Bayesian Optimization

CSC 412/2506 Tutorial

Geoffrey Roeder
Mar 31, 2017

Slides from Kevin Swersky, Nando de Freitas

Course of tutorial

- What problem are we solving with BayesOpt?
- Gaussian Process review: notebook
- Acquisition functions
- EI Example

What problem are we solving?

Machine Learning Basics

- Given: input/output pairs $\{(x_n, y_n)\}_{n=1}^{N}$
- Goal: find a function that maps inputs to outputs

$$f(x) = \widehat{y}$$

- Want to predict correct outputs for *unseen* inputs

Example: Linear Regression

- Problem: fit a curve to one-dimensional data

- Model: linear weights with polynomial basis

$$f(x;a) = a_0 + \textstyle\sum_{i=1}^{p} a_i x_i^p$$

- Training algorithm: least squares

$$\hat{a} = \mathrm{argmin}_a - \tfrac{1}{2} \textstyle\sum_{n=1}^{N} (y_n - f(x_n;a))^2$$

- Validation: 5-fold cross-validation

Meta-Parameters

- The model parameters are the regression coefficients.
- We train these with least squares.
- Are there any other parameters?
    - Yes!
    - Polynomial degree
    - Choice of basis (Polynomial, Fourier, Wavelet, …)
    - Learning algorithm (Least squares, gradient descent, …)
    - Regularization strength
    - Regularizer (L1, L2, …)

Meta-Parameters

- Modeling decisions or free variables that cannot be trained using gradient (or other principled) methods.
- Can only evaluate a setting of the meta-parameters by training the model.
- This is very expensive, we want to do this as little as possible.

Typical Search Strategies

- Expert Intuition
- Grid Search
- Random Search
- Grad Student Search

Optimization Framework

- Meta-parameter search is an optimization problem!
  - There is some latent, potentially noisy function that maps meta-parameter settings to a score.
  - The input domain is bounded to some reasonable range.
  - Find the setting that minimizes the score.
- Each function evaluation is expensive, so we need to be clever about how often we query it.

Uncertainty

- In order to perform global optimization we need to characterize our *uncertainty*.
  - Explore places we are unsure about.
  - Exploit when we are sure we can improve.
- Two major sources of uncertainty:
  - Process noise - the observations are not perfectly accurate.
  - Model uncertainty - the response surface is one of many sensible possibilities.

Bayesian Optimization

- Mockus, 1978

    1. Incorporate a prior over the space of possible objective functions.

    2. Combine the prior and likelihood (model fit to data) to get a posterior over function values given observations.

    3. Select the next input to evaluate based on the posterior.

- According to what strategy?

Gaussian Processes

- Distribution over functions $f : \mathcal{X} \to \mathbb{R}$
- The observations at points $\mathbf{X} = \{x_n \in \mathcal{X}\}_{n=1}^{N}$ are jointly Gaussian
- Specified by a mean $m : \mathcal{X} \to \mathbb{R}$ and covariance $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$
- Predictive mean and covariance given observations $(\mathbf{X}, \mathbf{y}) = \{x_n, y_n\}_{n=1}^{N}$

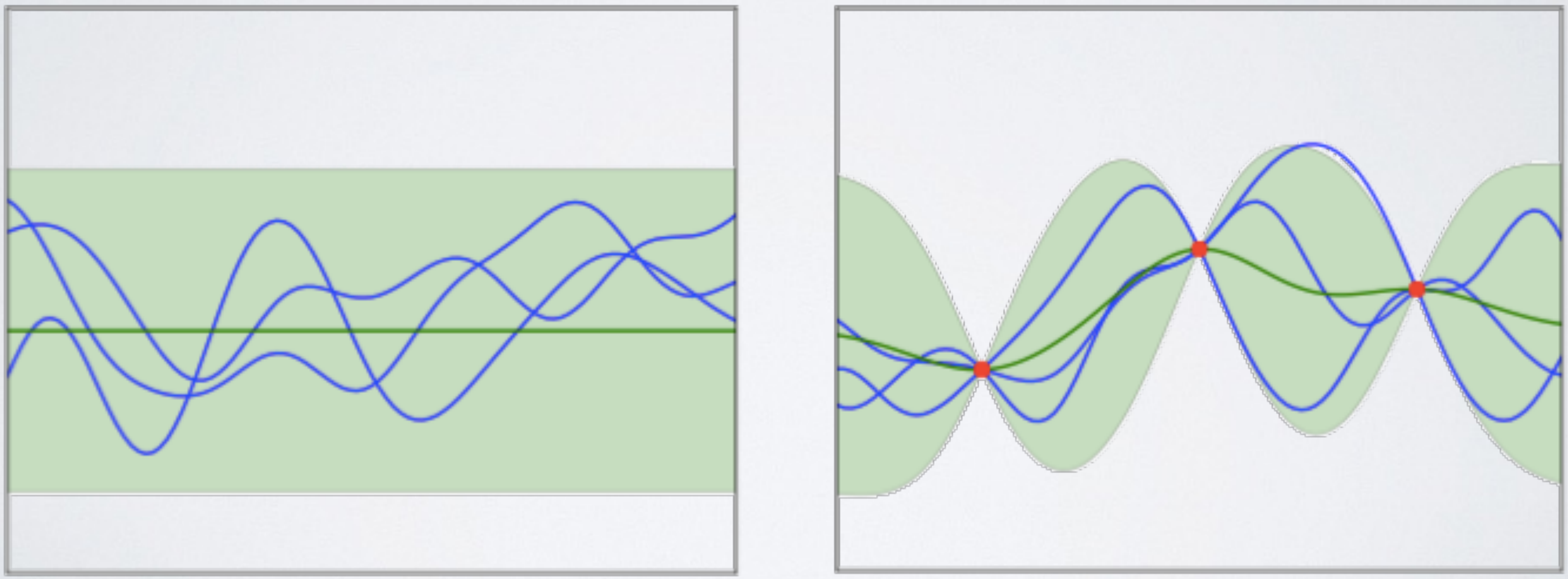
$$\mu(x; \mathbf{X}, \mathbf{y}, \theta) = K(\mathbf{X}, x)^\top K(\mathbf{X}, \mathbf{X})^{-1}(\mathbf{y} - m(\mathbf{X}))$$
$$\Sigma(x, x'; \mathbf{X}, \mathbf{y}, \theta) = K(x, x') - K(\mathbf{X}, x)^\top K(\mathbf{X}, \mathbf{X})^{-1} K(\mathbf{X}, x')$$

- Intuition:
  - A prior for smooth functions
  - Similar inputs (high covariance) have similar outputs
- Can compute expected value and uncertainty for test inputs easily

GPs as Distributions over Functions

Prior

Posterior

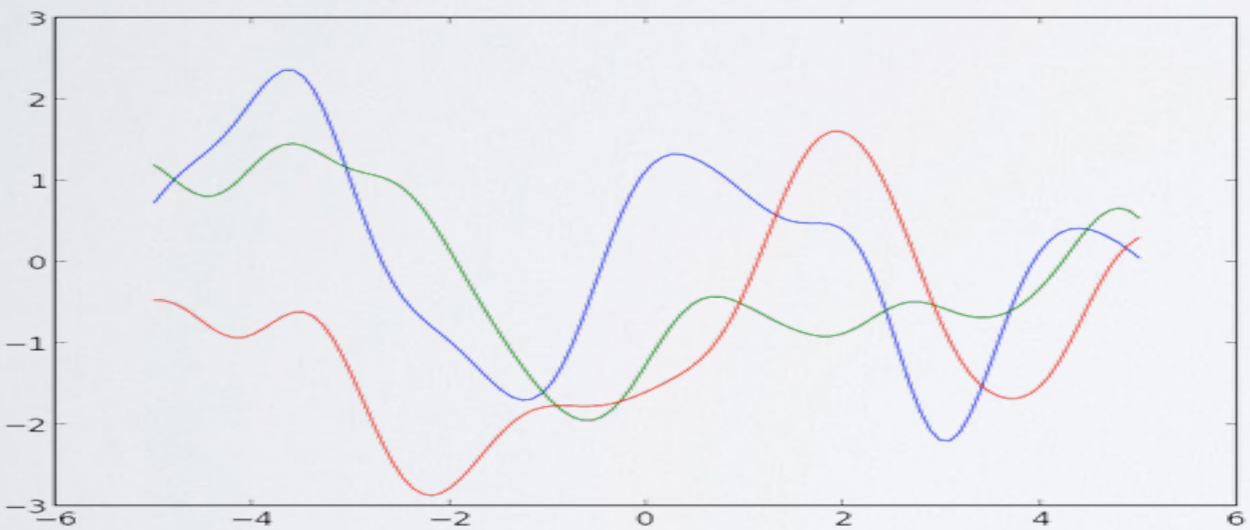


*Samples in blue

GPs Allow High Level Specification

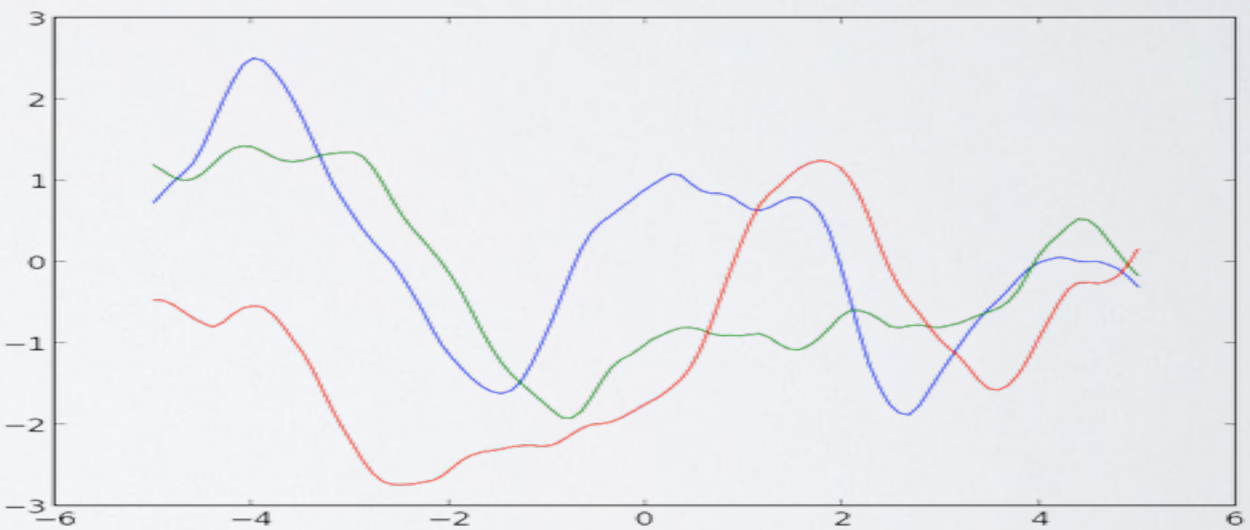
- Gaussian processes are nonparametric, their behaviour is specified at a high level by the choice of *kernel*.

Infinitely differentiable



$$K_{\mathrm{SE}}(x, x') = \theta_0 \exp(-r^2(x, x'))$$

Twice differentiable



$$K_{\mathrm{M52}}(x, x') = \theta_0 \left(1 + \sqrt{5r^2(x, x')} + \tfrac{5}{3}r^2(x, x')\right) \exp\left(-\sqrt{5r^2(x, x')}\right)$$

$$r^2(x, x') = \frac{(x - x')^2}{\theta_1^2}$$

See http://www.cs.toronto.edu/~duvenaud/cookbook/index.html for others

Gaussian Process Notebook

Acquisition Functions

Choosing where to Search Next

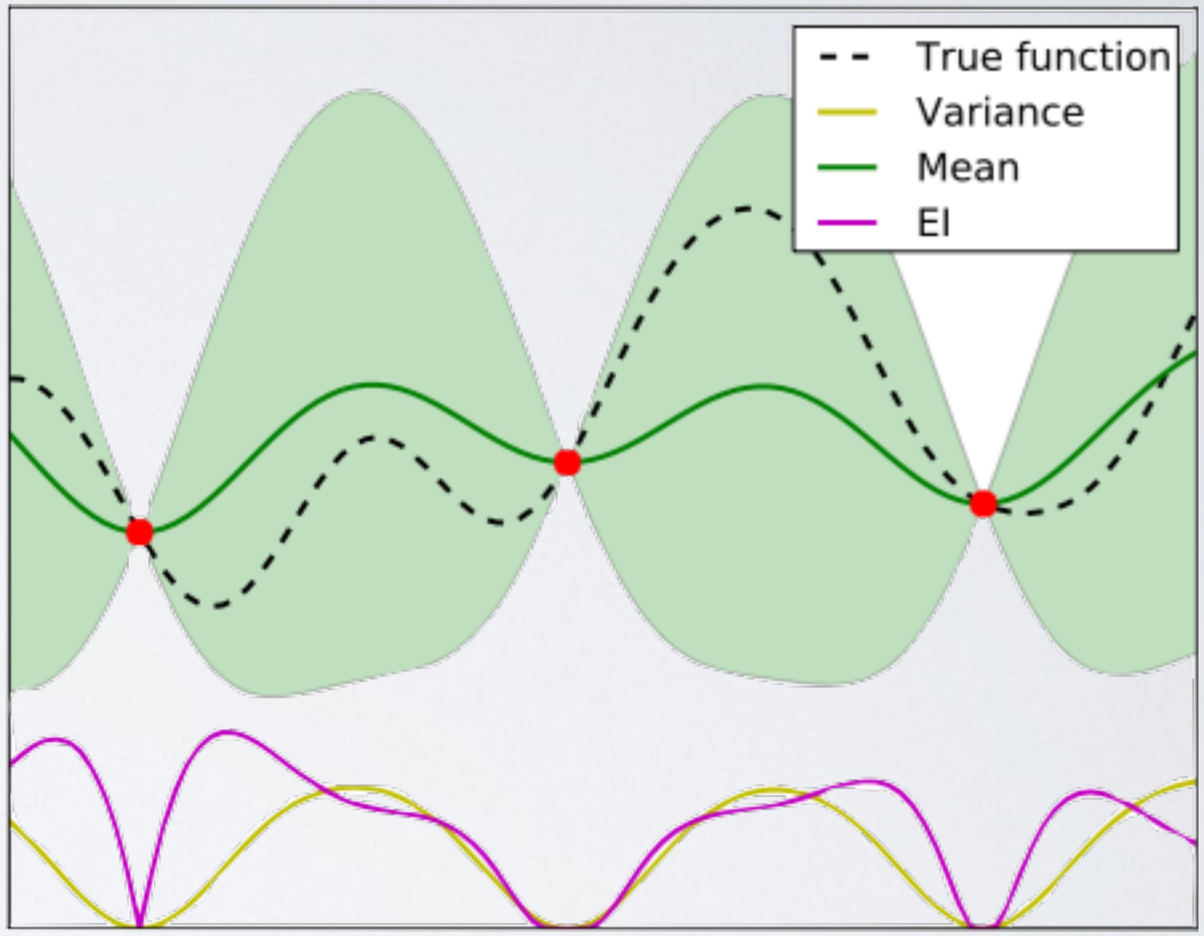
- The GP gives us a mean and variance for each input.

- Minimum expected value: purely exploitative

- Maximum uncertainty: purely explorative

- Expected improvement: trade-off (Mockus, 1978)

- Many other *acquisition functions* have been proposed in the literature



$$\mathrm{EI}(x; \mathbf{X}, \mathbf{y}, \theta) = \int_y \max(0, y_{\text{best}}) P(y|x; \mathbf{X}, \mathbf{y}, \theta) dy$$

Exploration-exploitation tradeoff

Recall the expressions for GP prediction:

$$P(y_{t+1}|\mathcal{D}_{1:t}, \mathbf{x}_{t+1}) = \mathcal{N}(\mu_t(\mathbf{x}_{t+1}), \sigma_t^2(\mathbf{x}_{t+1}) + \sigma^2_{\text{noise}})$$

$$\mu_t(\mathbf{x}_{t+1}) = \mathbf{k}^T[\mathbf{K} + \sigma^2_{\text{noise}}\mathbf{I}]^{-1}\mathbf{y}_{1:t}$$

$$\sigma_t^2(\mathbf{x}_{t+1}) = k(\mathbf{x}_{t+1}, \mathbf{x}_{t+1}) - \mathbf{k}^T[\mathbf{K} + \sigma^2_{\text{noise}}\mathbf{I}]^{-1}\mathbf{k}$$

Exploration-exploitation tradeoff

Recall the expressions for GP prediction:

$$P(y_{t+1}|\mathcal{D}_{1:t}, \mathbf{x}_{t+1}) = \mathcal{N}(\mu_t(\mathbf{x}_{t+1}), \sigma_t^2(\mathbf{x}_{t+1}) + \sigma_{\text{noise}}^2)$$

$$\mu_t(\mathbf{x}_{t+1}) = \mathbf{k}^T[\mathbf{K} + \sigma_{\text{noise}}^2\mathbf{I}]^{-1}\mathbf{y}_{1:t}$$

$$\sigma_t^2(\mathbf{x}_{t+1}) = k(\mathbf{x}_{t+1}, \mathbf{x}_{t+1}) - \mathbf{k}^T[\mathbf{K} + \sigma_{\text{noise}}^2\mathbf{I}]^{-1}\mathbf{k}$$

We should choose the next point ***x*** where the mean is high (exploitation) and the variance is high (exploration).

Exploration-exploitation tradeoff

Recall the expressions for GP prediction:

$$P(y_{t+1}|\mathcal{D}_{1:t}, \mathbf{x}_{t+1}) = \mathcal{N}(\mu_t(\mathbf{x}_{t+1}), \sigma_t^2(\mathbf{x}_{t+1}) + \sigma_{\text{noise}}^2)$$

$$\mu_t(\mathbf{x}_{t+1}) = \mathbf{k}^T[\mathbf{K} + \sigma_{\text{noise}}^2\mathbf{I}]^{-1}\mathbf{y}_{1:t}$$

$$\sigma_t^2(\mathbf{x}_{t+1}) = k(\mathbf{x}_{t+1}, \mathbf{x}_{t+1}) - \mathbf{k}^T[\mathbf{K} + \sigma_{\text{noise}}^2\mathbf{I}]^{-1}\mathbf{k}$$

We should choose the next point ***x*** where the mean is high (exploitation) and the variance is high (exploration).

We could balance this tradeoff with an acquisition function as follows:

$$\mu(\mathbf{x}) + \kappa\sigma(\mathbf{x})$$

Probability of Improvement

An acquisition function: Probability of Improvement

$$\mathrm{PI}(\mathbf{x}) = P(f(\underline{\mathbf{x}}) \geq \mu^{+} + \cancelto{0}{\xi})$$

μ^+ best observed value

An acquisition function: Probability of Improvement

μ^+ best observed value

$$
\begin{aligned}
\mathrm{PI}(\mathbf{x}) &= P(f(\mathbf{x}) \geq \mu^+ + \cancelto{0}{\xi}) \\
&= \Phi\left(\frac{\mu(\mathbf{x}) - \mu^+ - \xi}{\sigma(\mathbf{x})}\right)
\end{aligned}
$$

cumulative

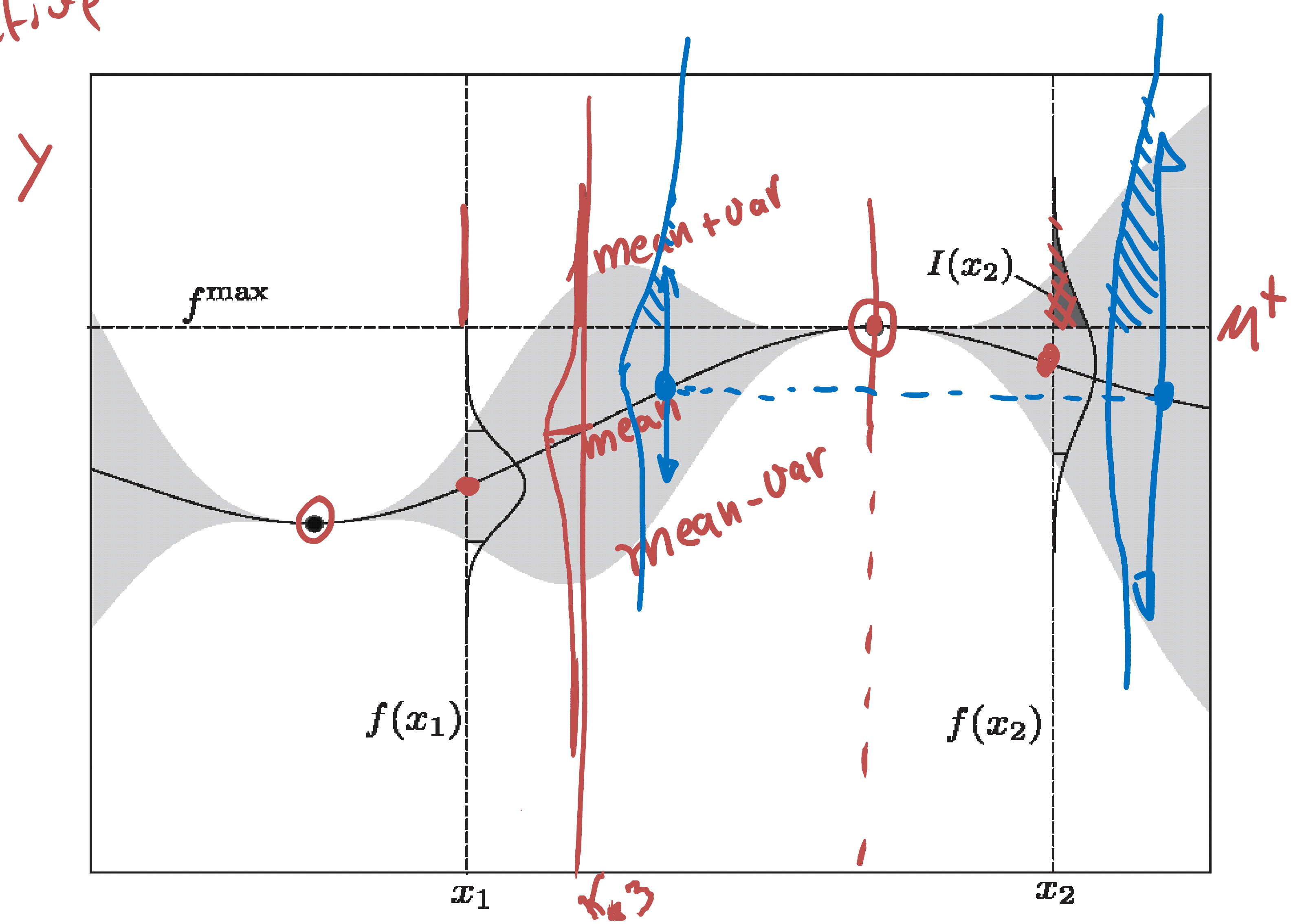
An acquisition function: Probability of Improvement

$$\mathrm{PI}(\mathbf{x}) = P(f(\mathbf{x}) \geq \mu^{+} + \cancel{\xi})$$
$$= \Phi\left(\frac{\mu(\mathbf{x}) - \mu^{+} - \xi}{\sigma(\mathbf{x})}\right)$$

cumulative

μ^{+} best observed value

$f^{max} = \mu^{+} + \xi$ small



Expected Improvement

Bayes and decision theory

Utilitarian view: We need models to make the right decisions under uncertainty. Inference and decision making are intertwined.

Bayes and decision theory

Utilitarian view: We need models to make the right decisions under uncertainty. Inference and decision making are intertwined.

Learned posterior

$$\begin{cases} P(x=\text{healthy}|\text{data}) = 0.9 \\ P(x=\text{cancer}|\text{data}) = 0.1 \end{cases}$$

Cost/Reward model ***u(x,a)***

	$\mathbf{a} = \mathit{no\ treatment}$	$\mathbf{a} = \mathit{treatment}$
$\mathbf{x} = \mathit{healthy}$	0	-30
$\mathbf{x} = \mathit{cancer}$	-100	-20

Bayes and decision theory

Utilitarian view: We need models to make the right decisions under uncertainty. Inference and decision making are intertwined.

Learned posterior

$$\begin{cases} P(x=\text{healthy}|data) = 0.9 \\ P(x=\text{cancer}|data) = 0.1 \end{cases}$$

Cost/Reward model ***u(x,a)***

	$\mathbf{a} = \textit{no treatment}$	$\mathbf{a} = \textit{treatment}$
$\mathbf{x} = \textit{healthy}$	0	-30
$\mathbf{x} = \textit{cancer}$	-100	-20

We choose the action that maximizes the expected utility, or equivalently, which minimizes the expected cost.

$$EU(a) = \sum u(x,a)\, P(x|data)$$

Bayes and decision theory

Utilitarian view: We need models to make the right decisions under uncertainty. Inference and decision making are intertwined.

Learned posterior

$$\begin{cases} P(x=\text{healthy}|\text{data}) = 0.9 \\ P(x=\text{cancer}|\text{data}) = 0.1 \end{cases}$$

Cost/Reward model *u(x,a)*

	$\mathbf{a} = no\ treatment$	$\mathbf{a} = treatment$
$\mathbf{x} = healthy$	0	-30
$\mathbf{x} = cancer$	-100	-20

We choose the action that maximizes the expected utility, or equivalently, which minimizes the expected cost.

$$EU(a) = \sum_x u(x,a)\, P(x|data)$$

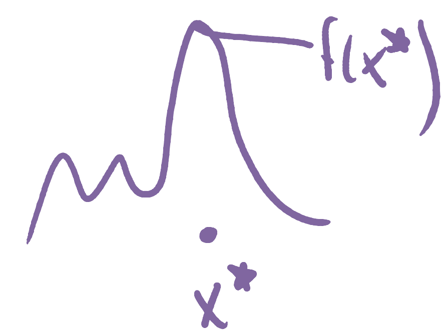
$EU(a=\text{treatment}) =$ $u(\text{healthy}, \text{treatment})\, P(x=\text{healthy}|d) + u(\text{cancer}, \text{treatmt.})\, P(\text{can}|d)$

$(-30)(0.9) + (-20)(0.1) \approx$

$EU(a=\text{no treatment}) =$

An expected utility criterion

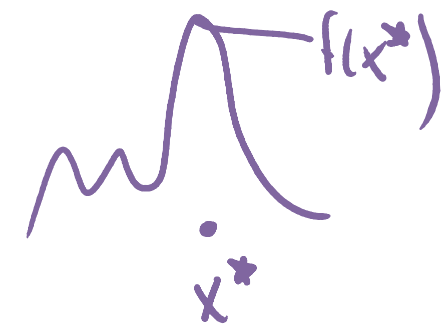
At iteration *n+1*, choose the point that minimizes the distance to the objective evaluated at the maximum x^*:



$$\begin{aligned}
\mathbf{x}_{n+1} &= \arg\min_{\mathbf{x}} \mathbb{E}(\|\overset{\text{GP}}{f_{n+1}}(\mathbf{x}) - \overset{\text{true } f}{f}(\mathbf{x}^\star)\| \,|\mathcal{D}_n) \\
&= \arg\min_{\mathbf{x}} \int \|f_{n+1}(\mathbf{x}) - f(\mathbf{x}^\star)\| p(f_{n+1}|\mathcal{D}_n) df_{n+1}
\end{aligned}$$

An expected utility criterion

At iteration *n+1*, choose the point that minimizes the distance to the objective evaluated at the maximum x^*:



GP true f

$$
\begin{aligned}
\mathbf{x}_{n+1} &= \arg\min_{\mathbf{x}} \mathbb{E}(\|f_{n+1}(\mathbf{x}) - f(\mathbf{x}^\star)\| \,|\mathcal{D}_n) \\
&= \arg\min_{\mathbf{x}} \int \|f_{n+1}(\mathbf{x}) - f(\mathbf{x}^\star)\| p(f_{n+1}|\mathcal{D}_n) df_{n+1}
\end{aligned}
$$

We don't know the true objective at the maximum. To overcome this, Mockus proposed the following acquisition function:

$$
\mathbf{x} = \arg\max_{\mathbf{x}} \mathbb{E}(\max\{0, f_{n+1}(\mathbf{x}) - f^{\max}\} \,|\mathcal{D}_n)
$$

Expected improvement

$$\mathbf{x} = \arg\max_{\mathbf{x}} \mathbb{E}(\max\{0, f_{n+1}(\mathbf{x}) - \underbrace{f^{\max}}_{\mu^+ + \xi}\} \,|\mathcal{D}_n)$$

For this acquisition, we can obtain an analytical expression:

$$\mathrm{EI}(\mathbf{x}) = \begin{cases} (\mu(\mathbf{x}) - \mu^+ - \xi)\Phi(Z) + \sigma(\mathbf{x})\phi(Z) & \text{if } \sigma(\mathbf{x}) > 0 \\ 0 & \text{if } \sigma(\mathbf{x}) = 0 \end{cases}$$

$$Z = \frac{\mu(\mathbf{x}) - \mu^+ - \xi}{\sigma(\mathbf{x})}$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ denote the PDF and CDF of the standard Normal

Acquisition functions

$$\boxed{\mu^+ = \text{argmax}_{\mathbf{x}_i \in \mathbf{x}_{1:t}} \mu(\mathbf{x}_i)}$$

- Probability of Improvement

$$\text{PI}(\mathbf{x}) = \Phi\left(\frac{\mu(\mathbf{x}) - \mu^+ - \xi}{\sigma(\mathbf{x})}\right)$$

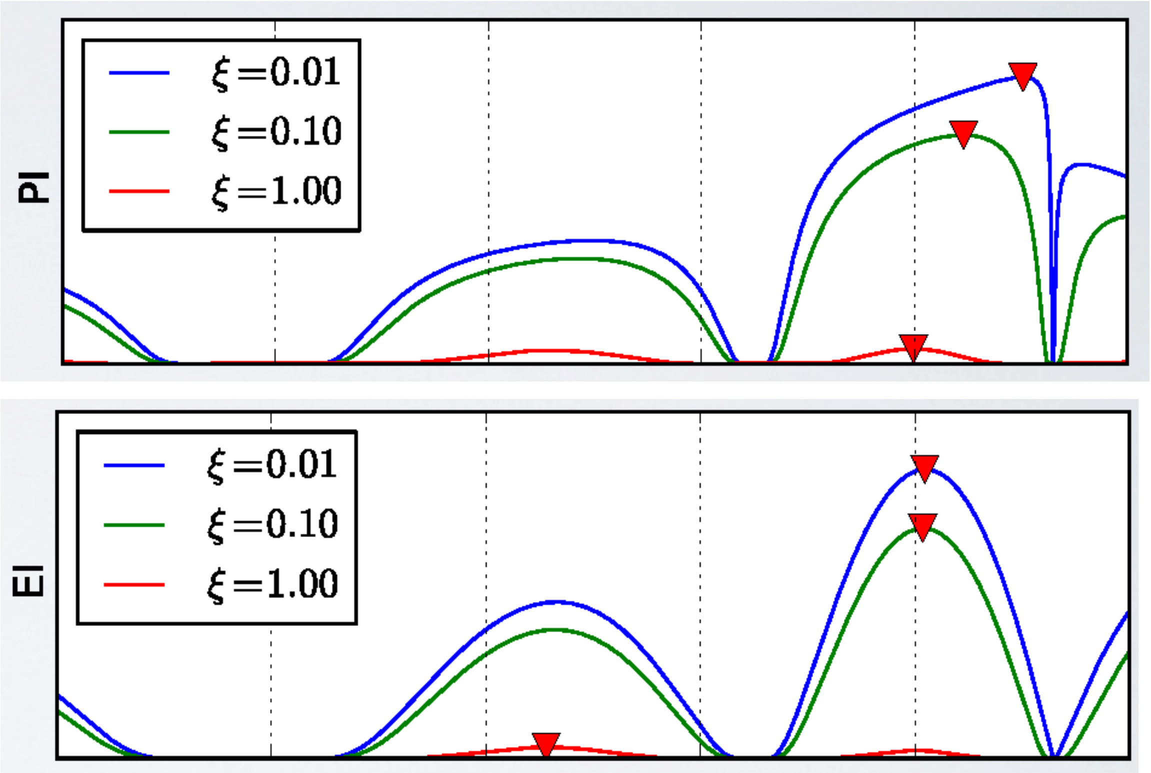
Kushner 1964

- Expected Improvement

$$\text{EI}(\mathbf{x}) = (\mu(\mathbf{x}) - \mu^+ - \xi)\Phi(Z) + \sigma(\mathbf{x})\phi(Z)$$

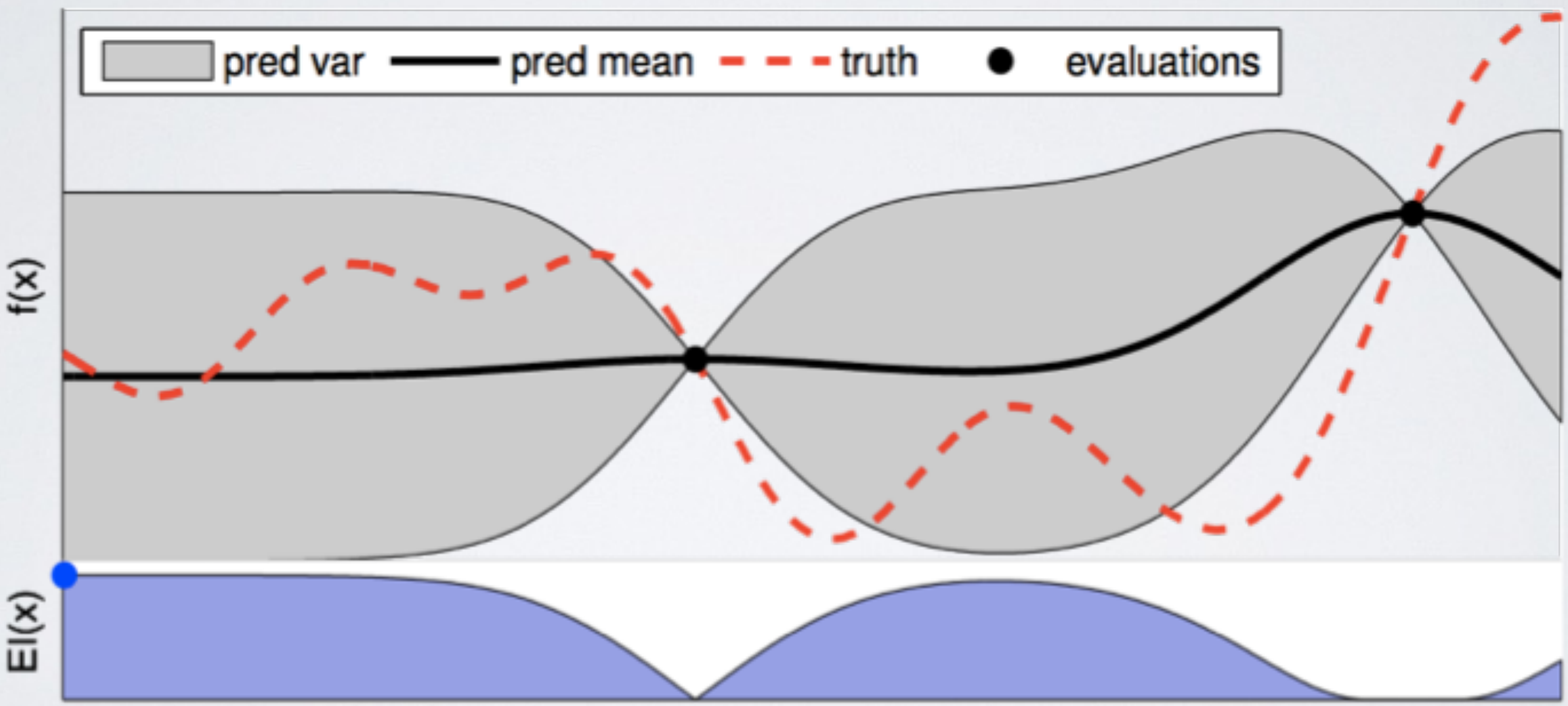
$$Z = \frac{\mu(\mathbf{x}) - \mu^+ - \xi}{\sigma(\mathbf{x})}$$

Mockus 1978

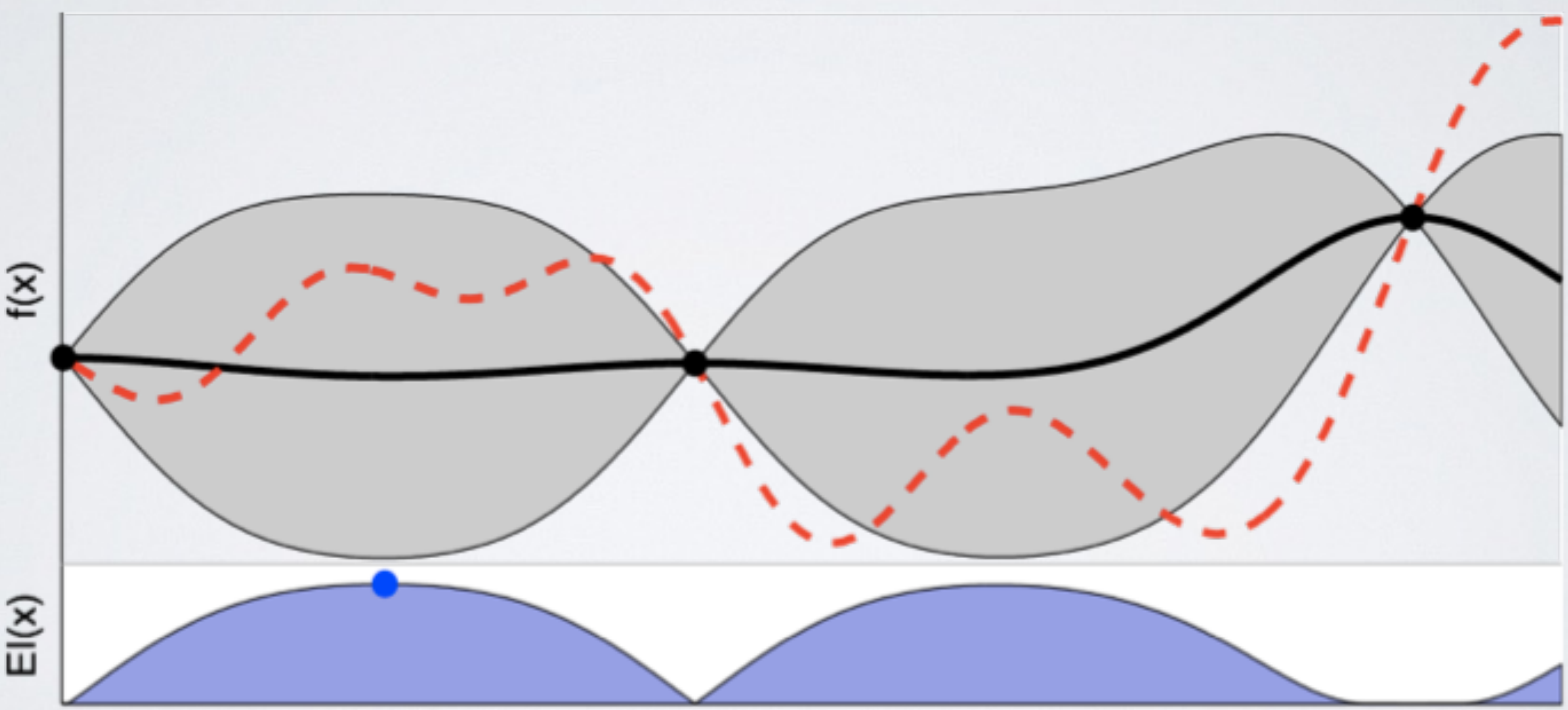


Example: Expected Improvement

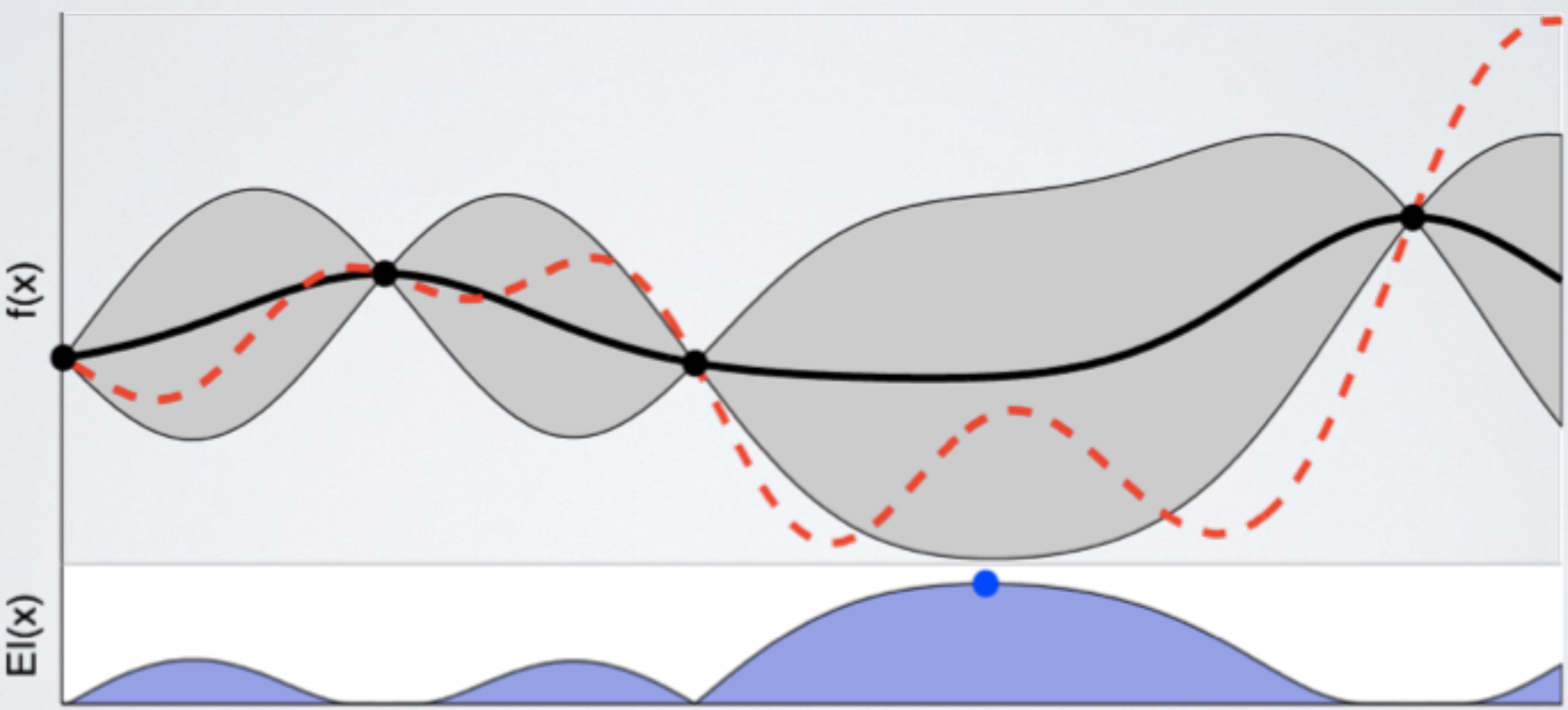
Simple Example



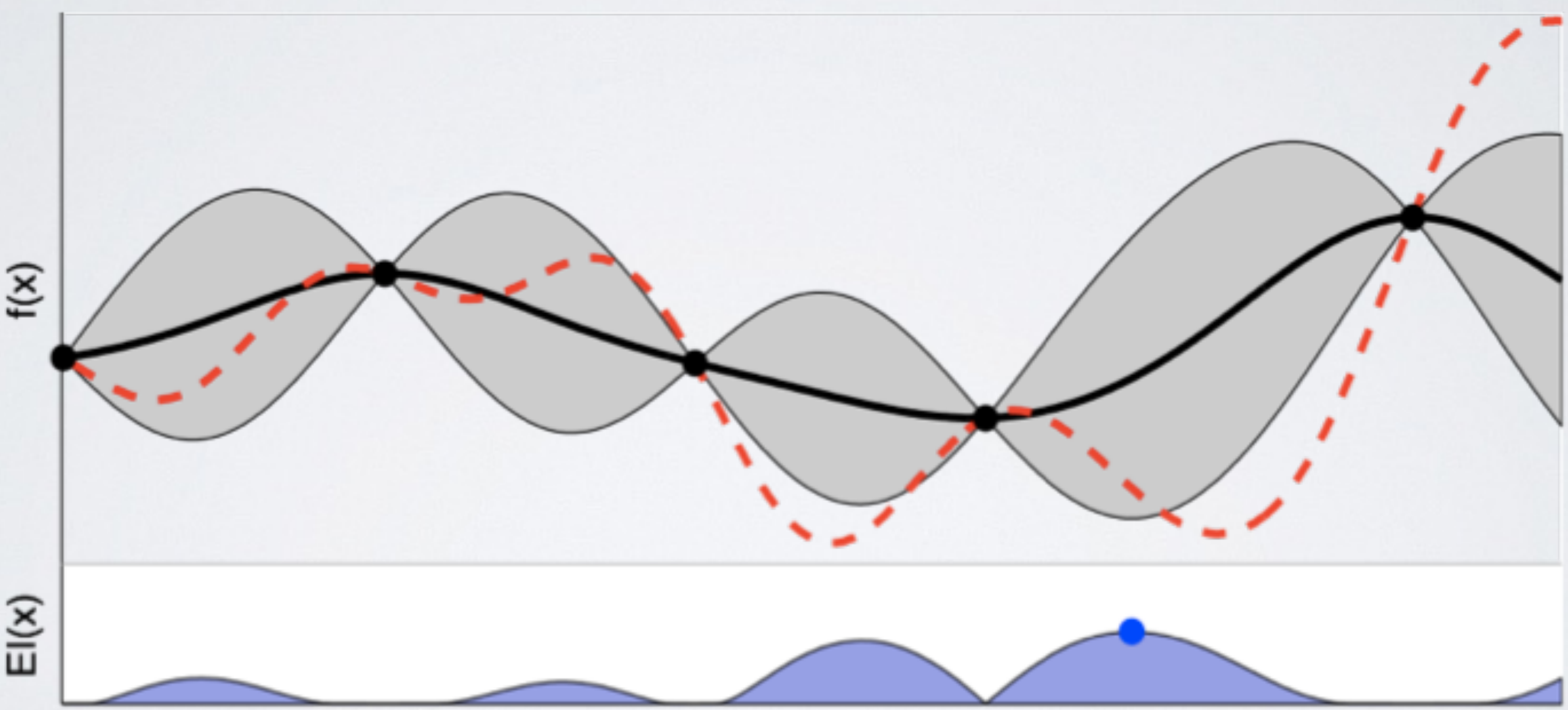
Simple Example



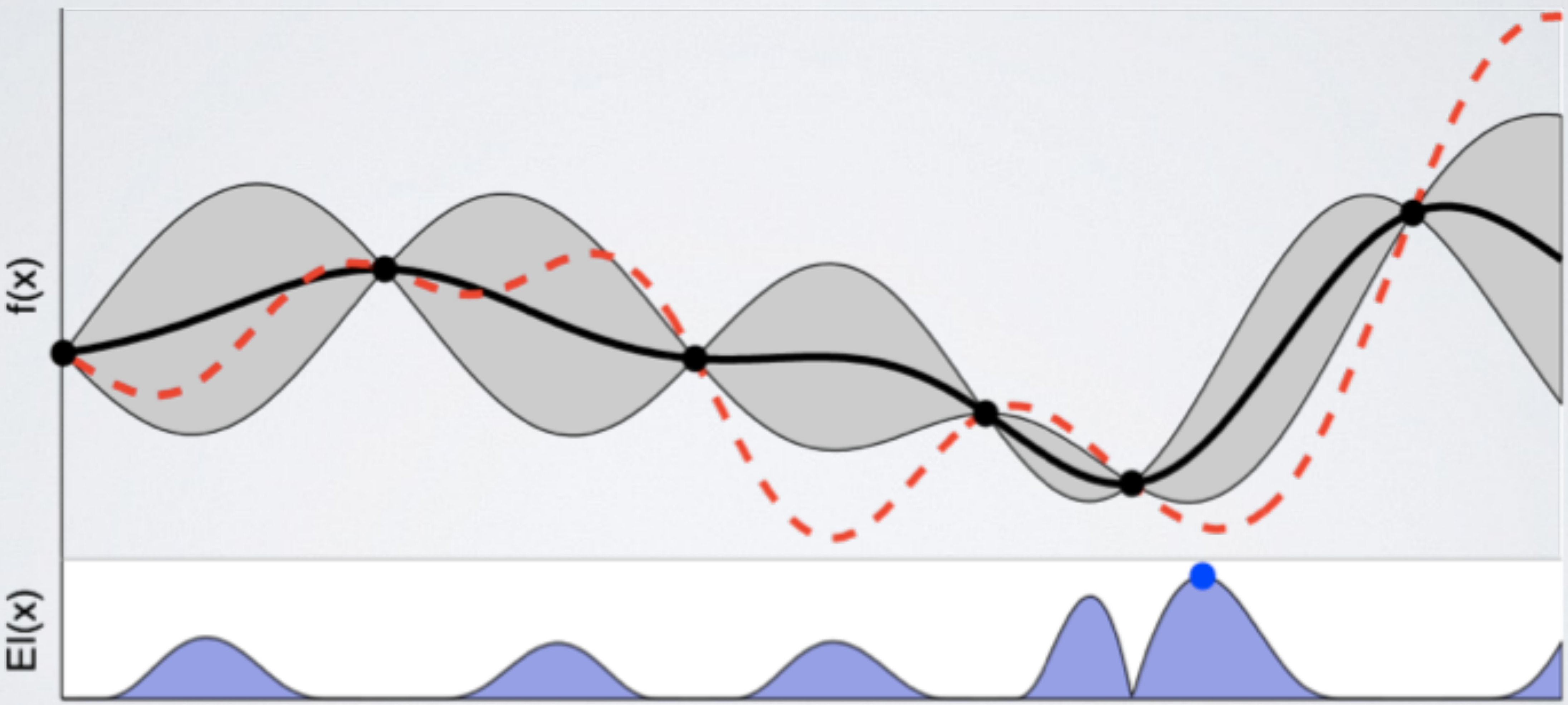
Simple Example



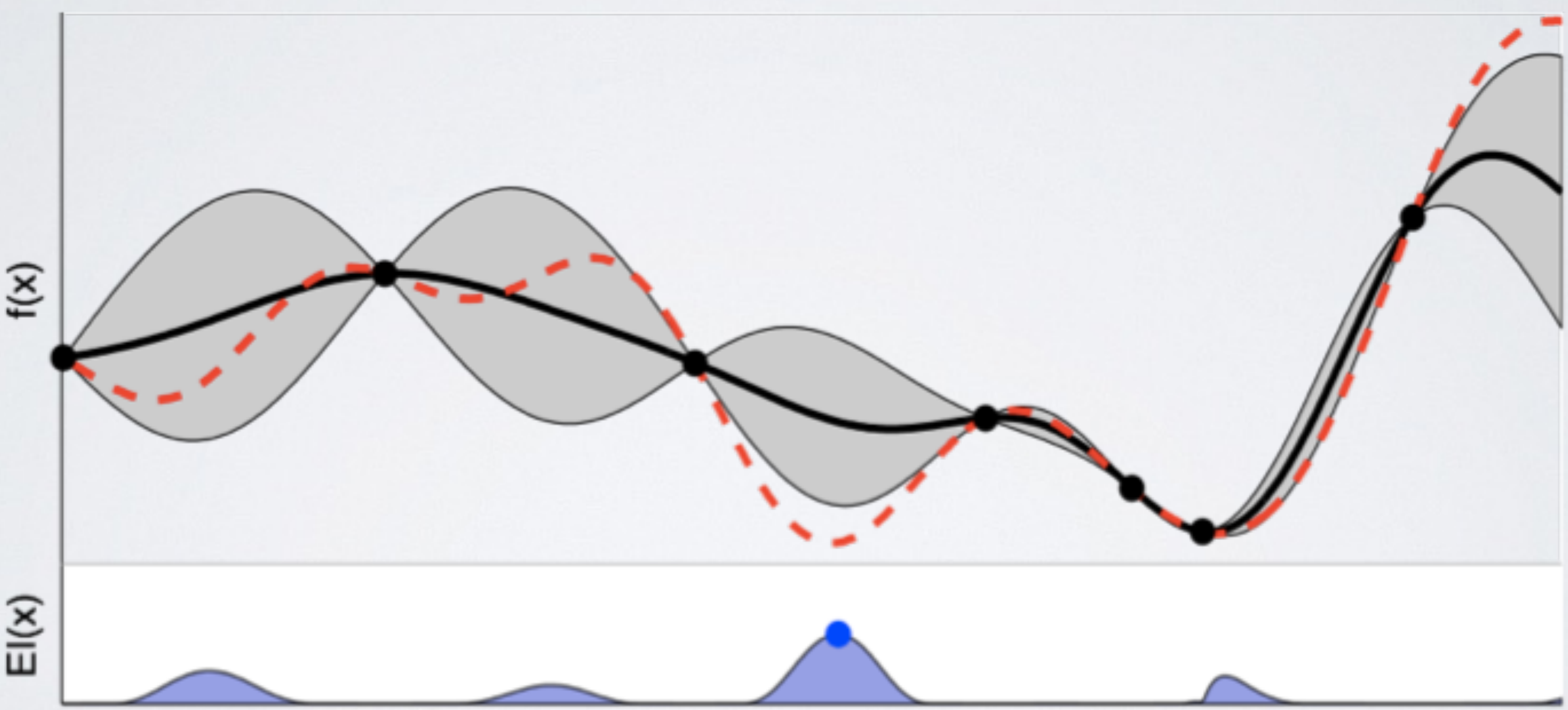
Simple Example



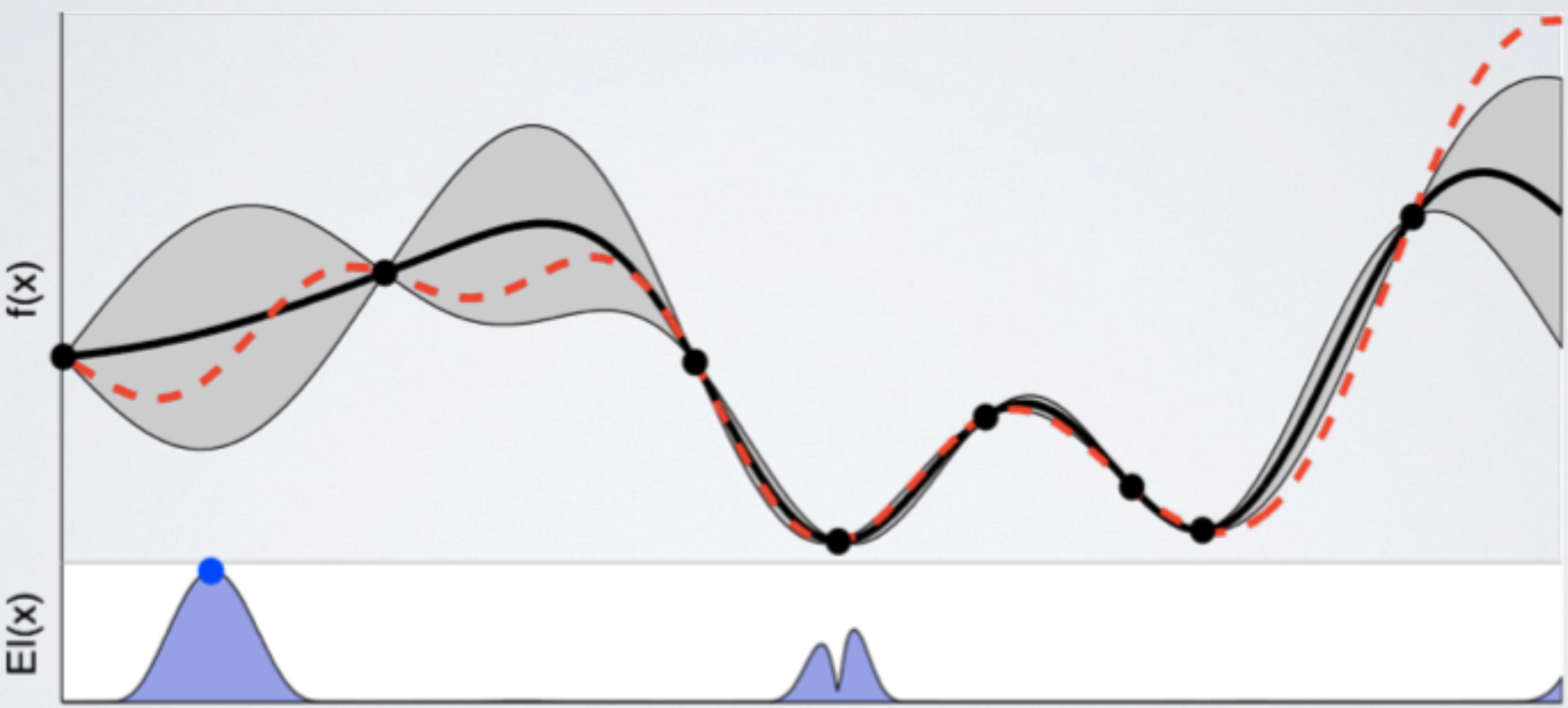
Simple Example



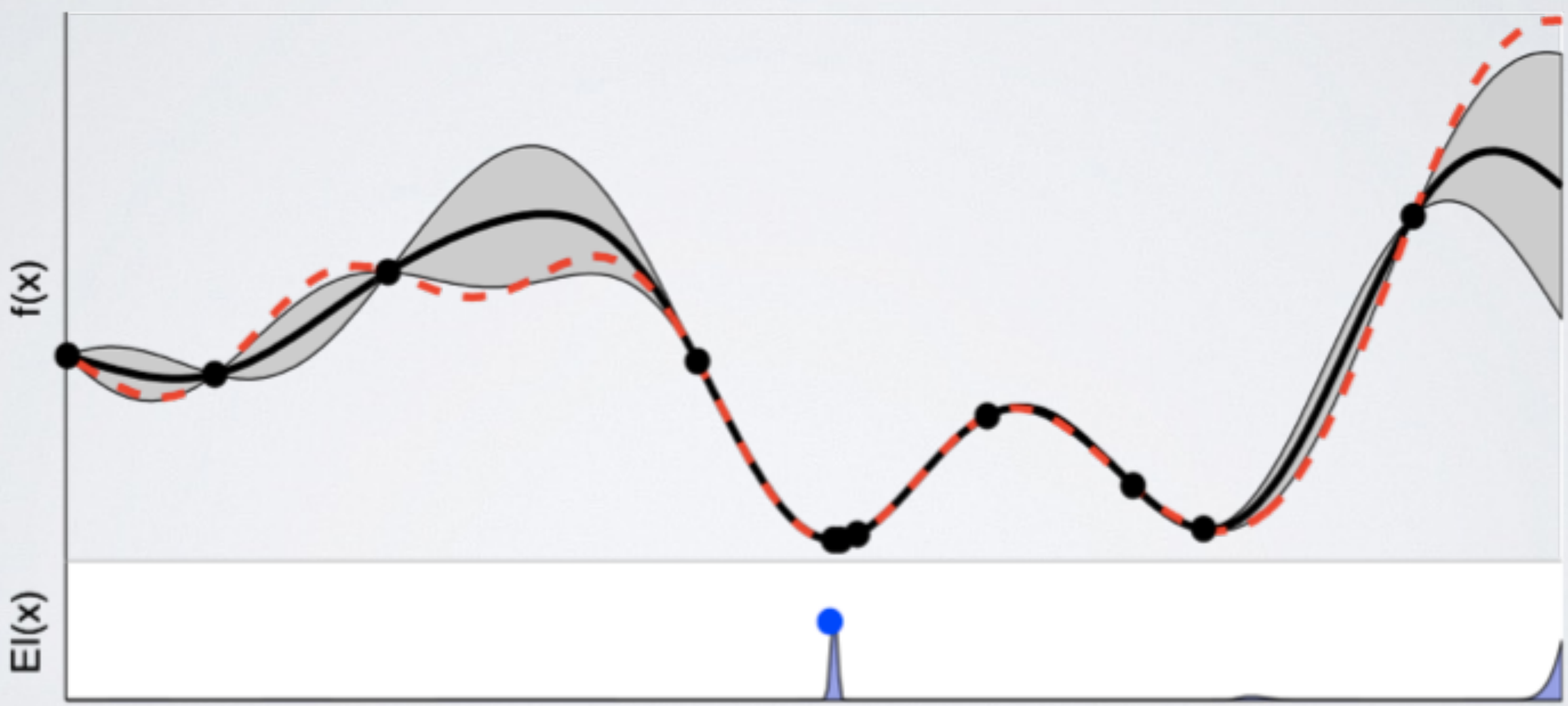
Simple Example



Simple Example



Simple Example



Check out Spearmint:
https://github.com/HIPS/Spearmint

Bayesian Optimization package for Python, Matlab