the parameters of the GP.

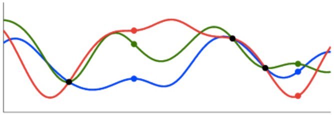
[Snoek et al. 2012]

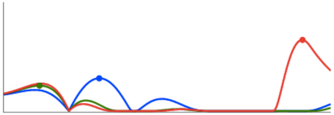
Integrate out across parameter values or location outputs.

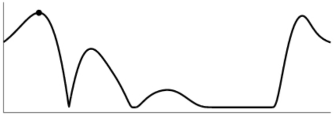

(a) Posterior samples under varying hyperparameters


(b) Expected improvement under varying hyperparameters


(c) Integrated expected improvement


(a) Posterior samples after three data


(b) Expected improvement under three fantasies


(c) Expected improvement across fantasies

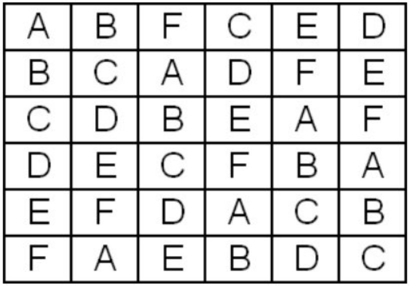
How to initialise the model?

- One point in the centre of the domain.
- Uniformly selected random locations.
- Latin design.
- Halton sequences.
- Determinantal point processes.

The idea is always to start at some locations trying to minimise the initial model uncertainty.

Latin design

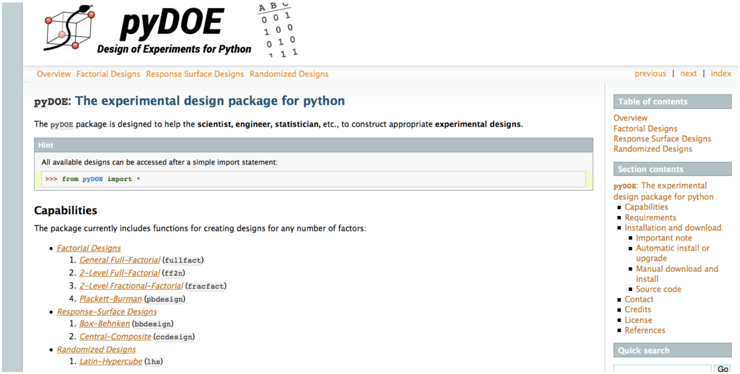
$n \times n$ array filled with n different symbols, each occurring exactly once in each row and exactly once in each column.



A	B	F	C	E	D
B	C	A	D	F	E
C	D	B	E	A	F
D	E	C	F	B	A
E	F	D	A	C	B
F	A	E	B	D	C

pyDOE

Python framework for standard experimental design



Overview Factorial Designs Response Surface Designs Randomized Designs previous | next | index

pyDOE: The experimental design package for python

The pyDOE package is designed to help the **scientist, engineer, statistician,** etc., to construct appropriate **experimental designs**.

Hint

All available designs can be accessed after a simple import statement:

```
>>> from pyDOE import *
```

Capabilities

The package currently includes functions for creating designs for any number of factors:

- *Factorial Designs*
  1. *General Full-Factorial* (`fullfact`)
  2. *2-Level Full-Factorial* (`ff2n`)
  3. *2-Level Fractional-Factorial* (`fracfact`)
  4. *Plackett-Burman* (`pbdesign`)
- *Response-Surface Designs*
  1. *Box-Behnken* (`bbdesign`)
  2. *Central-Composite* (`ccdesign`)
- *Randomized Designs*
  1. *Latin-Hypercube* (`lhs`)

Table of contents

Overview
Factorial Designs
Response Surface Designs
Randomized Designs

Section contents

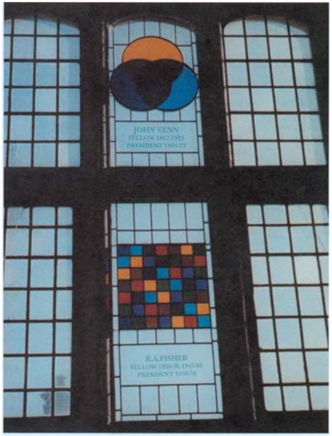
pyDOE: The experimental design package for python
- Capabilities
- Requirements
- Installation and download
  - Important note
  - Automatic install or upgrade
  - Manual download and install
  - Source code
- Contact
- Credits
- License
- References

Quick search

Go

Latin design

Window honors Ronald Fisher. Fisher's student, A. W. F. Edwards, designed this window for Caius College, Cambridge.



Halton sequences

[Halton, 1964]

- Used to generate points in $(0,1) \times (0,1)$
- Sequence that is constructed according to a deterministic method that uses a prime number as its base.

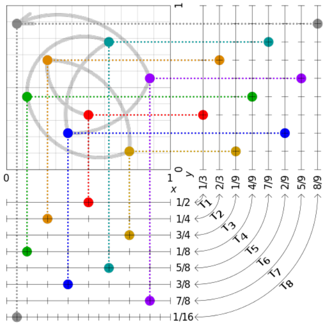
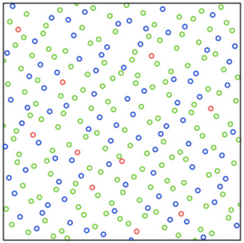


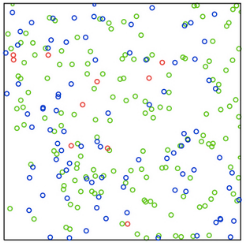
Figure source: Wikipedia

Halton sequences

[Halton, 1964]

Better coverage than random.





Halton *Random*

Figure source: Wikipedia

Determinantal point processes

Kulesza and Taskar, [2012]

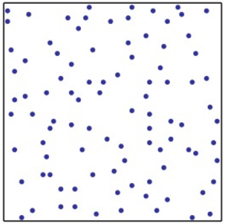
We say that X is a ‘determinantal point process’ on Λ with kernel K if it is a simple point process on Λ with a joint intensity or ‘correlation function’ given by

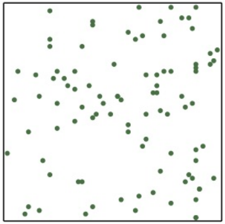
$$\rho_n(x_1, \ldots, x_n) = det(K(x_i, x_j)_{1 \leq i,j \leq n})$$

- Probability measures over subsets.
- Possible to characterise the samples in terms of quality and diversity.

Determinantal point processes

Kulesza and Taskar, [2012]


DPP


Independent

Key idea:

$$\begin{aligned}\mathcal{P}(i,j \in \boldsymbol{Y}) &= \begin{vmatrix} K_{ii} & K_{ij} \\ K_{ji} & K_{jj} \end{vmatrix} \\ &= K_{ii}K_{jj} - K_{ij}K_{ji} \\ &= \mathcal{P}(i \in \boldsymbol{Y})\mathcal{P}(j \in \boldsymbol{Y}) - K_{ij}^2 \,.\end{aligned}$$

Determinantal point processes

Kulesza and Taskar, [2012]

Methods to optimise the acquisition function

This may not be easy.

- Gradient descent methods: Conjugate gradient, BFGS, etc.
- Lipschitz based heuristics: DIRECT.
- Evolutionary algorithms: CMA.

Some of these methods can also be used to directly optimize f

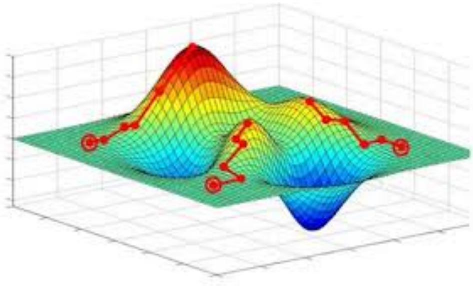
Gradient descent

[Avriel,2013], but many others

```
Algorithm 2: Gradient Descent
input  : $f : \mathbb{R}^n \to \mathbb{R}$ a differentiable function
         $\mathbf{x}^{(0)}$ an initial solution
output: $\mathbf{x}^\star$, a local minimum of the cost function $f$.
begin
    $k \leftarrow 0$ ;
    while STOP-CRIT and $(k < k_{max})$ do
        $\mathbf{x}^{(k+1)} \leftarrow \mathbf{x}^{(k)} - \alpha^{(k)} \nabla f(\mathbf{x})$ ;
        with $\alpha^{(k)} = \arg\min_{\alpha \in \mathbb{R}_+} f\big(\mathbf{x}^{(k)} - \alpha \nabla f(\mathbf{x})\big)$ ;
        $k \leftarrow k + 1$ ;
    return $\mathbf{x}^{(k)}$
end
```

We need to know the gradients. This is the case for most acquisitions but not for all of them (PES for instance).

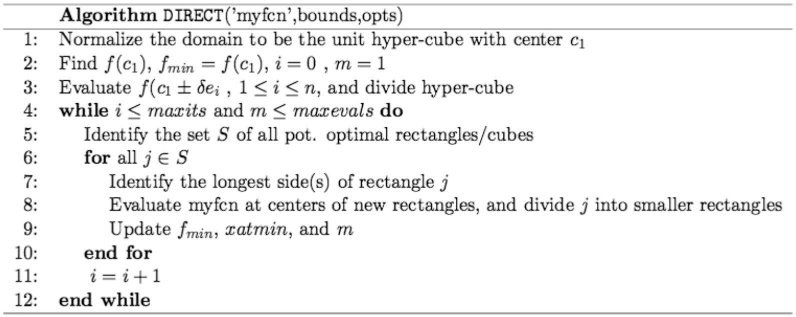
Gradient descent



May fall in local minima if the function is multimodal: multiple initializations.

‘DIviding RECTangles’, DIRECT

[Perttunen at al. 1993]

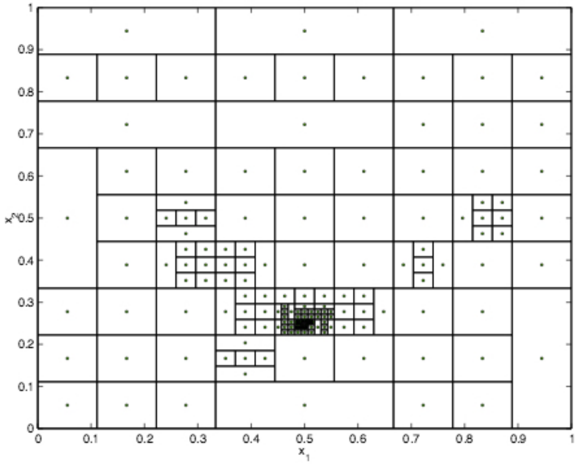


```
Algorithm DIRECT('myfcn',bounds,opts)
Normalize the domain to be the unit hyper-cube with center $c_1$
Find $f(c_1)$, $f_{min} = f(c_1)$, $i = 0$ , $m = 1$
Evaluate $f(c_1 \pm \delta e_i$ , $1 \leq i \leq n$, and divide hyper-cube
while $i \leq maxits$ and $m \leq maxevals$ do
    Identify the set $S$ of all pot. optimal rectangles/cubes
    for all $j \in S$
        Identify the longest side(s) of rectangle $j$
        Evaluate myfcn at centers of new rectangles, and divide $j$ into smaller rectangles
        Update $f_{min}$, $xatmin$, and $m$
    end for
    $i = i + 1$
end while
```

Minimal hypothesis about the acquisition

‘DIviding RECTangles’, DIRECT

[Perttunen at al. 1993]

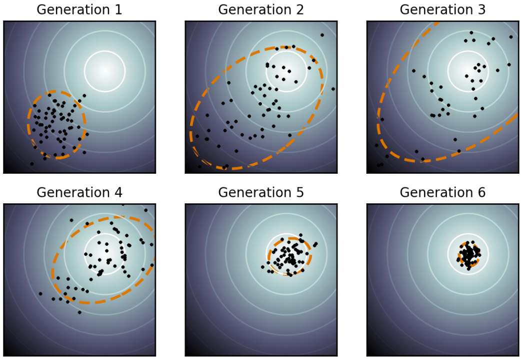


Finds good solution in general and doesn’t need gradient. Not generalizable to non-squared domains.

Covariance Matrix Adaptation, CMA

[Hansen and Ostermeier, 2001].

- Sample for a Gaussian with some mean μ and covariance matrix Σ.
- Select the best points and use them to update μ and Σ.
- Sample form the new Gaussian.



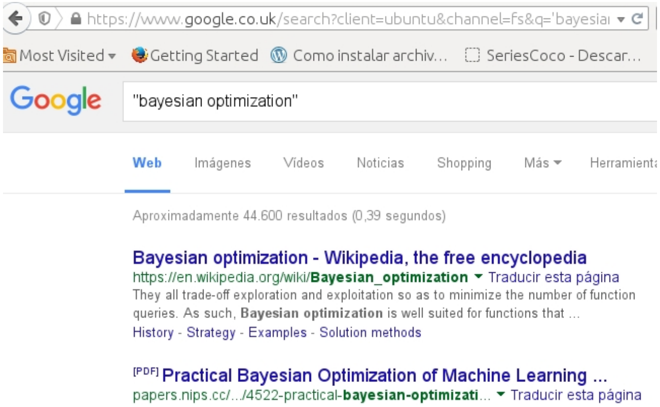
Took a while to start using these ideas in ML

Although in the stats community have been there for a while

- BO depends on its own parameters.
- Lack of software to apply these methods as a black optimization boxes.
- Reduced scalability in dimensions and number of evaluations (this is still a problem).

Practical Bayesian Optimization of Machine Learning Algorithms. Snoek, Larochelle and Adams. NIPS 2012 (Spearmint)

Increasing popular field


https://www.google.co.uk/search?client=ubuntu&channel=fs&q='bayesia

"bayesian optimization"

Aproximadamente 44.600 resultados (0,39 segundos)

Bayesian optimization - Wikipedia, the free encyclopedia
https://en.wikipedia.org/wiki/**Bayesian_optimization** Traducir esta página
They all trade-off exploration and exploitation so as to minimize the number of function queries. As such, **Bayesian optimization** is well suited for functions that ...
History - Strategy - Examples - Solution methods

[PDF] **Practical Bayesian Optimization of Machine Learning ...**
papers.nips.cc/.../4522-practical-**bayesian-optimizati**... Traducir esta página

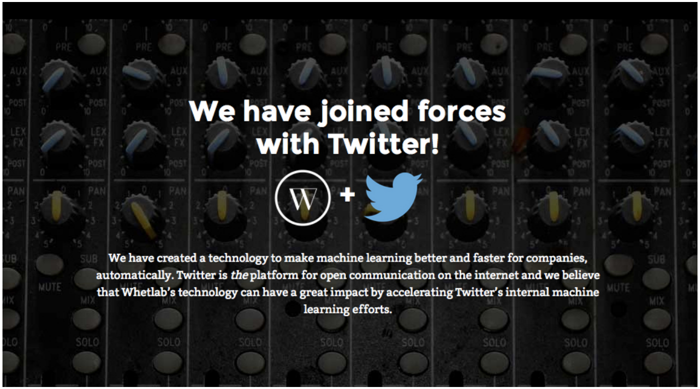
- Hot topic in Machine Learning.
- The BO workshop at NIPS is well stablished and it is a mini-conference itself.

Bayesian optimization now

It has become increasingly popular since it allows to configure algorithms without human intervention.

BO takes to human out of the loop!

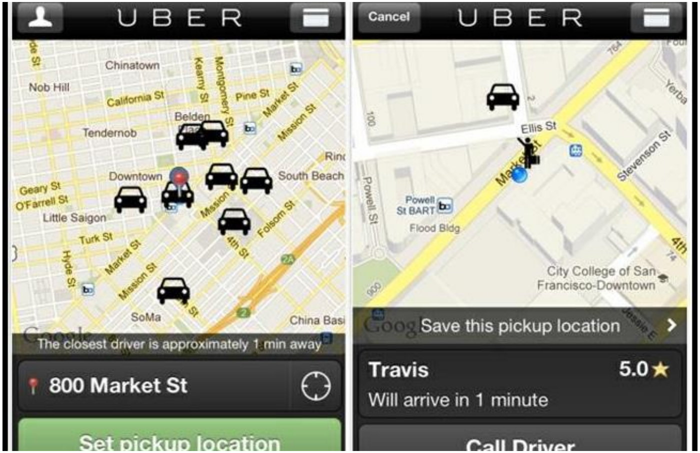
BO in industry: Twitter



We have joined forces with Twitter!

We have created a technology to make machine learning better and faster for companies, automatically. Twitter is *the* platform for open communication on the internet and we believe that Whetlab's technology can have a great impact by accelerating Twitter's internal machine learning efforts.

BO in industry: Uber



Questions?