UNIVERSITY OF CAMBRIDGE

Machine Learning and the Physical World

Lecture 6 : Sequential Decision Making - Bayesian Optimisation

---

Carl Henrik Ek - che29@cam.ac.uk

31st of October, 2024

http://carlhenrik.com

Re-Cap

- The infeasibility of truth and the search for knowledge

Re-Cap

- The infeasibility of truth and the search for knowledge
- Parametrise our ignorance/beliefs

Re-Cap

- The infeasibility of truth and the search for knowledge
- Parametrise our ignorance/beliefs
- Statistical Inference to update our knowledge from experiment

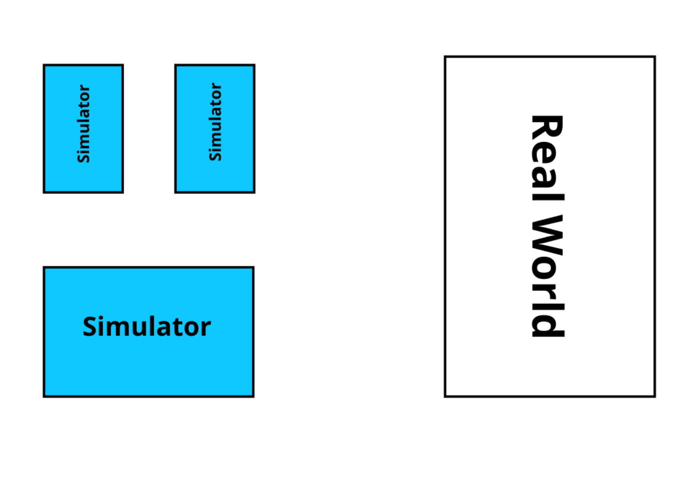
Re-Cap

- The infeasibility of truth and the search for knowledge
- Parametrise our ignorance/beliefs
- Statistical Inference to update our knowledge from experiment
- Emergent Behaviours

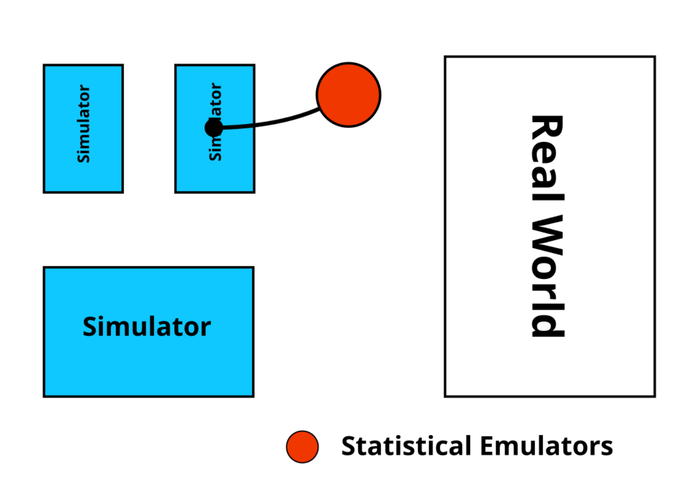
Re-Cap

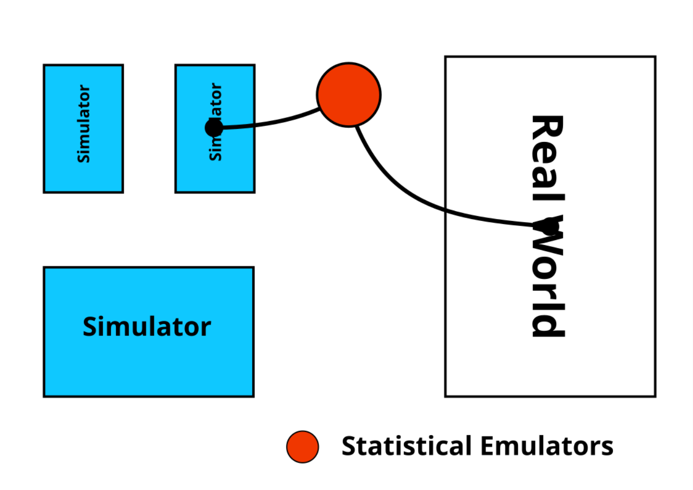
- The infeasibility of truth and the search for knowledge
- Parametrise our ignorance/beliefs
- Statistical Inference to update our knowledge from experiment
- Emergent Behaviours
- Simulation and Emulation

Simulation and Emulation

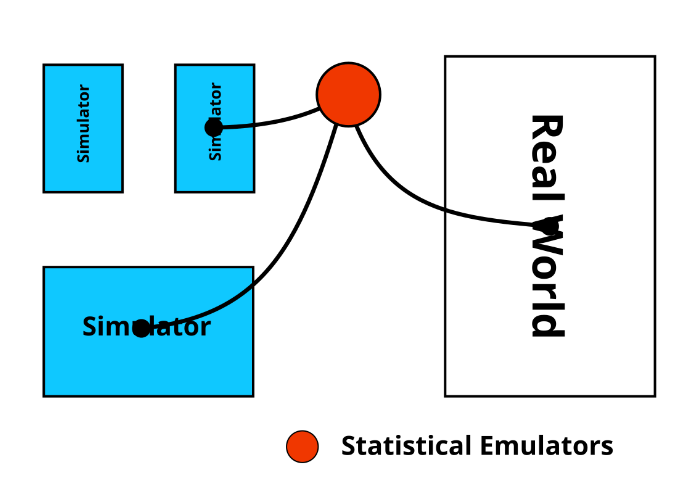


Simulation and Emulation




Simulator
Simulator
Simulator
Real World
Statistical Emulators

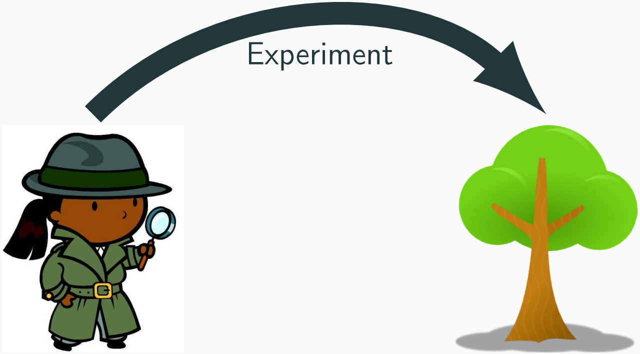
Simulation and Emulation []



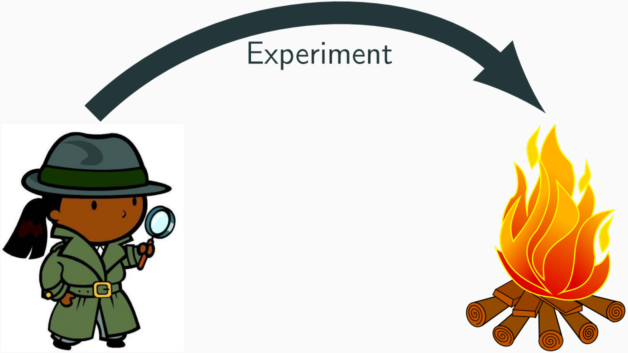
Active Learning

Active Learning

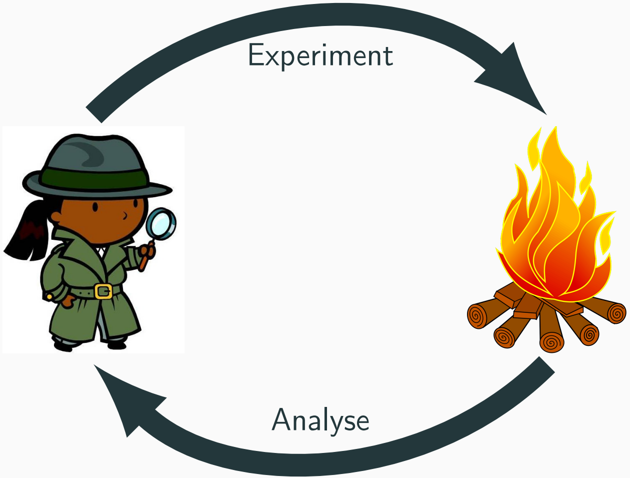
Active Learning



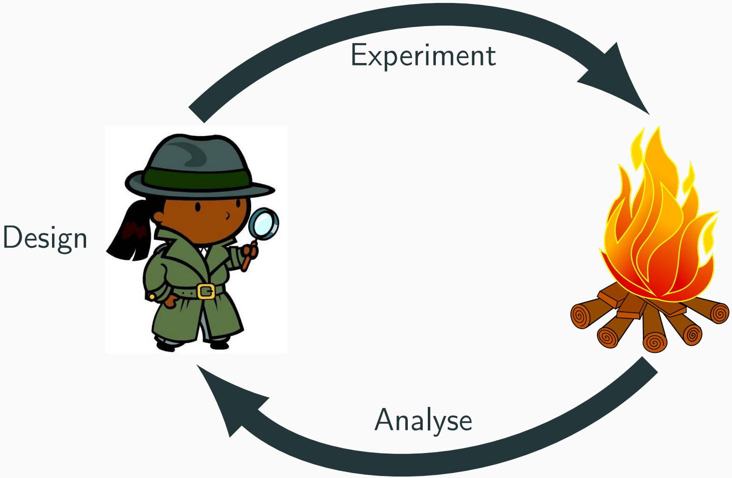
Active Learning



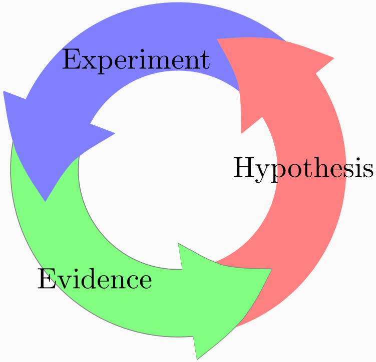
Active Learning



Active Learning

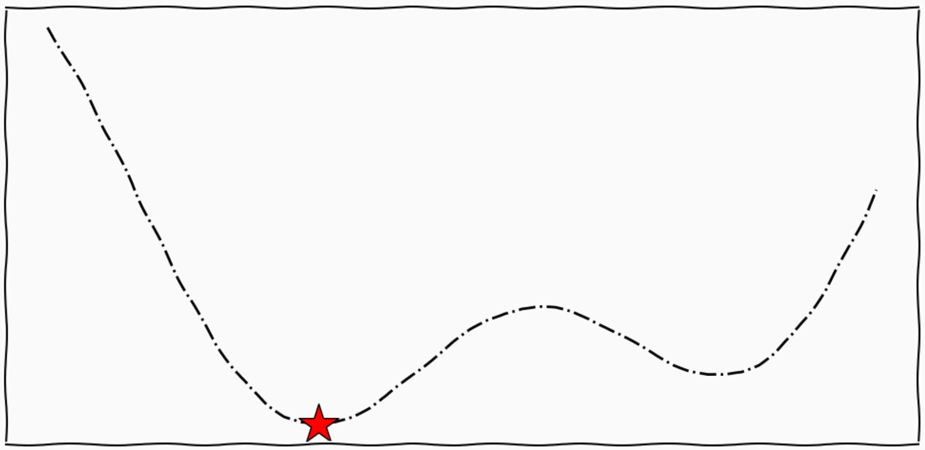


The Scientific Principle



PUB QUIZ

Finding Extremum of a function



Today

Black-Box Optimisation how can we find the extremum of an explicitly unknown function?

Today

Black-Box Optimisation how can we find the extremum of an explicitly unknown function?

Surrogate Models how can we build a model as a surrogate for the unknown function?

Today

Black-Box Optimisation how can we find the extremum of an explicitly unknown function?

Surrogate Models how can we build a model as a surrogate for the unknown function?

Sequential decision making how can we come up with a strategy for sequentially exploring the function?

Bayesian Optimisation

Black-box Optimisation

$$x^{(*)} = \operatorname*{argmin}_{x \in \mathcal{X}} f(x)$$

- $\mathcal{X}$ is a bounded domain

Black-box Optimisation

$$x^{(*)} = \underset{x \in \mathcal{X}}{\operatorname{argmin}} f(x)$$

- $\mathcal{X}$ is a bounded domain
- f is explicitly unknown

Black-box Optimisation

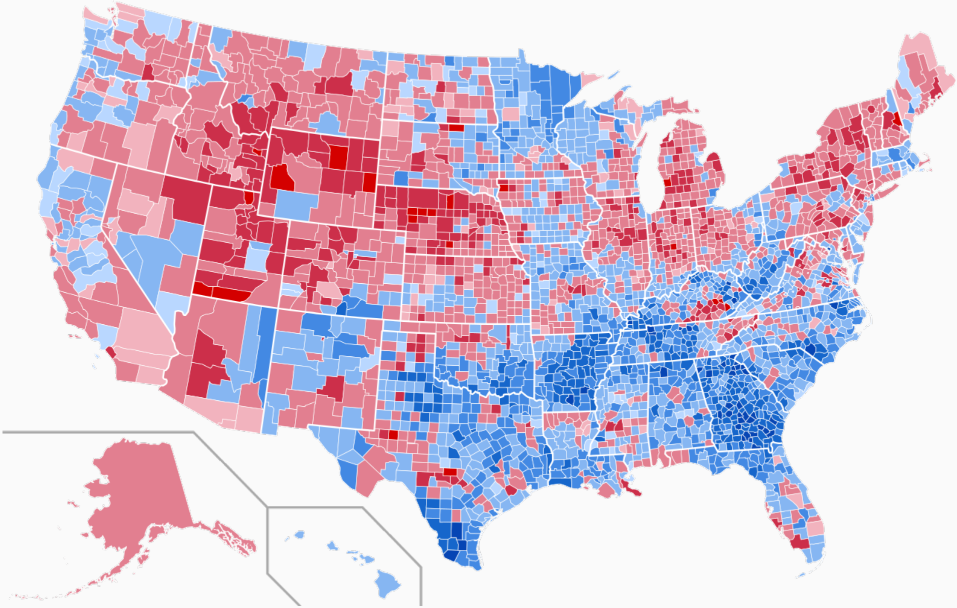
$$x^{(*)} = \underset{x \in \mathcal{X}}{\operatorname{argmin}} f(x)$$

- $\mathcal{X}$ is a bounded domain
- f is explicitly unknown
- Evaluations of f may be noisy

Black-box Optimisation

$$x^{(*)} = \underset{x \in \mathcal{X}}{\operatorname{argmin}} f(x)$$

- $\mathcal{X}$ is a bounded domain
- f is explicitly unknown
- Evaluations of f may be noisy
- Evaluations of f is expensive



Random Search [Brochu et al., 2010]

- Random Search

$$f(x^{(-)}) \leq f(x^{(*)}) - \epsilon$$

Random Search [Brochu et al., 2010]

- Random Search

$$f(x^{(-)}) \leq f(x^{(*)}) - \epsilon$$

- Lipschitz Continuity

$$\|f(x_1) - f(x_2)\| \leq C\|x_1 - x_2\|$$

Random Search [Brochu et al., 2010]

- Random Search

$$f(x^{(-)}) \leq f(x^{(*)}) - \epsilon$$

- Lipschitz Continuity

$$\|f(x_1) - f(x_2)\| \leq C\|x_1 - x_2\|$$

- Requires $\left(\frac{C}{2\epsilon}\right)^d$ evaluations on a d -dimensional hypercube

Random Search [Brochu et al., 2010]

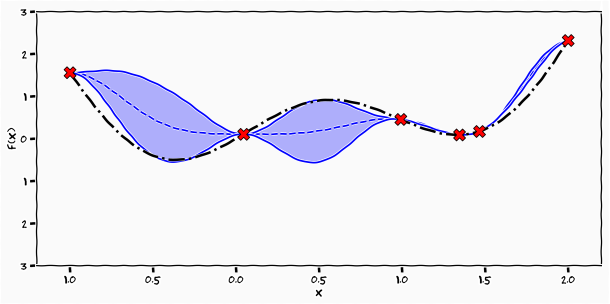
- Random Search

$$f(x^{(-)}) \leq f(x^{(*)}) - \epsilon$$

- Lipschitz Continuity

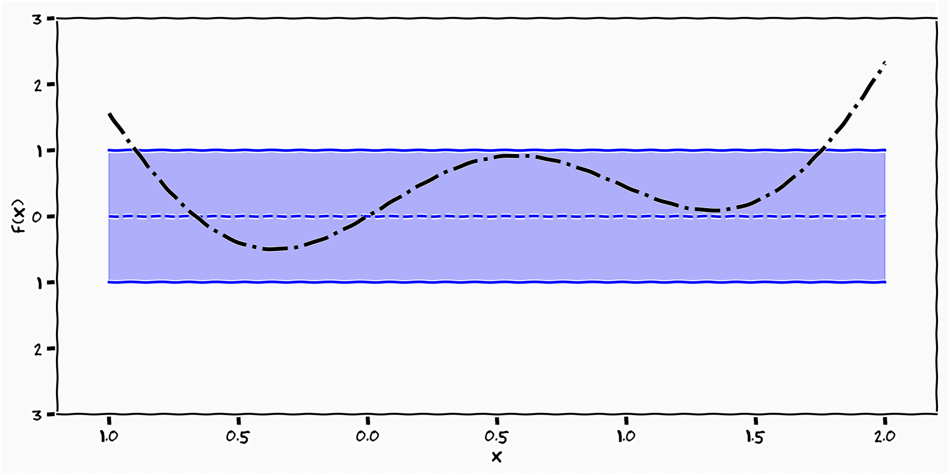
$$\|f(x_1) - f(x_2)\| \leq C\|x_1 - x_2\|$$

- Requires $\left(\frac{C}{2\epsilon}\right)^d$ evaluations on a d -dimensional hypercube
- Surrogate model $p(f)$

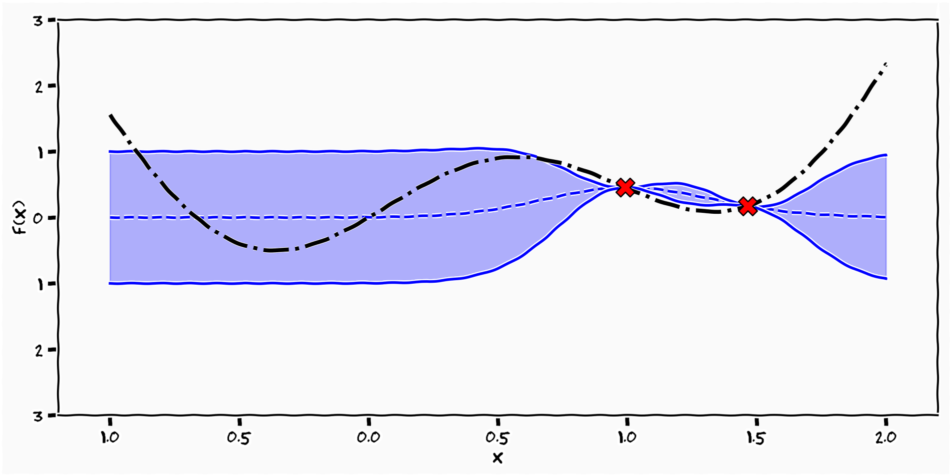


- allows for principled priors and narrow priors
- provides belief over the whole domain

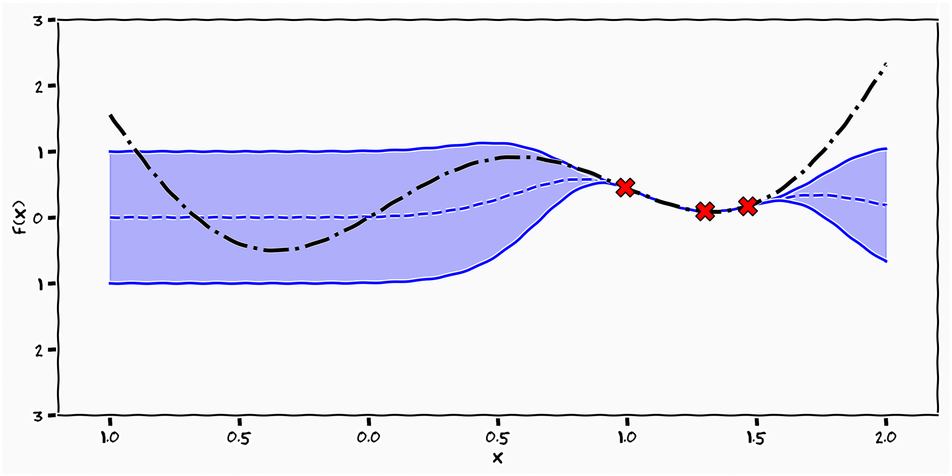
Posterior Search: Random



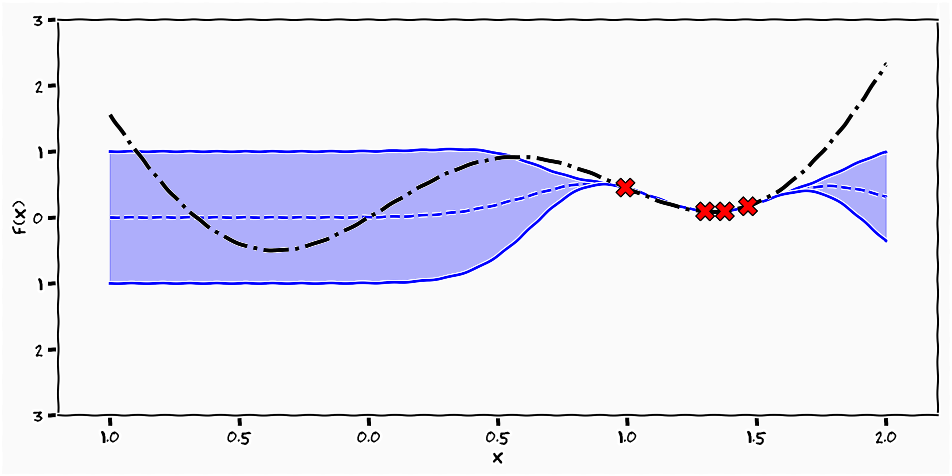
Posterior Search: Random



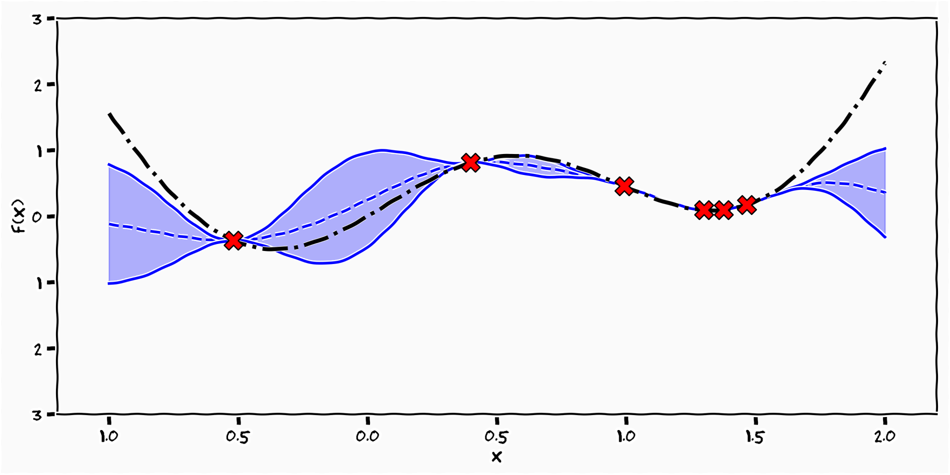
Posterior Search: Random

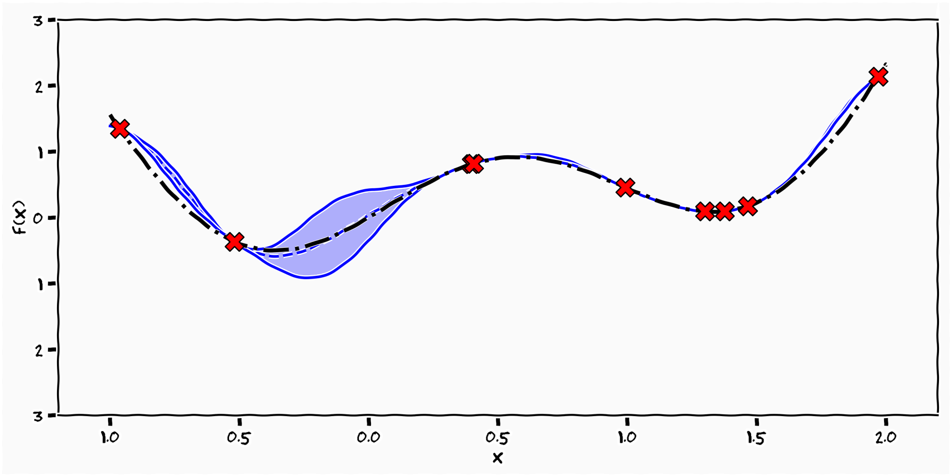


Posterior Search: Random

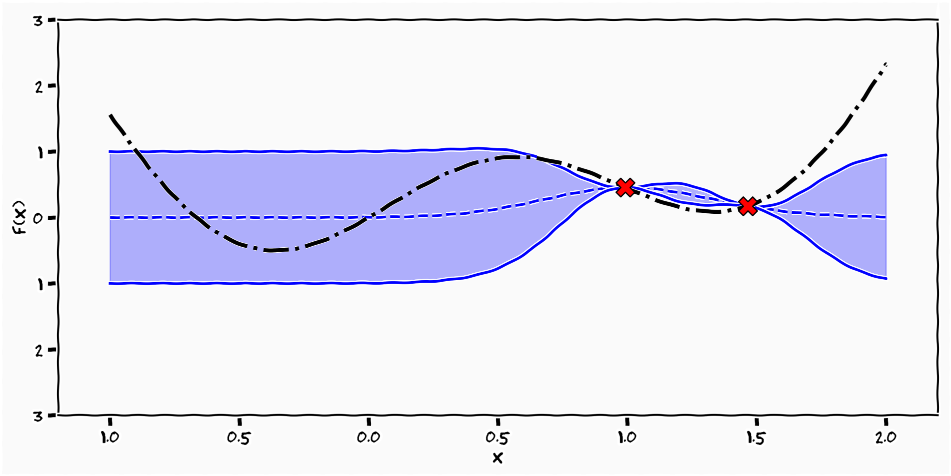


Posterior Search: Random

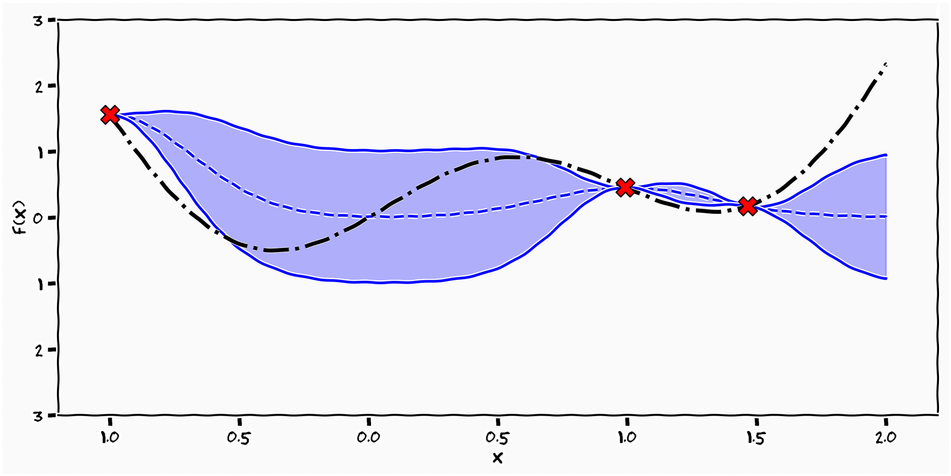



3
2
1
0
1
2
3
f(x)
1.0
0.5
0.0
0.5
1.0
1.5
2.0
x

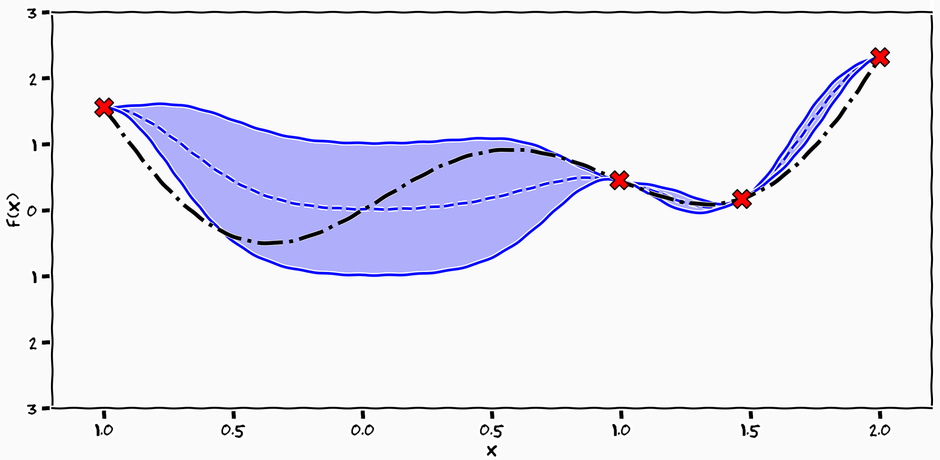
Posterior Search: Min



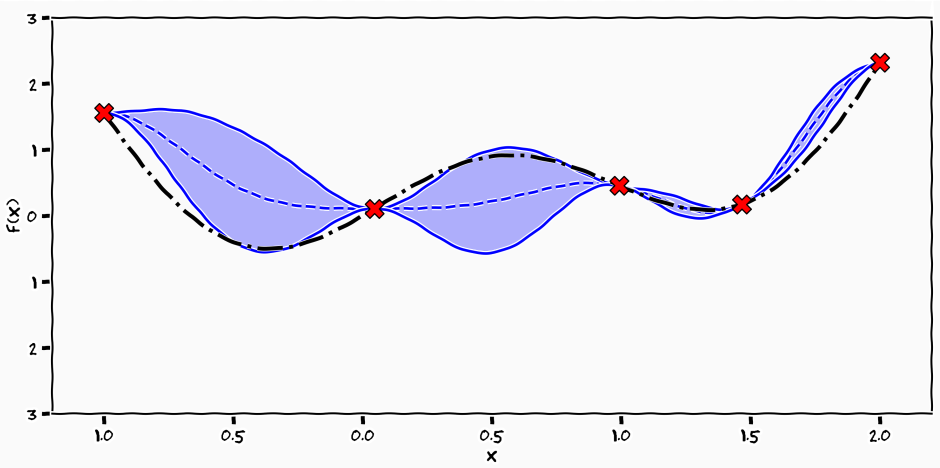
Posterior Search: Min



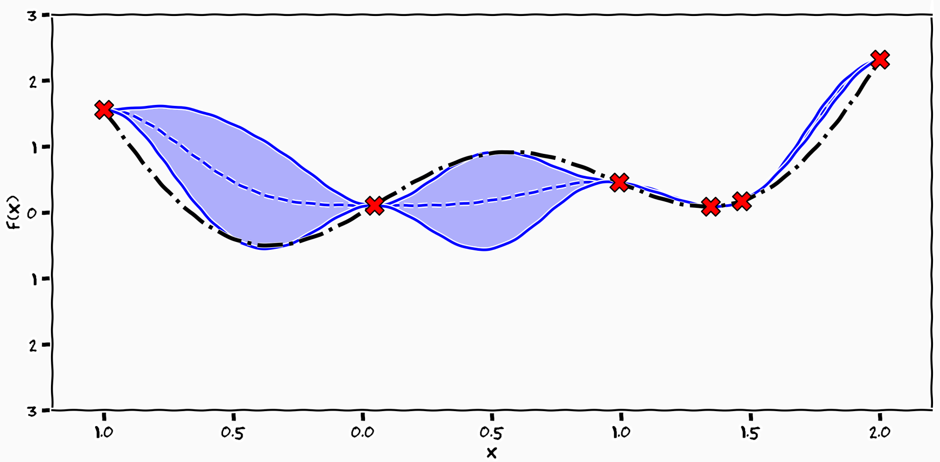
Posterior Search: Min



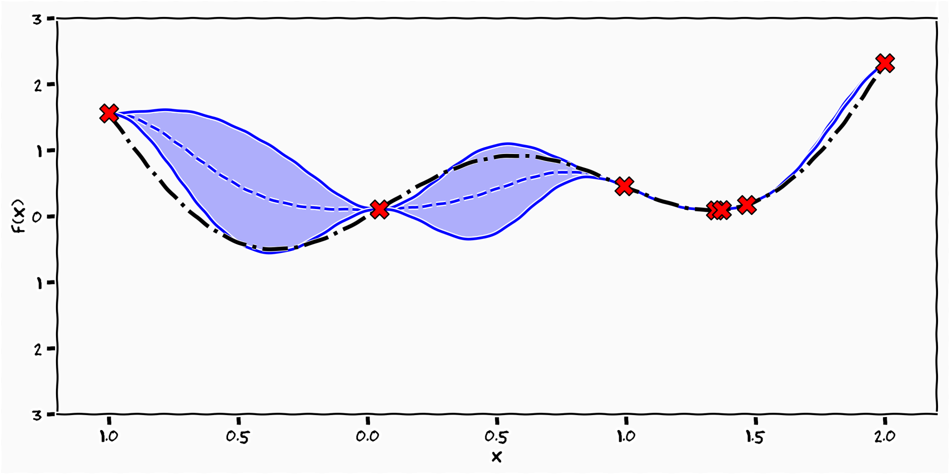
Posterior Search: Min



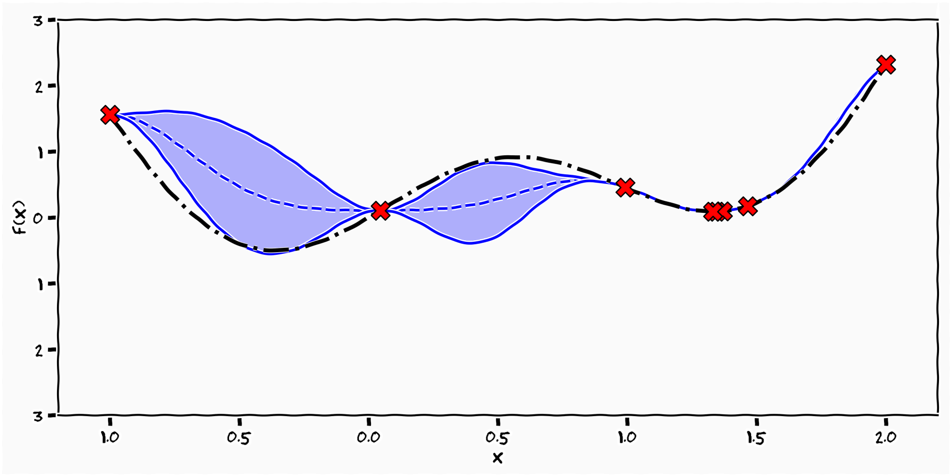
Posterior Search: Min



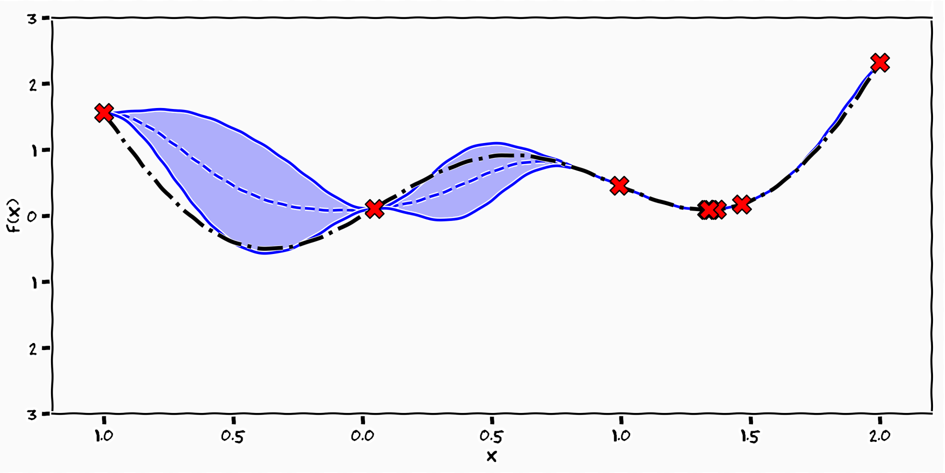
Posterior Search: Min



Posterior Search: Min



Posterior Search: Min




STRONGARM
CAMERONS
TIMOTHY TAYLOR'S
LANDLORD
FURNACE
NICK STAFFORD'S
HAMBLETON ALES
West Coast
Mosaic IPA
4.0% ABV

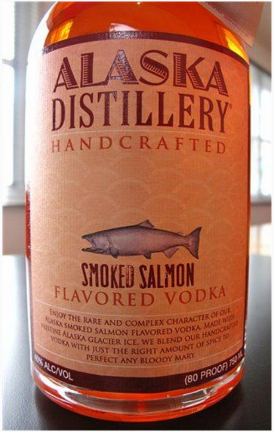

Coca-Cola

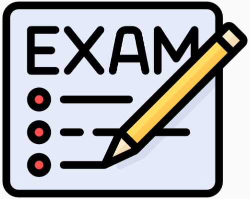

Coca-Cola


WYE VALLEY
BITTER
Superior Quality
ALC 3.7% VOL
WYE VALLEY
BREWERY
WYE VALLEY
BREWERY


Coca-Cola


WYE VALLEY
BITTER
Superior Quality
ALC 3.7% VOL
WYE VALLEY
BREWERY
WYE VALLEY
BREWERY

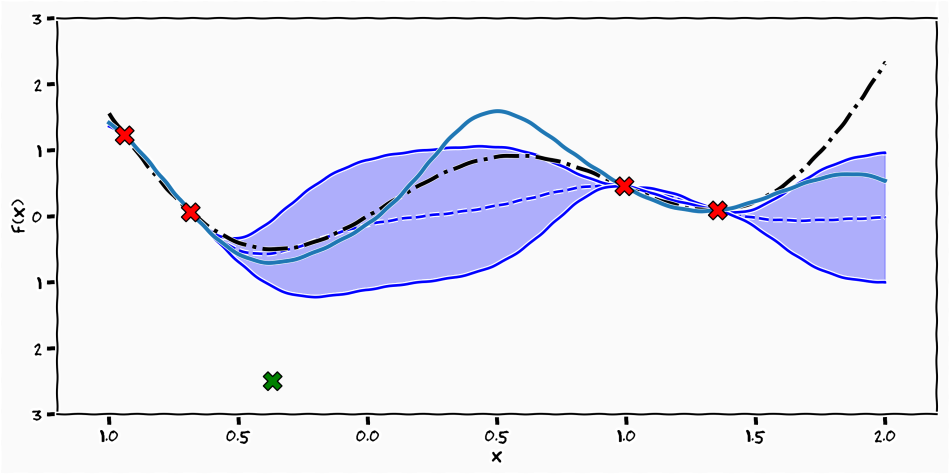

ALASKA
DISTILLERY
HANDCRAFTED
SMOKED SALMON
FLAVORED VODKA



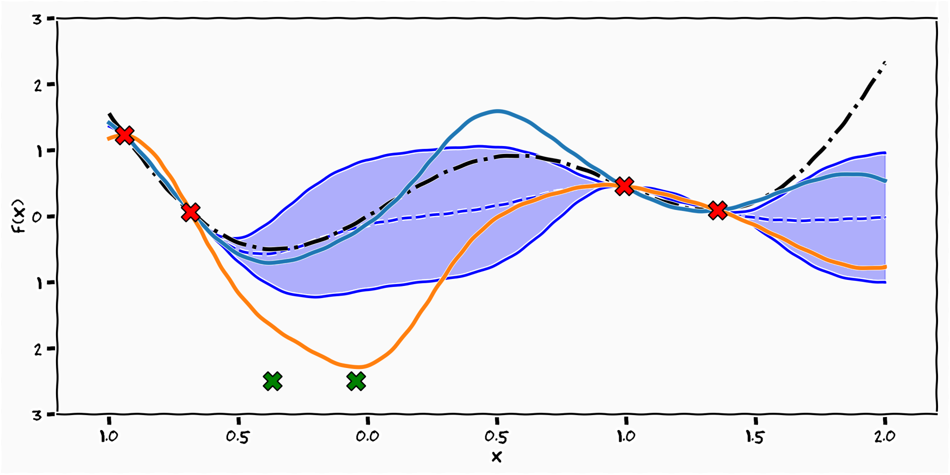
Exploitation use the knowledge that we currently have

Exploration try to gain new knowledge by trying new things

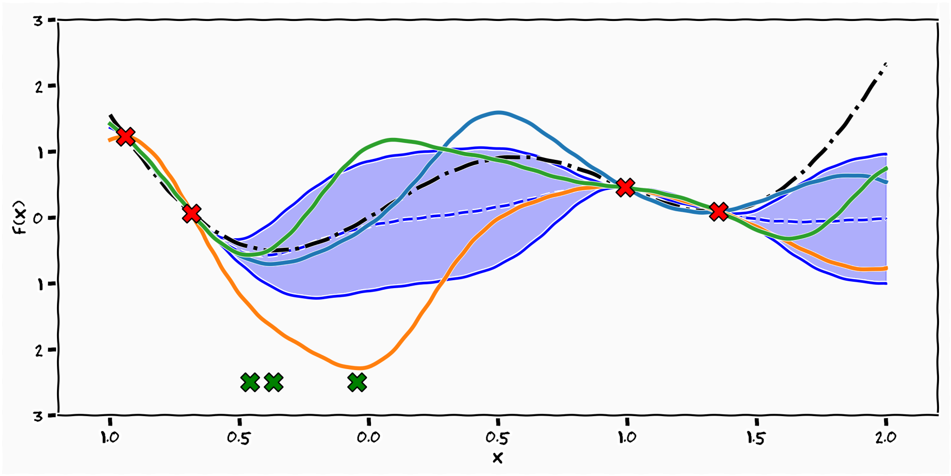
Surrogate Uncertainty



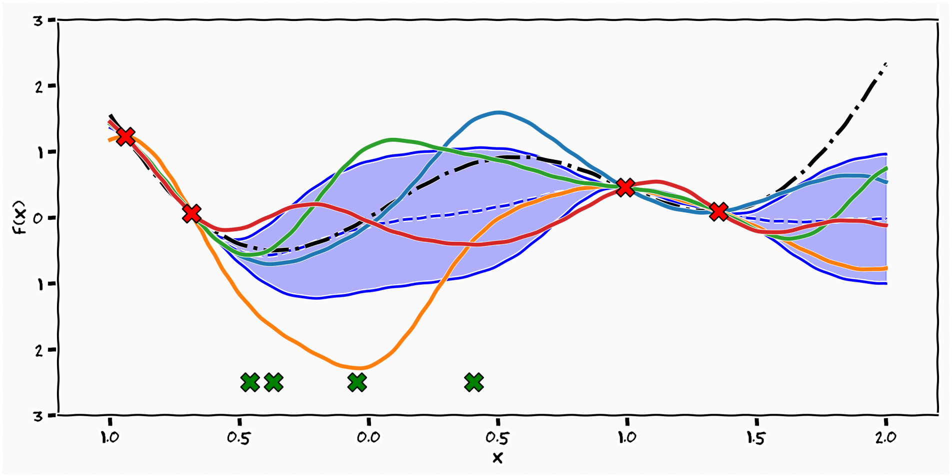
Surrogate Uncertainty



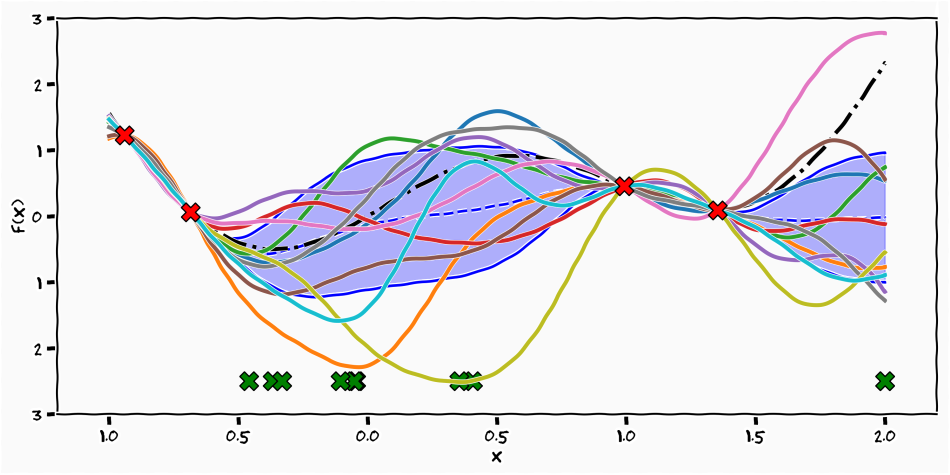
Surrogate Uncertainty

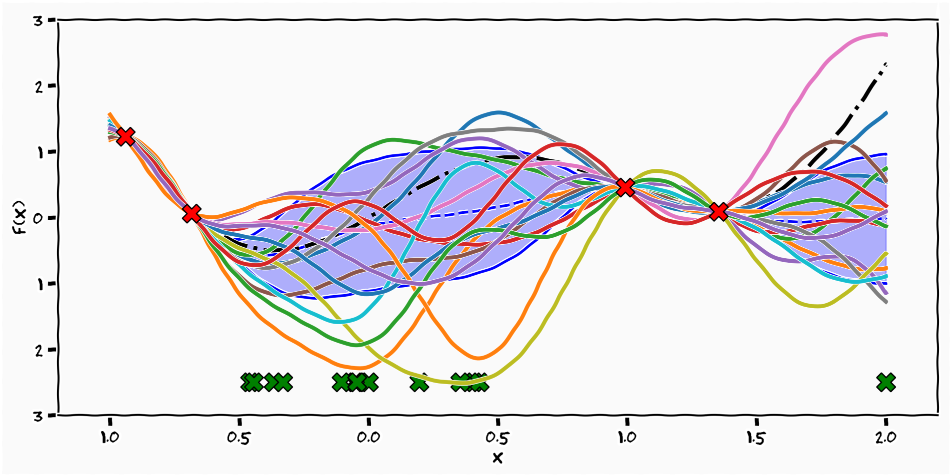


Surrogate Uncertainty



Surrogate Uncertainty





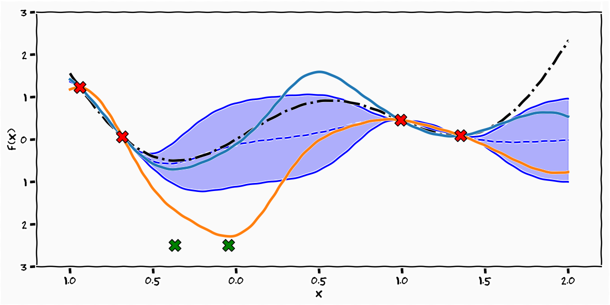
3
2
1
0
1
2
3
f(x)
1.0
0.5
0.0
0.5
1.0
1.5
2.0
x

Acquisition Function

$$x_{n+1} = \underset{x \in \mathcal{X}}{\operatorname{argmax}}\, \alpha(x; \{x_i, y_i\}_{i=1}^{n}, \mathcal{M}_n)$$

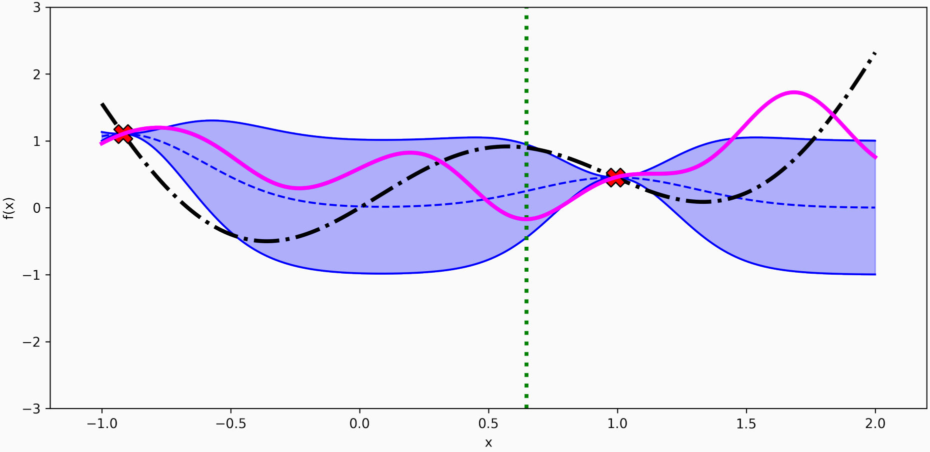
- Formulate a sequential decision problem
- This will work well if $\alpha(x)$
  - is cheap to compute
  - balances *exploration* and *exploitation*

Thompson Sampling [Thompson, 1933]

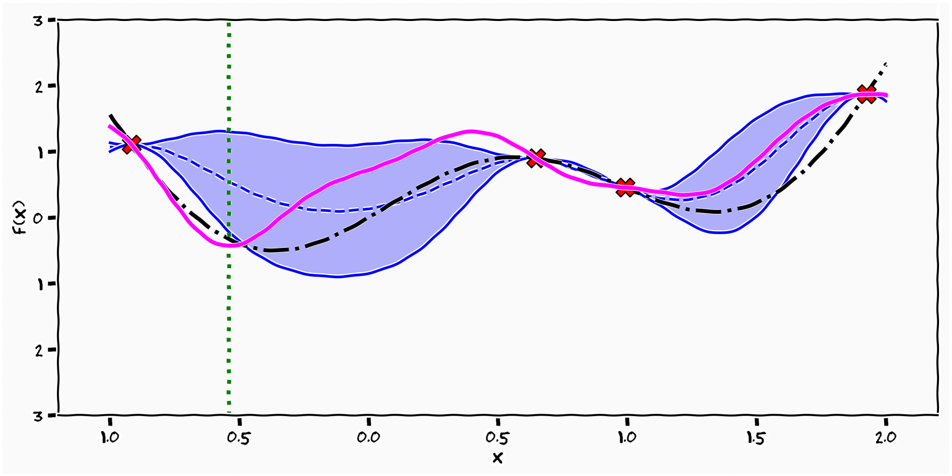


$$-\alpha(x; \{x_i, y_i\}_{i=1}^{n}, \mathcal{M}_n) \sim p(f \mid \{x_i, y_i\}_{i=1}^{n})$$

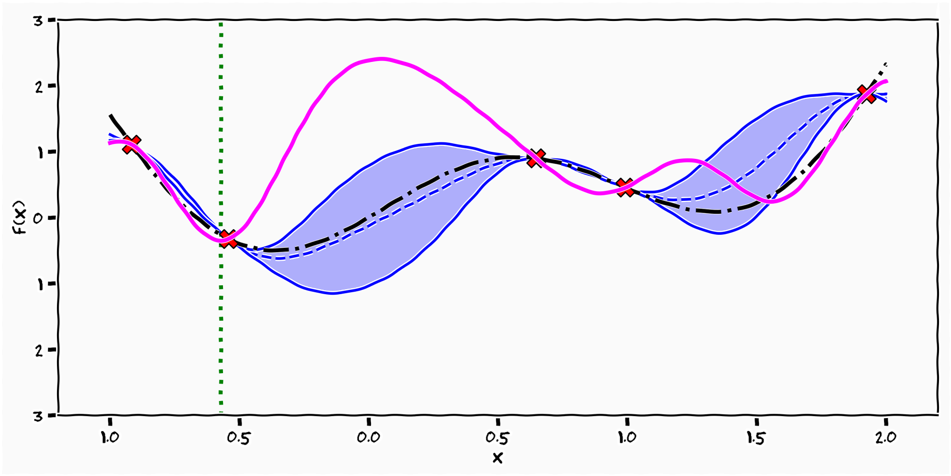
Thompson Sampling



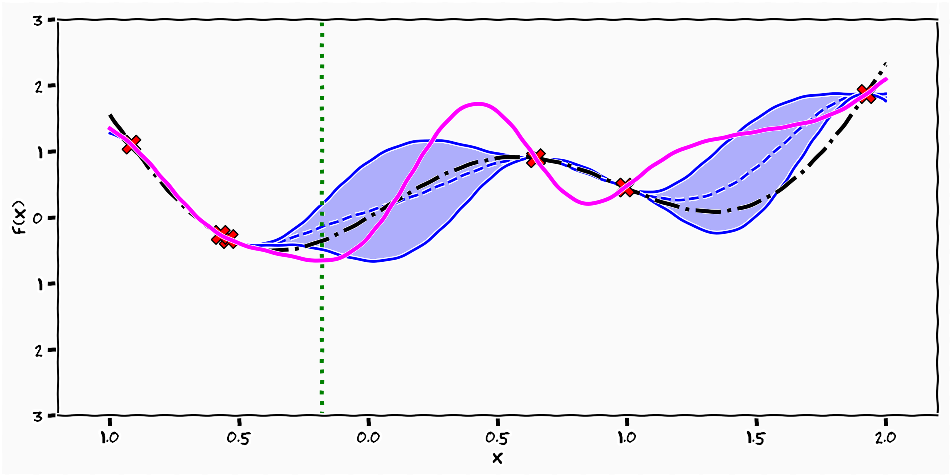
Thompson Sampling



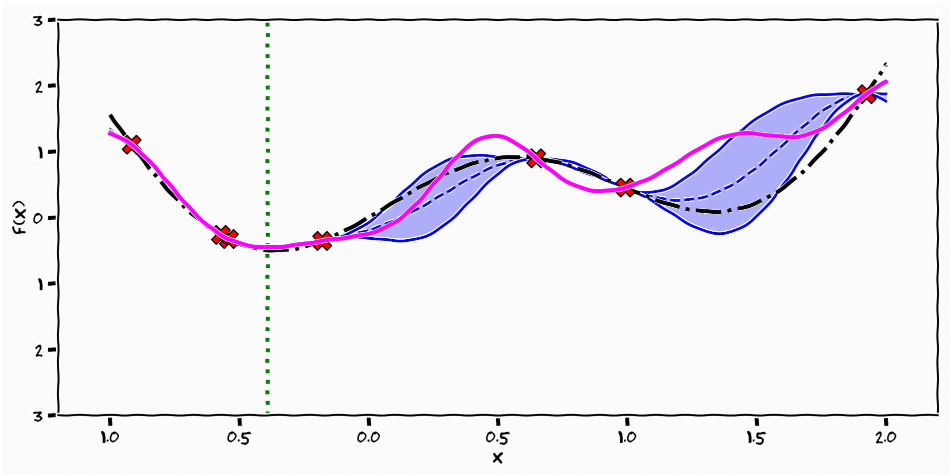
Thompson Sampling



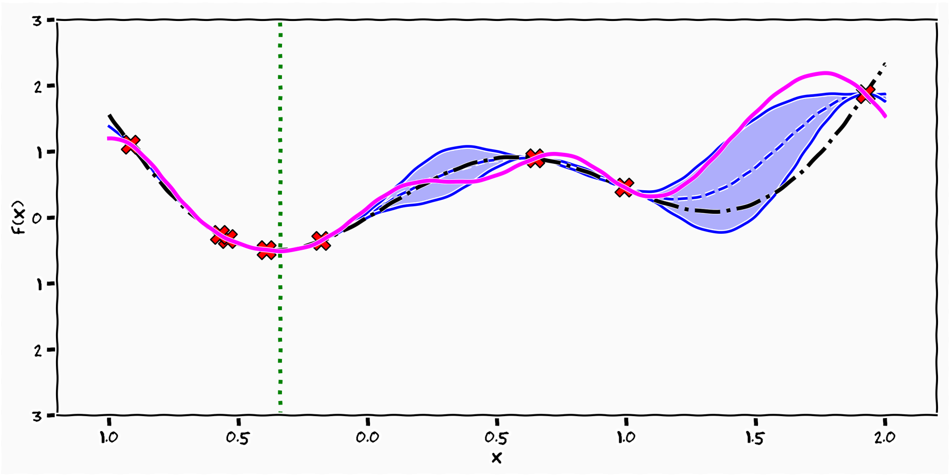
Thompson Sampling



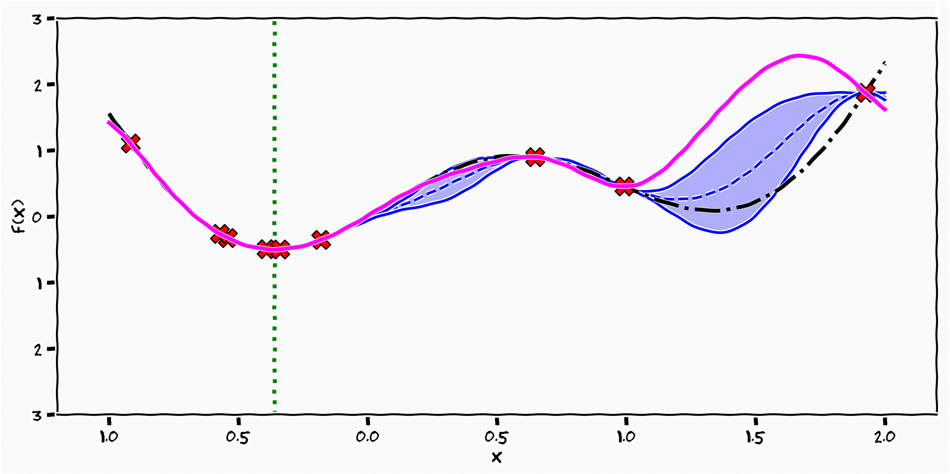
Thompson Sampling



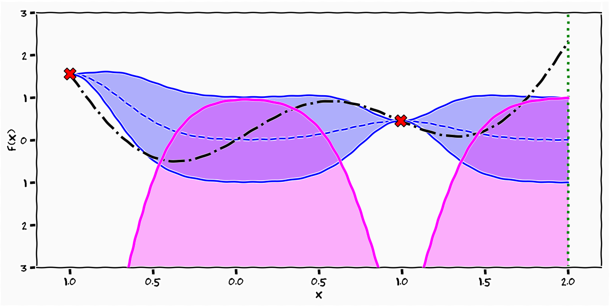
Thompson Sampling



Thompson Sampling



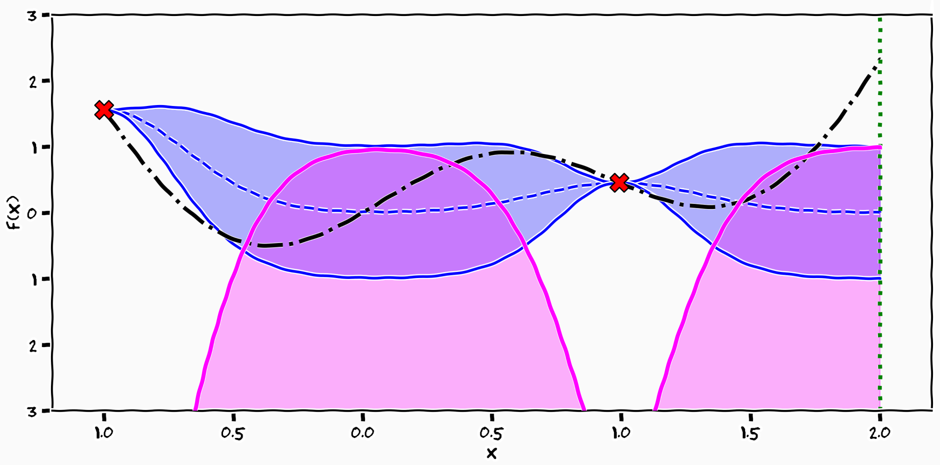
Upper Confidence Bound [Cox et al., 1997]



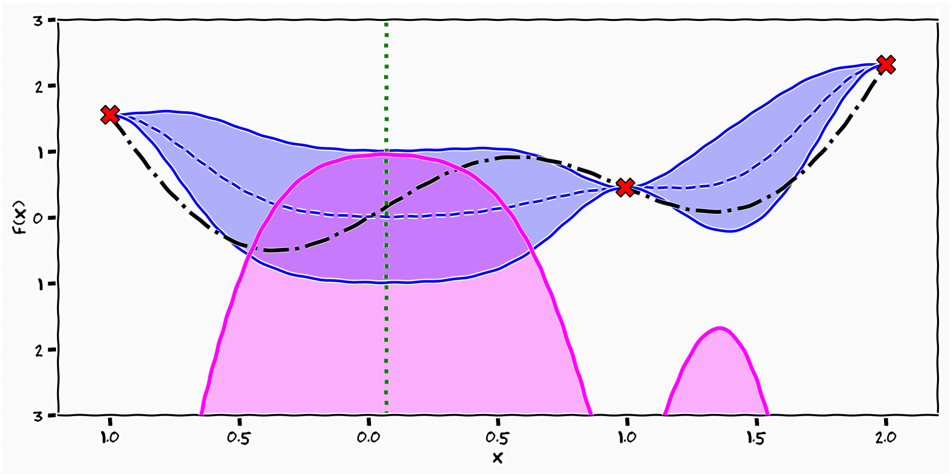
- Acquisition Function

$$\alpha(x; \{x_i, y_i\}_{i=1}^n, \mathcal{M}_n) = -\mu(x; \{x_i, y_i\}_{i=1}^n) + \beta\sigma(x; \{x_i, y_i\}_{i=1}^n)$$

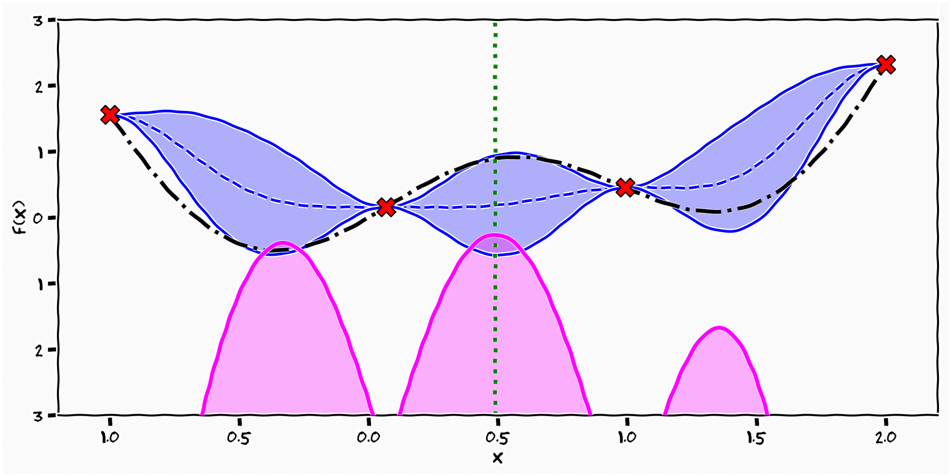
Upper Confidence Bound



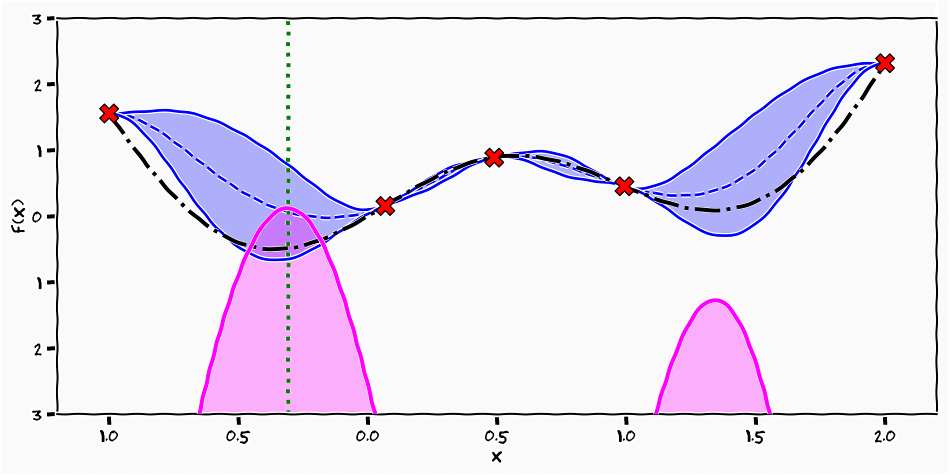
Upper Confidence Bound



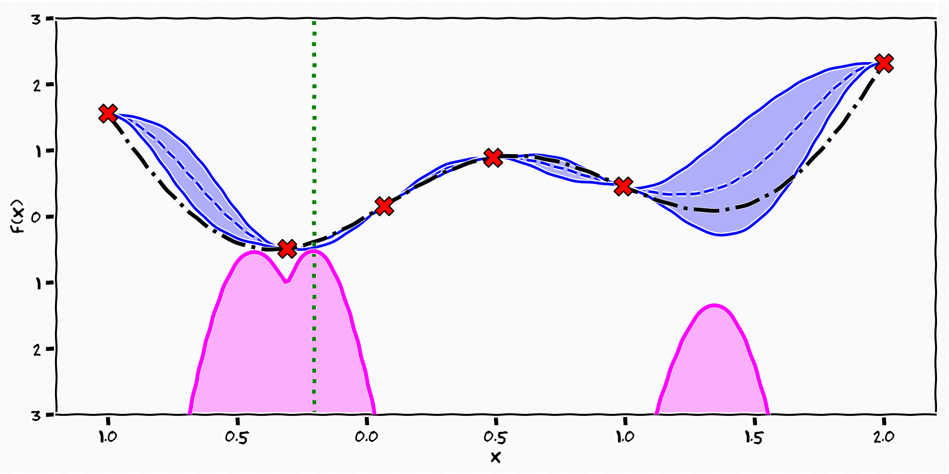
Upper Confidence Bound



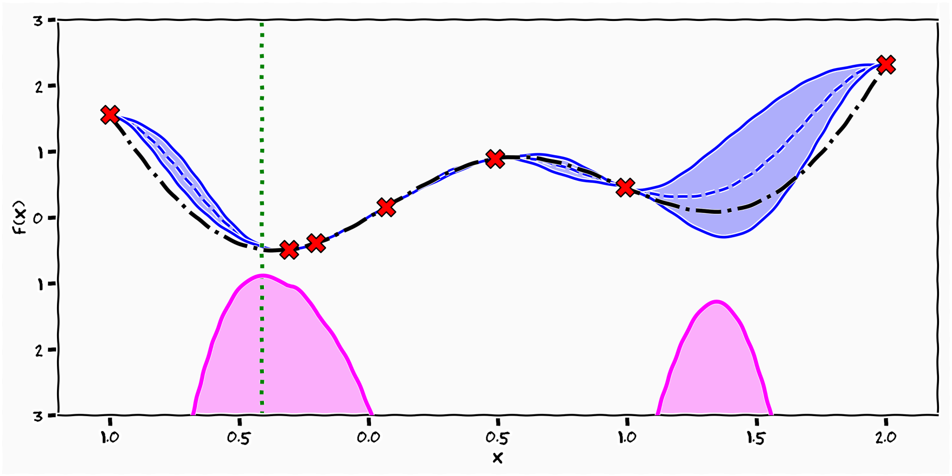
Upper Confidence Bound



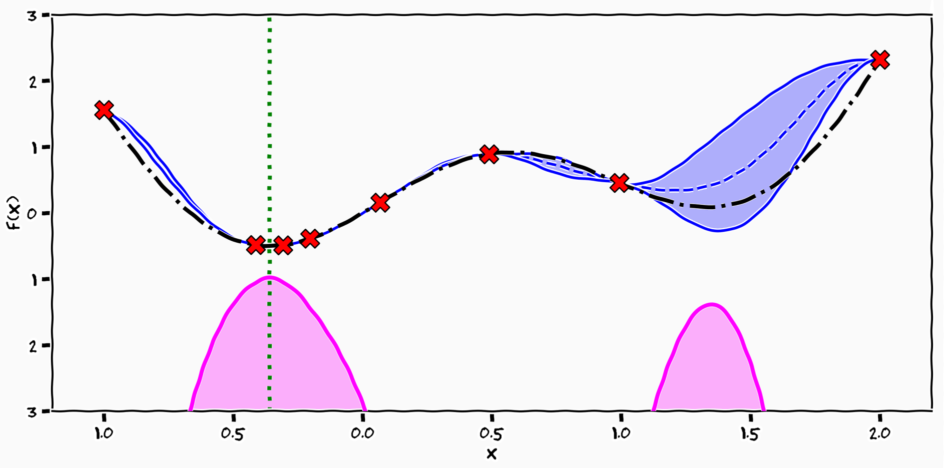
Upper Confidence Bound



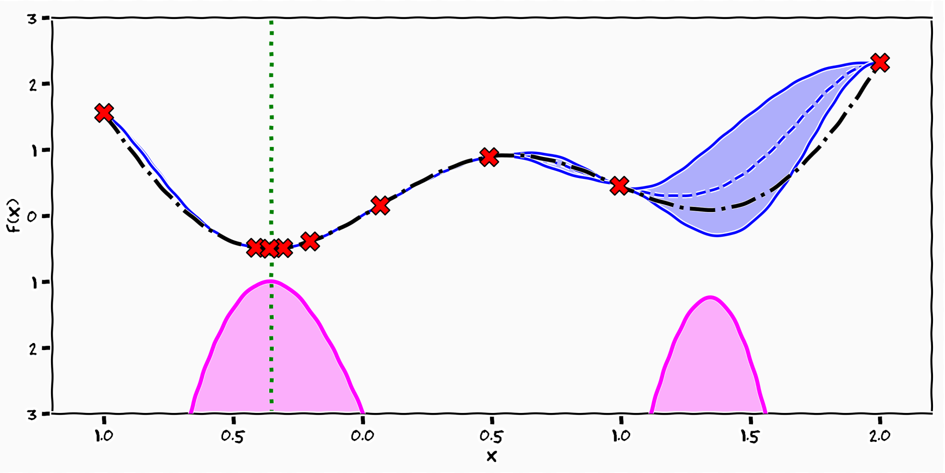
Upper Confidence Bound



Upper Confidence Bound



Upper Confidence Bound



Utility

- We can come up with lots of heuristics of how to define acquisition functions

Utility

- We can come up with lots of heuristics of how to define acquisition functions
- Define a function that defines the utility of observing each location

$$u(x, f(x^{(*)}), \mathcal{M}_n)$$

Utility

- We can come up with lots of heuristics of how to define acquisition functions
- Define a function that defines the utility of observing each location

$$u(x, f(x^{(*)}), \mathcal{M}_n)$$

- Define the acquisition function as the expected utility

$$\begin{aligned}\alpha(x; \{x_i, y_i\}_{i=1}^{n}, \mathcal{M}_n) &= \mathbb{E}_{p(f)}[u(x)] \\ &= \int u(x, f(x^{(*)}), \mathcal{M}_n) p(f \mid \{x_i, y_i\}_{i=1}^{n}) \mathrm{d}f\end{aligned}$$

Probability of Improvement [Kushner, 1963]

- Utility Function

$$u(x) = \begin{cases} 0 & f(x) > f(x^{(*)}) \\ 1 & f(x) \leq f(x^{(*)}) \end{cases}$$

Probability of Improvement [Kushner, 1963]

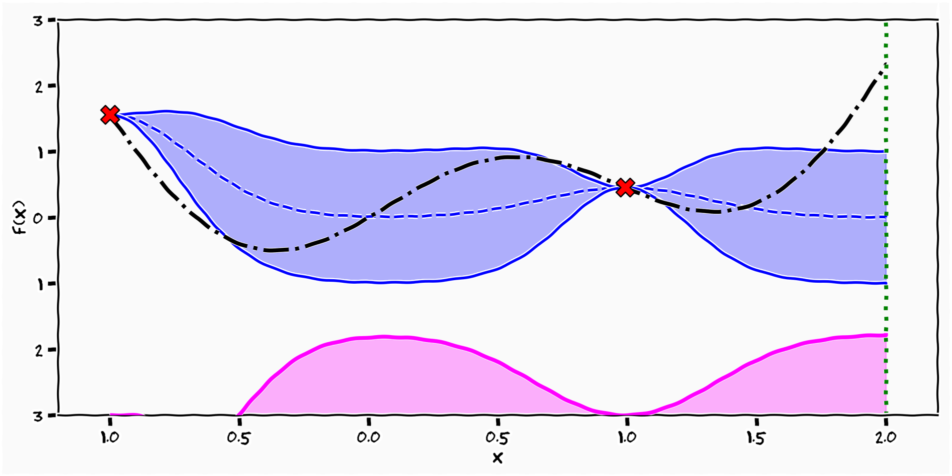
- Utility Function

$$u(x) = \begin{cases} 0 & f(x) > f(x^{(*)}) \\ 1 & f(x) \leq f(x^{(*)}) \end{cases}$$

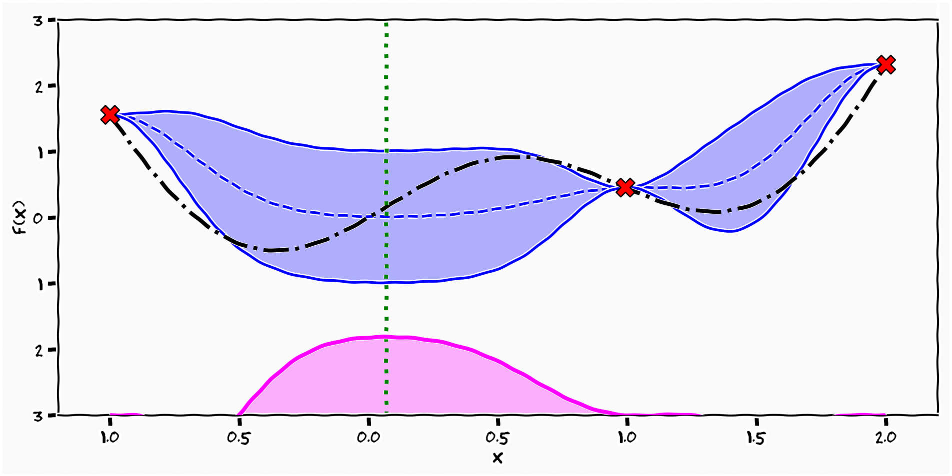
- Acquisition Function

$$\begin{aligned} \alpha(x; \{x_i, y_i\}_{i=1}^{n}, f(x^{(*)}), \mathcal{M}_n) &= \mathbb{E}[u(x)] = p(f(x) \leq f(x^{(*)})) \\ &= \int_{-\infty}^{f(x^{(*)})} \mathcal{N}\left(f \mid \mu(x), K(x,x)\right) \mathrm{d}f \\ &= \Phi\left(f(x^{(*)}) \mid \mu(x), K(x,x)\right) \end{aligned}$$

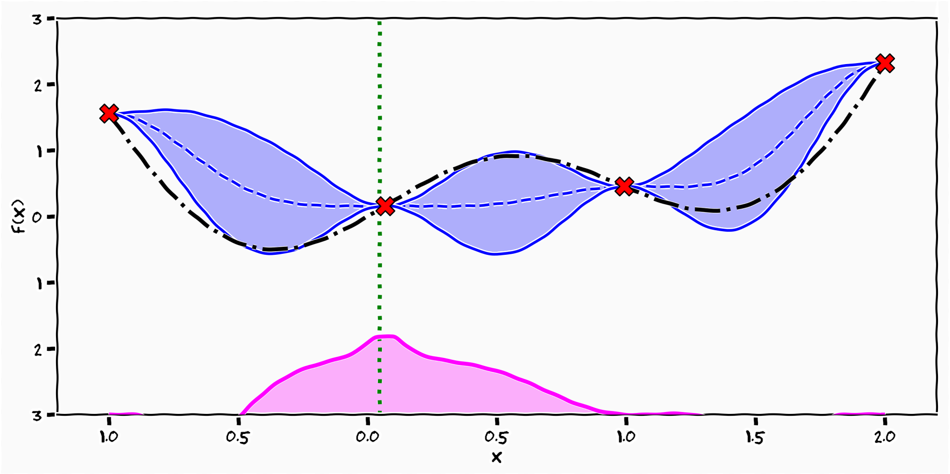
Probability of Improvement



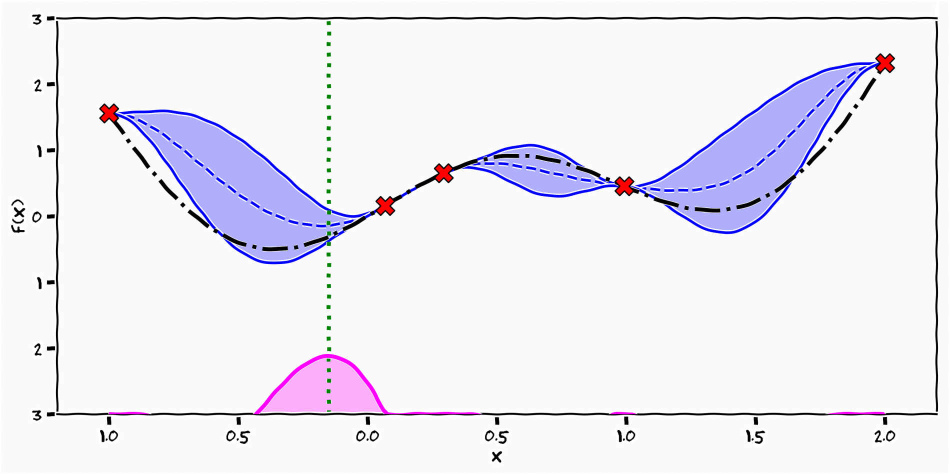
Probability of Improvement



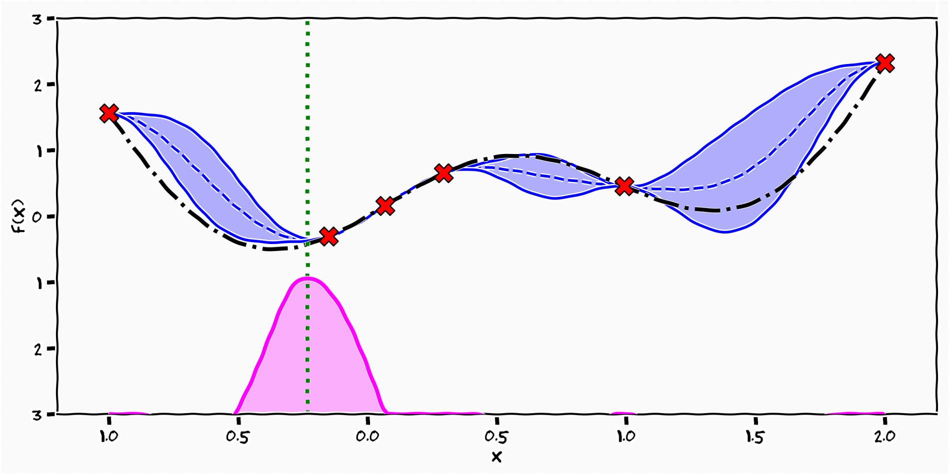
Probability of Improvement



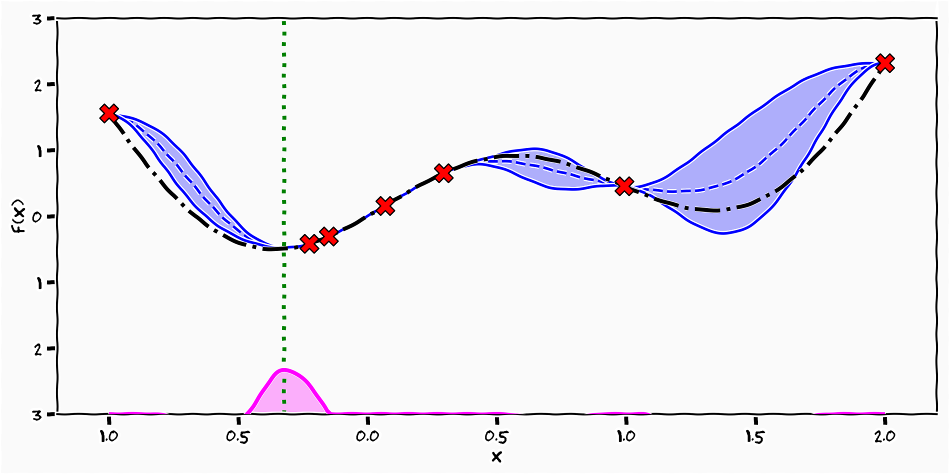
Probability of Improvement



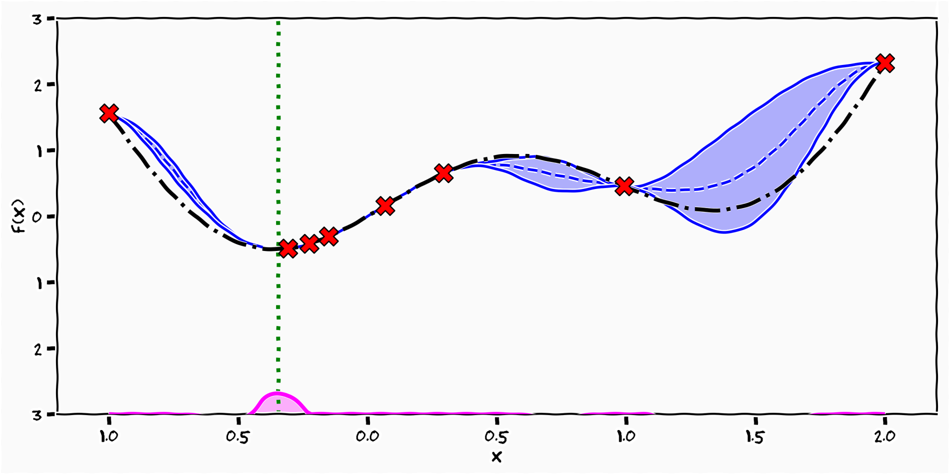
Probability of Improvement



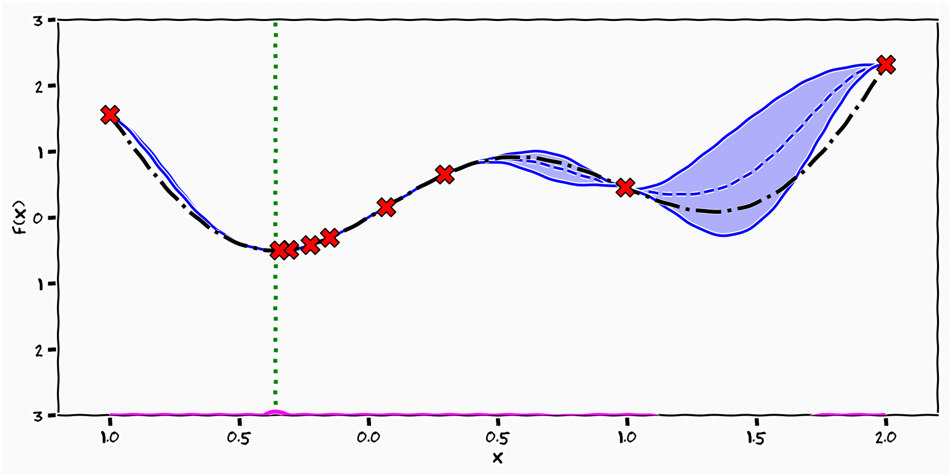
Probability of Improvement



Probability of Improvement



Probability of Improvement



Expected Improvement Mockus et al., 1978

- Utility Function

$$u(x) = \max(0, f(x^{(*)}) - f(x))$$

Expected Improvement Mockus et al., 1978

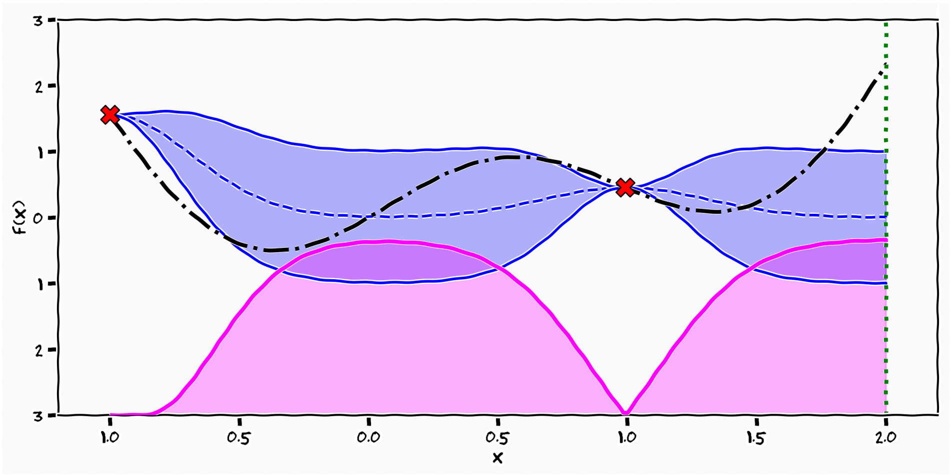
- Utility Function

$$u(x) = \max(0, f(x^{(*)}) - f(x))$$

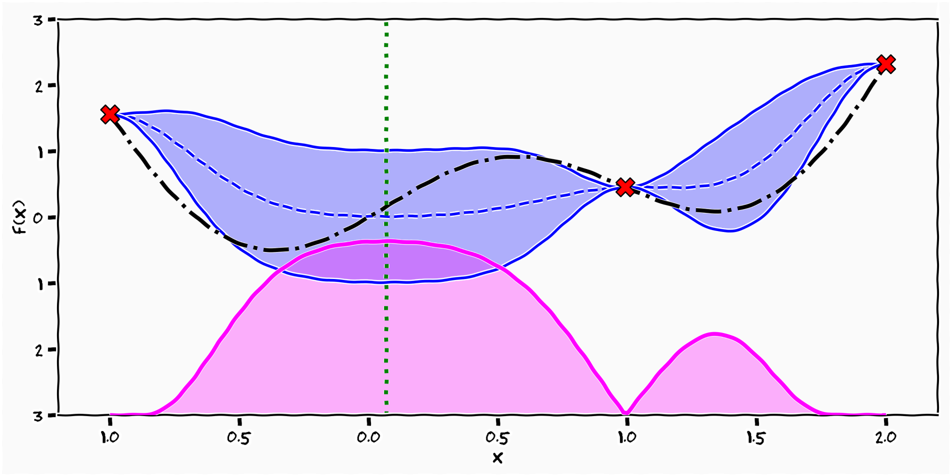
- Acquisition Function

$$\begin{aligned}
\alpha(x; \{x_i, y_i\}_{i=1}^{n}, f(x^{(*)}), \mathcal{M}_n) &= \mathbb{E}[u(x)] \\
&= \int_{-\infty}^{f(x^{(*)})} (f(x^{(*)}) - f)\mathcal{N}\left(f \mid \mu(x), K(x,x)\right) \mathrm{d}f \\
&= (f(x^{(*)}) - \mu(x))\Phi\left(f(x^{(*)}) \mid \mu(x), K(x,x)\right) \\
&+ K(x,x)\mathcal{N}\left(f(x^{(*)}) \mid \mu(x), K(x,x)\right)
\end{aligned}$$

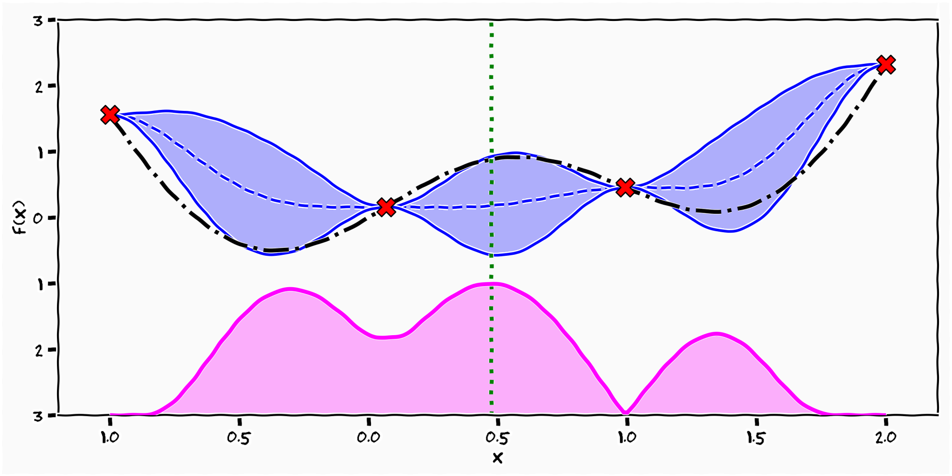
Expected Improvement



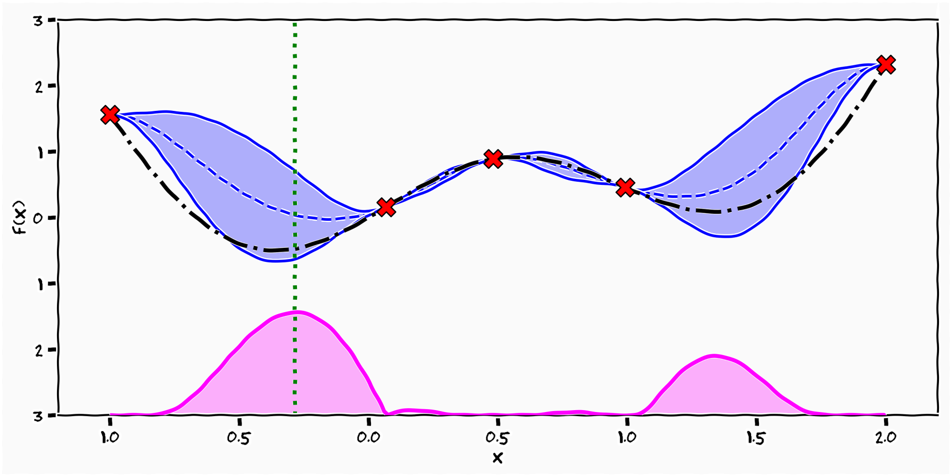
Expected Improvement



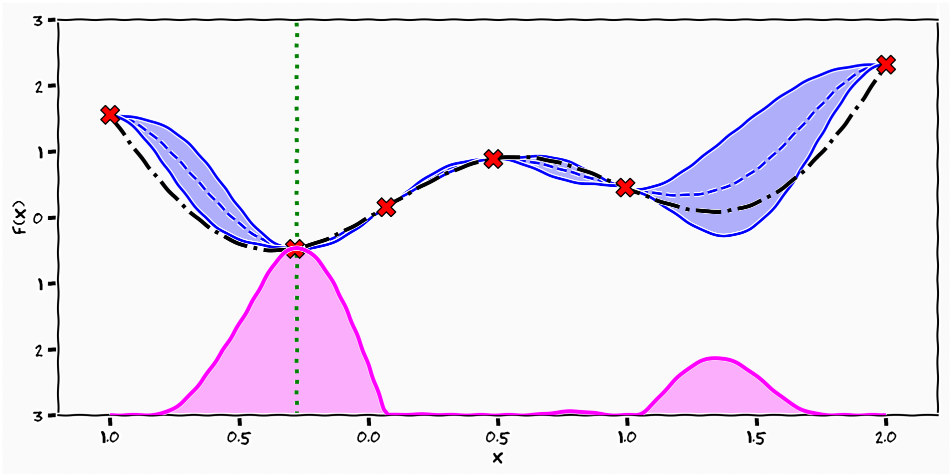
Expected Improvement

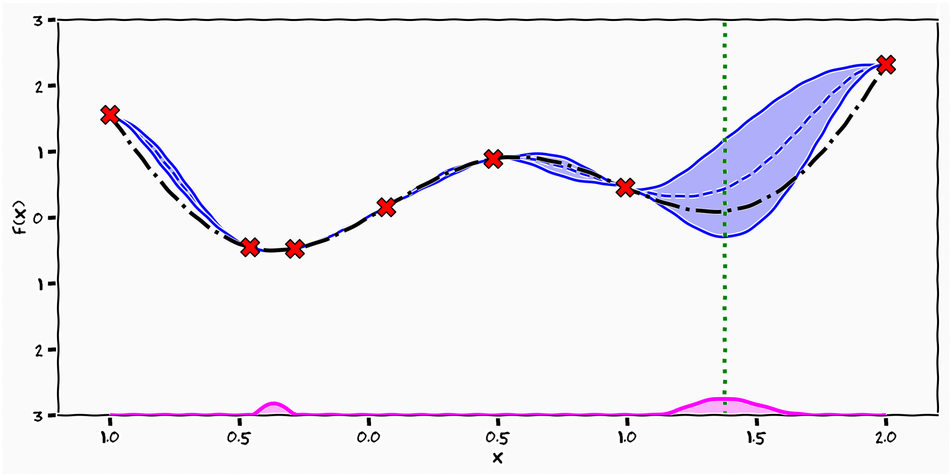


Expected Improvement

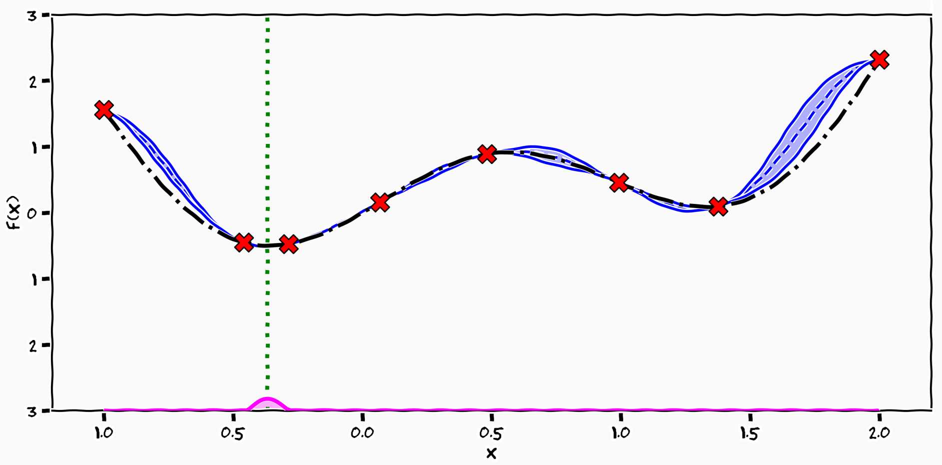


Expected Improvement

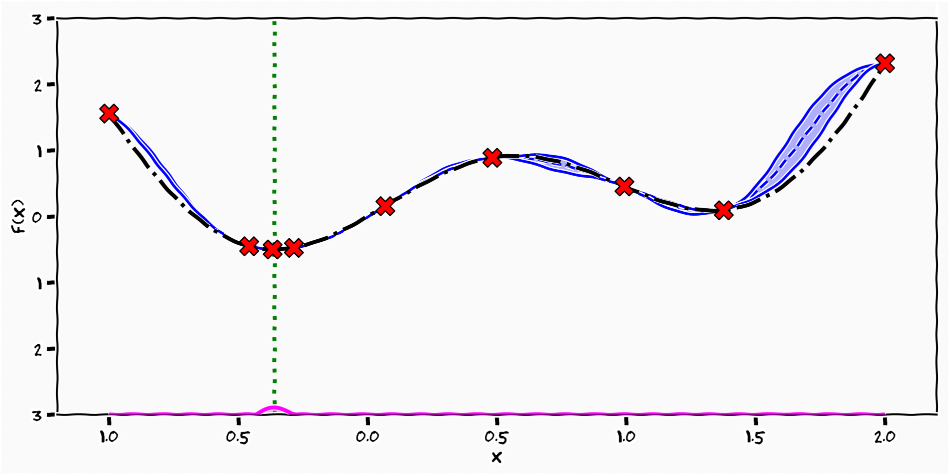



3
2
1
0
1
2
3
f(x)
1.0
0.5
0.0
0.5
1.0
1.5
2.0
x

Expected Improvement



Expected Improvement



Task 1 encode your knowledge about the function in the GP prior

[1] till they open the door to the exam.

Task 1 encode your knowledge about the function in the GP prior

Task 2 randomly sample some data

[1]till they open the door to the exam.

Task 1 encode your knowledge about the function in the GP prior

Task 2 randomly sample some data

Task 3 specify your acquition function

[1] till they open the door to the exam.

BO in Practice

Task 1 encode your knowledge about the function in the GP prior

Task 2 randomly sample some data

Task 3 specify your acquition function

Task 4 evaluate and maximise the acquisition function

[1] till they open the door to the exam.

BO in Practice

Task 1 encode your knowledge about the function in the GP prior

Task 2 randomly sample some data

Task 3 specify your acquition function

Task 4 evaluate and maximise the acquisition function

Task 5 add new data to model and re-estimate hyperparameters

[1]till they open the door to the exam.

BO in Practice

Task 1 encode your knowledge about the function in the GP prior

Task 2 randomly sample some data

Task 3 specify your acquition function

Task 4 evaluate and maximise the acquisition function

Task 5 add new data to model and re-estimate hyperparameters

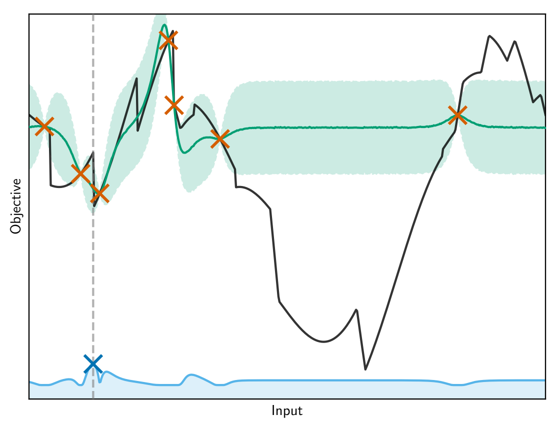
Loop 4-5 till budget is gone[1]

[1] till they open the door to the exam.

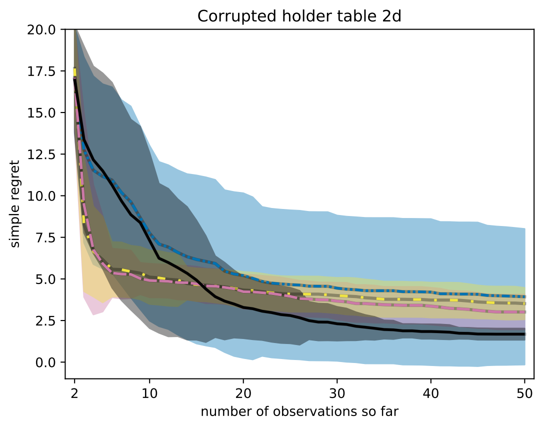
Bayesian Optimisation in Practice

Academia vs. Industry

Challenges: Initial Experiments [Bodin et al., 2020]



Challenges: Initial Experiments [Bodin et al., 2020]



Hyper-parameters

- Fixed data

$$\hat{\theta} = \operatorname{argmax}_{\theta} p(\mathbf{y} \mid x\theta)$$

$$p(f_* \mid \mathbf{y}, \mathbf{x}, x_*, \theta = \hat{\theta})$$

Hyper-parameters

- Fixed data

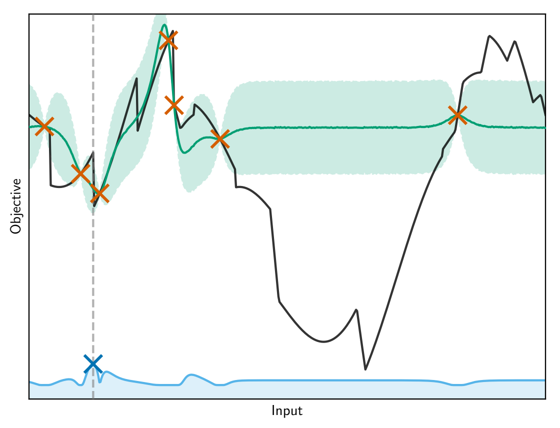
$$\hat{\theta} = \mathrm{argmax}_{\theta}\, p(\mathbf{y} \mid x\theta)$$

$$p(f_* \mid \mathbf{y}, \mathbf{x}, x_*, \theta = \hat{\theta})$$

- Active setting

$$p(f_* \mid \mathbf{y}, \mathbf{x}, x_*) = \int p(f_* \mid \mathbf{y}, \mathbf{x}, x_*, \theta) p(\theta)$$

Challenges: Function is just a proxy



Challenges: High Dimensional Structures

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\frac{\mathbf{x}_i^T \mathbf{x}_j}{\ell^2}$$

Challenges: Greedy Acquisition

Summary

Summary

- GPs are quite useful surrogates!

Summary

- GPs are quite useful surrogates!
- Degrees of beliefs are really useful

Summary

- GPs are quite useful surrogates!
- Degrees of beliefs are really useful
- The uncertainty allows us to design rich strategies for how to aquire data

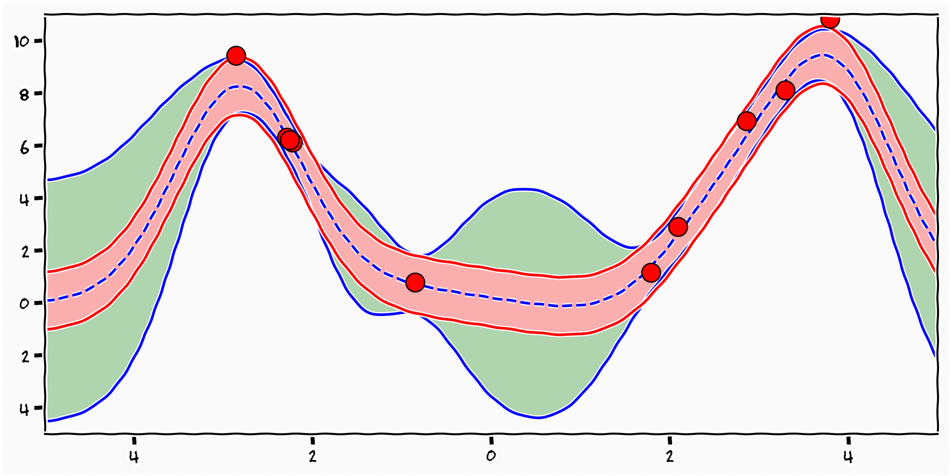
Summary

- GPs are quite useful surrogates!
- Degrees of beliefs are really useful
- The uncertainty allows us to design rich strategies for how to aquire data
- The factorisation of uncertainty allows us to describe search strategies in simple acquisition functions

Individual Submission

- I have reluctantly made another `Jupyter Notebook`
- It will be online by the end of the day
- Similar to the material in the `PDF`
- Deadline Friday 7rd of November at 16:00

Uncertainty Quantification/Factorisation

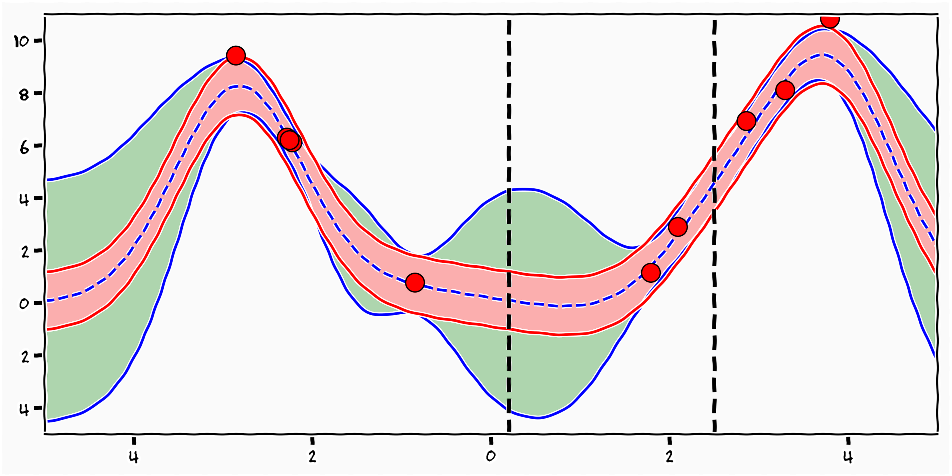


Uncertainty Quantification

Aleatoric/Stochastic "Randomness" inherent in system, or noise in our measurement of system

Epistemic Uncertainty related to our ignorance of a the underlying system

Uncertainty for Decision Making



eof

References

Bodin, Erik et al. (2020). "**Modulating Surrogates for Bayesian Optimization..**" In: *Proceedings of the 37th International Conference on Machine Learning, ICML 2019, 12-18 July 2020, Virtual*.

Brochu, Eric, Vlad M. Cora, and Nando de Freitas (2010). "**A Tutorial on Bayesian Optimization of Expensive Cost Functions, With Application To Active User Modeling and Hierarchical Reinforcement Learning.**" In: *CoRR*.

- Cox, Dennis and Susan John (Mar. 1997). "**SDO: A Statistical Method for Global Optimization.**" In: *Multidisciplinary Design Optimization: State of the Art*. Ed. by M. N. Alexandrov and M. Y. Hussaini, pp. 315–329.
- Kushner, Harold J. (1963). "**A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise.**" Undetermined. In: *Joint Automatic Control Conference* 1, pp. 69 –79.
- Mockus, J., Vytautas Tiesis, and Antanas Zilinskas (Sept. 1978). "**The application of Bayesian methods for seeking the extremum.**" In: vol. 2, pp. 117–129.

Thompson, William R. (1933). **"On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples."** In: *Biometrika* 25.3/4, pp. 285–294.