Regret bounds for meta Bayesian optimization with an **unknown** Gaussian process prior

Zi Wang* | Beomjoon Kim* | Leslie Pack Kaelbling

Dec 5 @ NeurIPS 18

CSAIL

LIS LEARNING & INTELLIGENT SYSTEMS

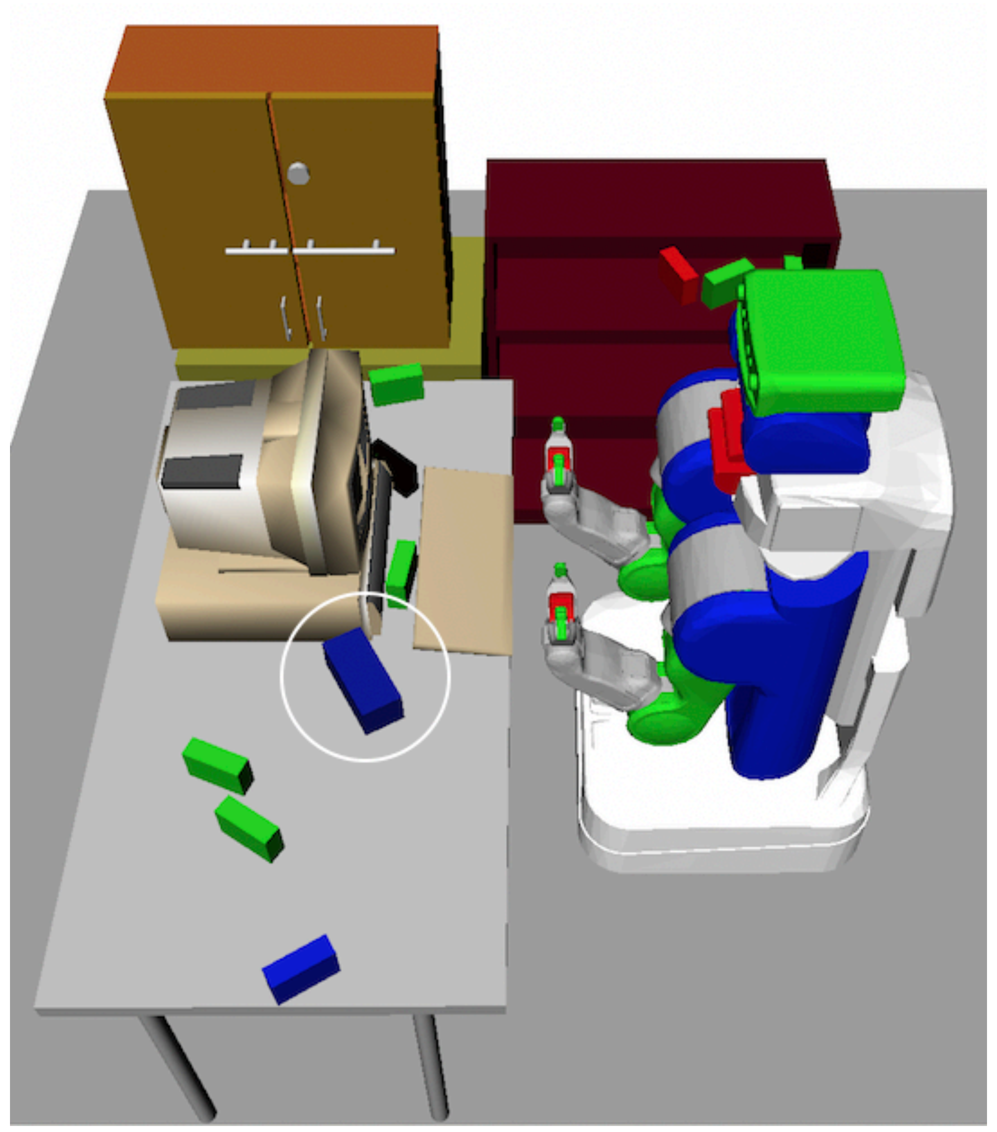
Poster #22

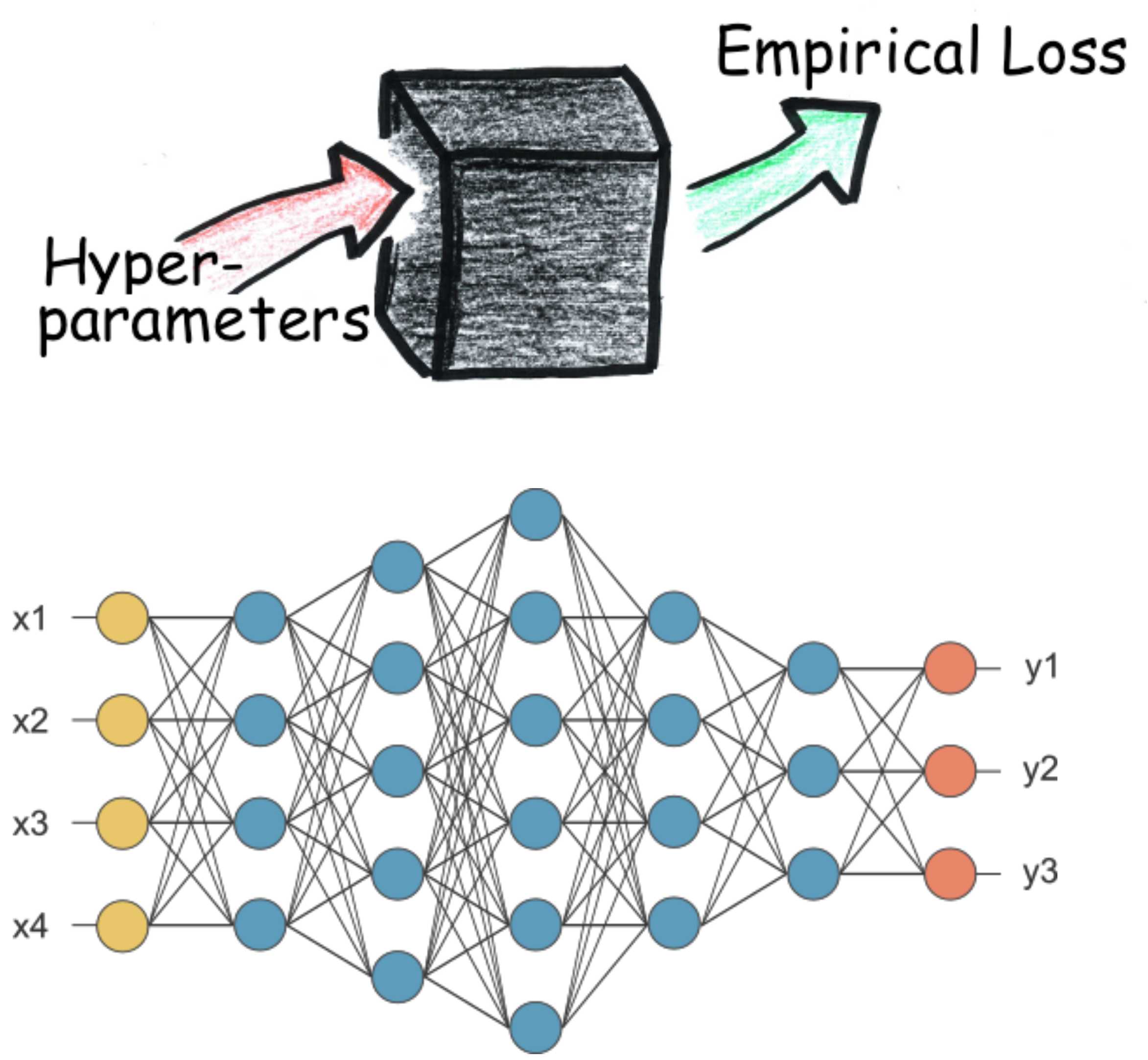
Bayesian optimization

Goal: $$x^* = \underset{x \in \mathfrak{X}}{\operatorname{argmax}} f(x)$$

Challenges:

- f is expensive to evaluate
- f is multi-peak
- no gradient information
- evaluations can be noisy



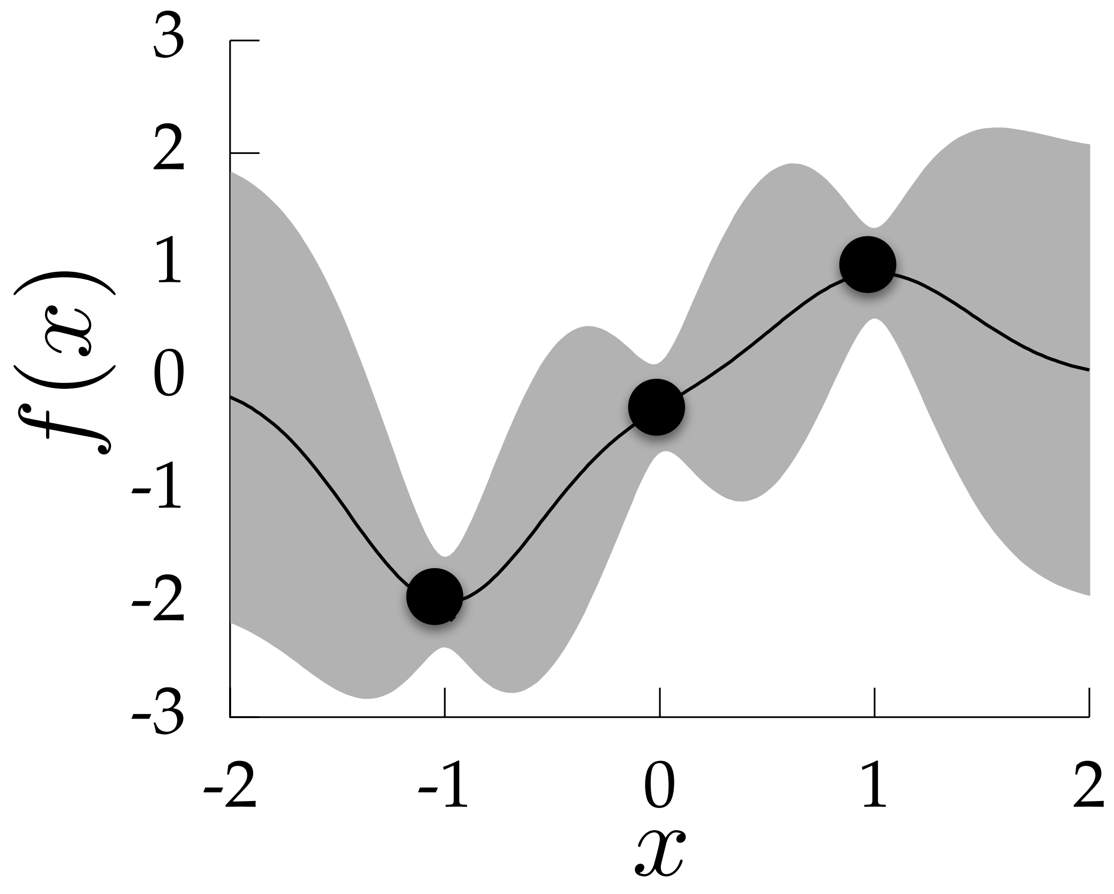


Bayesian optimization

Goal: $x^* = \underset{x \in \mathfrak{X}}{\operatorname{argmax}} f(x)$

Challenges:

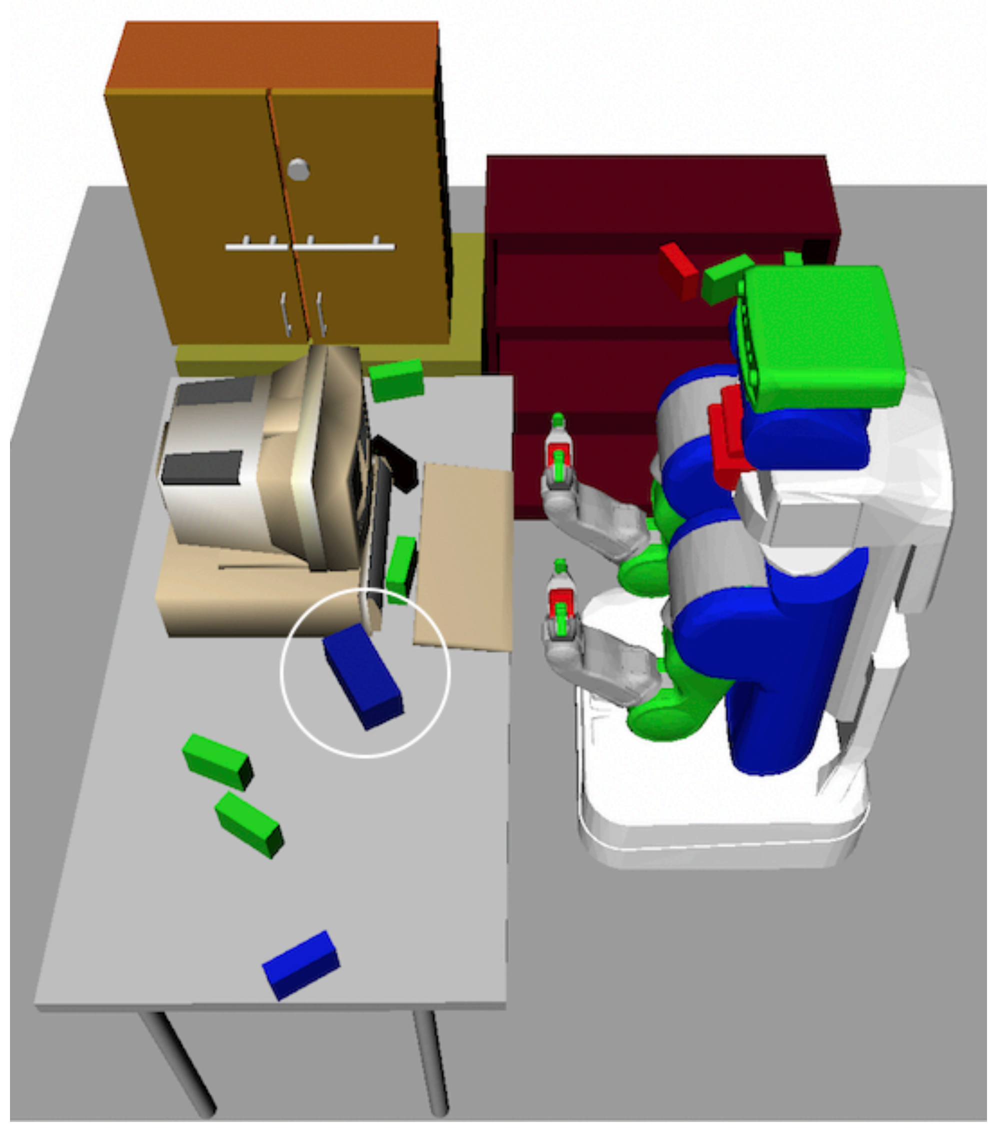
- f is expensive to evaluate
- f is multi-peak
- no gradient information
- evaluations can be noisy

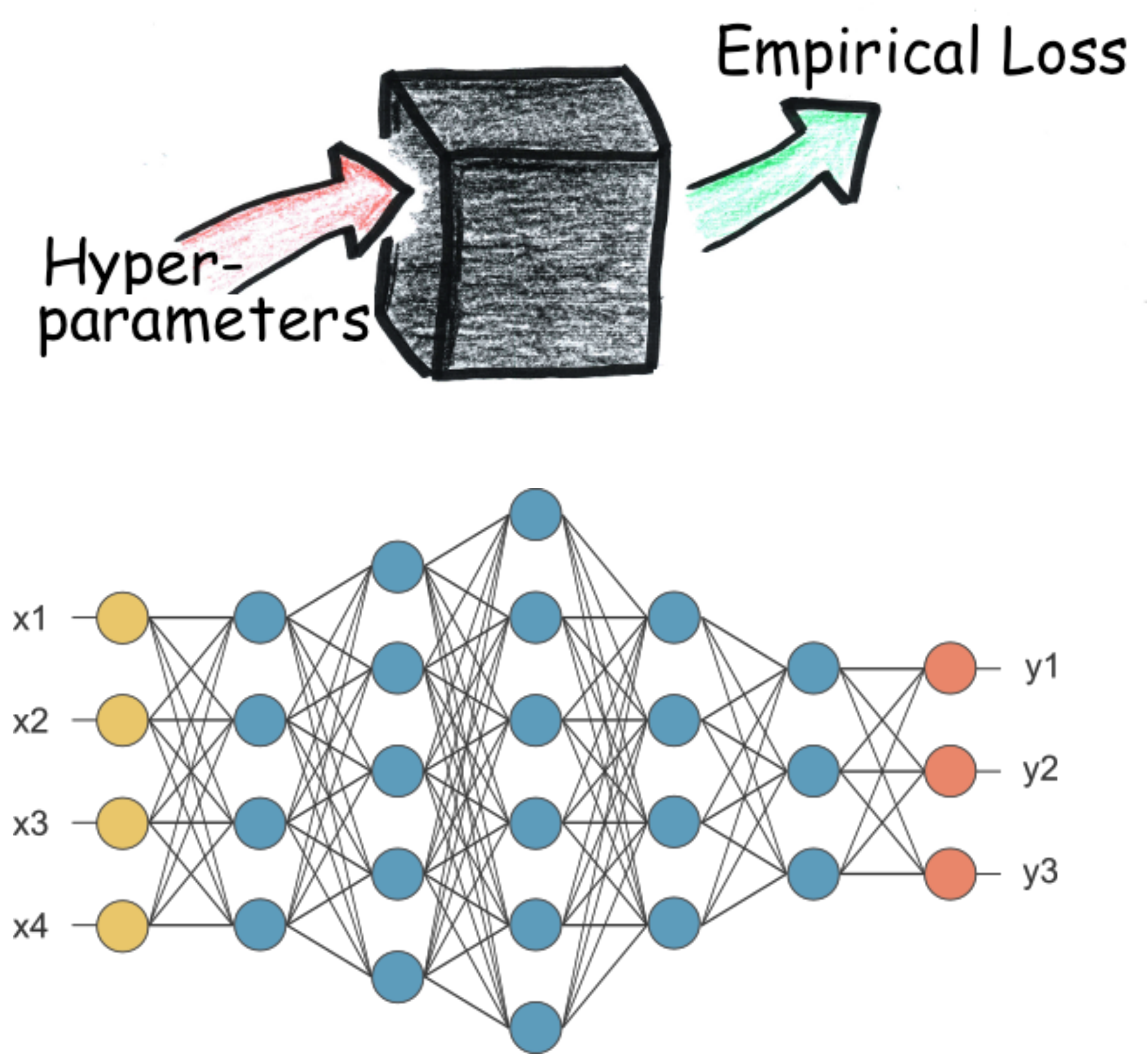


Assume a GP prior $f \sim GP(\mu, k)$

LOOP

- choose new query point(s) to evaluate
- compute the posterior GP model



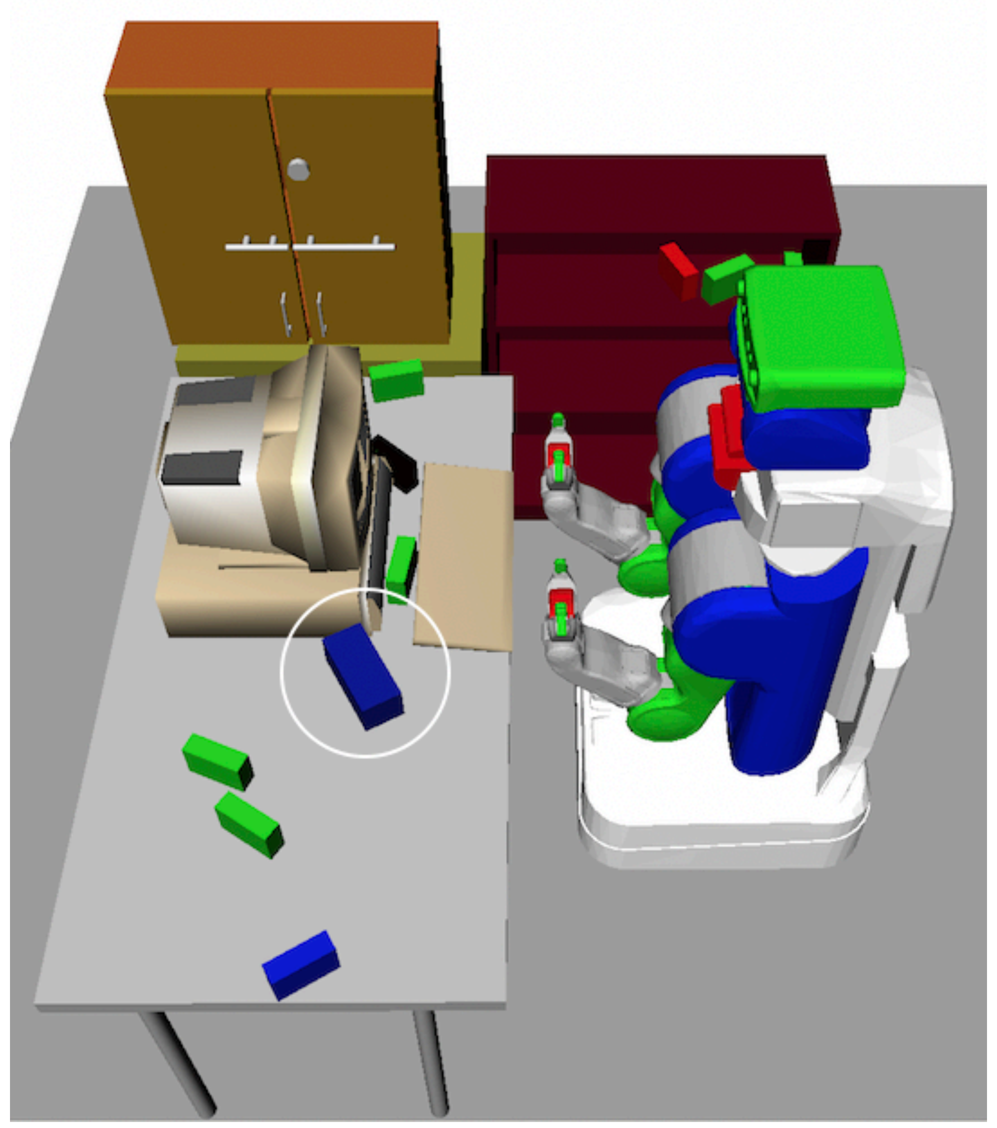


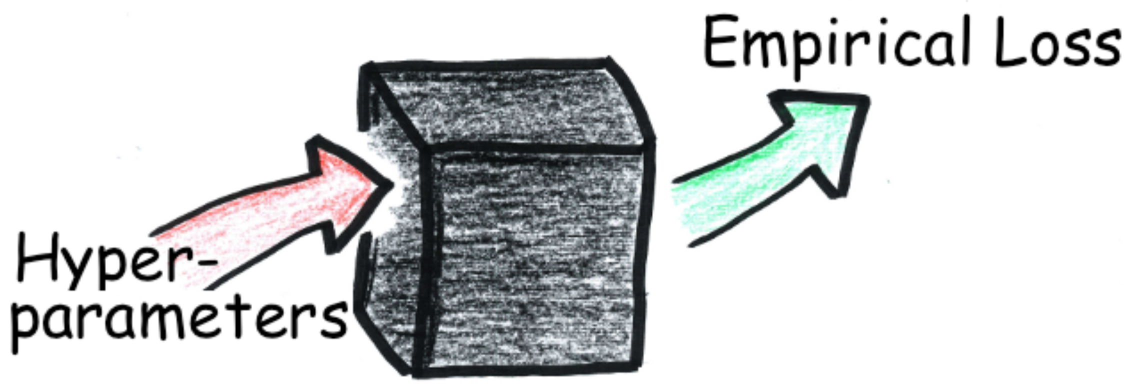
Bayesian optimization

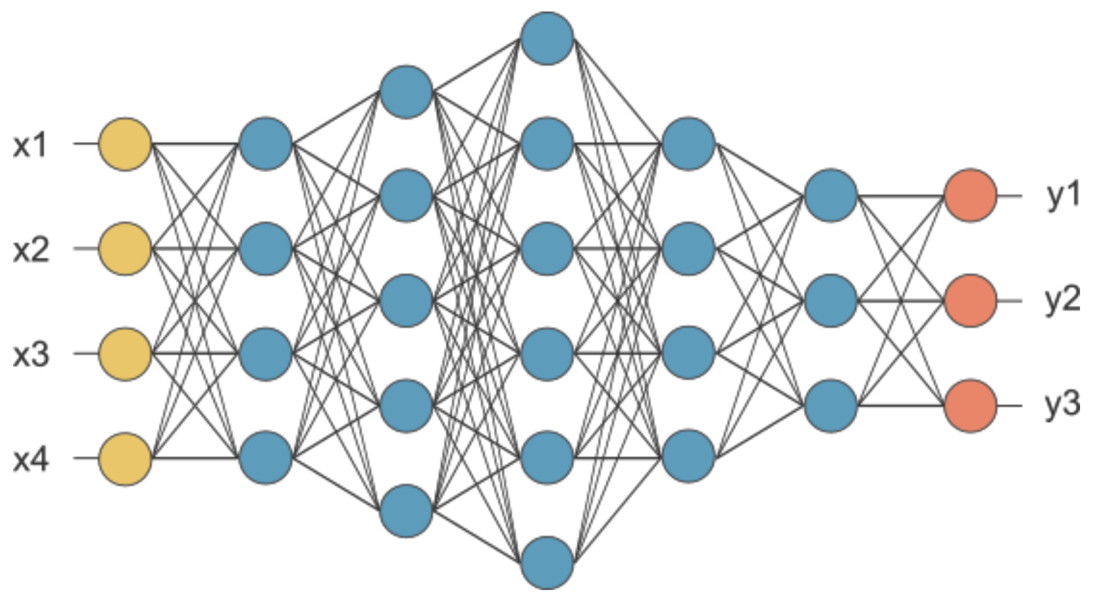
Goal: $$x^* = \underset{x \in \mathfrak{X}}{\operatorname{argmax}} f(x)$$

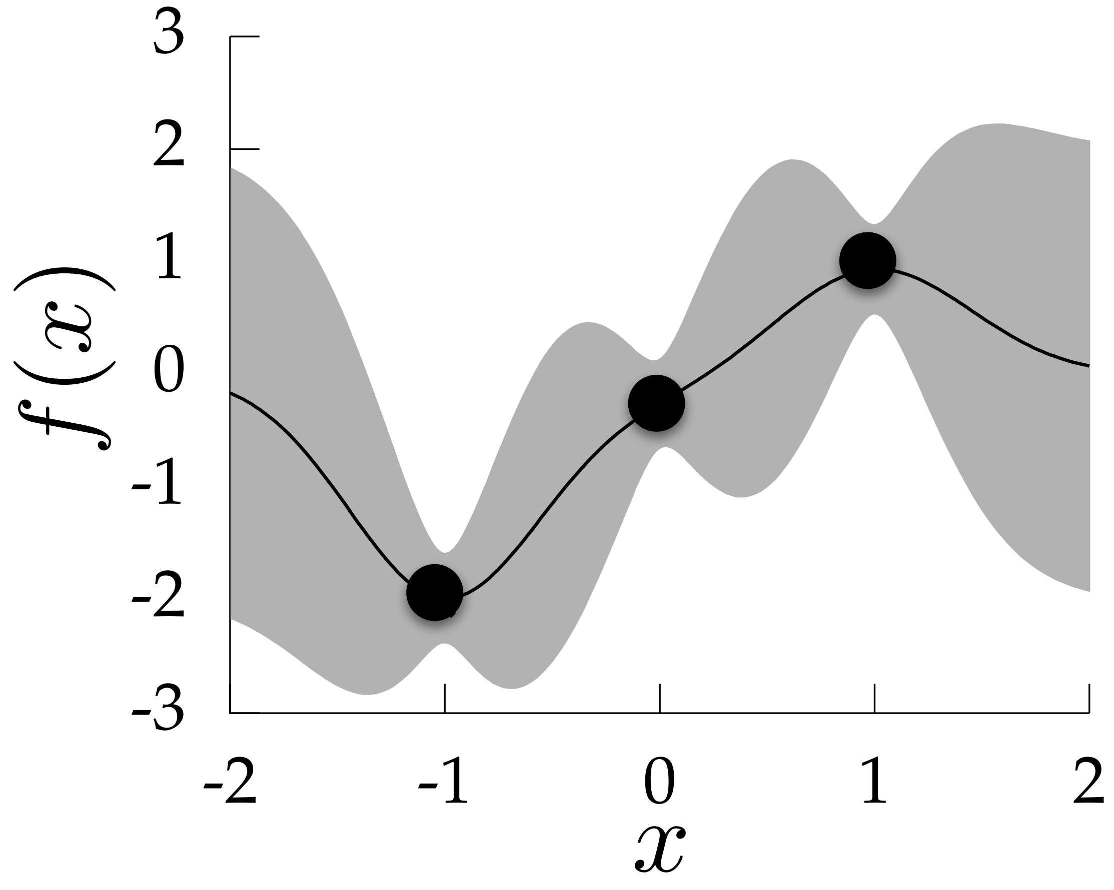
Challenges:

- f is expensive to evaluate
- f is multi-peak
- no gradient information
- evaluations can be noisy









Assume a GP prior $f \sim GP(\mu, k)$

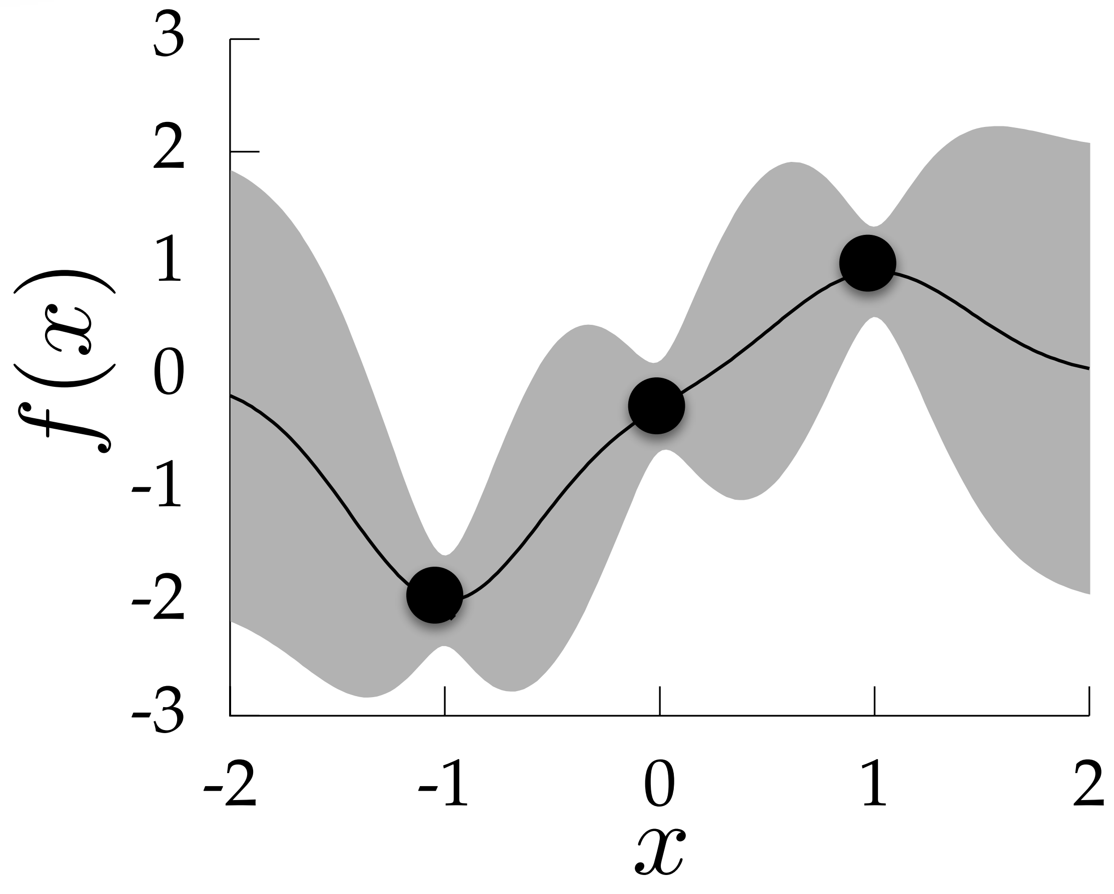
LOOP

- choose new query point(s) to evaluate
- compute the posterior GP model

How to choose the prior?

Bayesian optimization

Goal: $x^* = \underset{x \in \mathfrak{X}}{\operatorname{argmax}} f(x)$



Challenges:

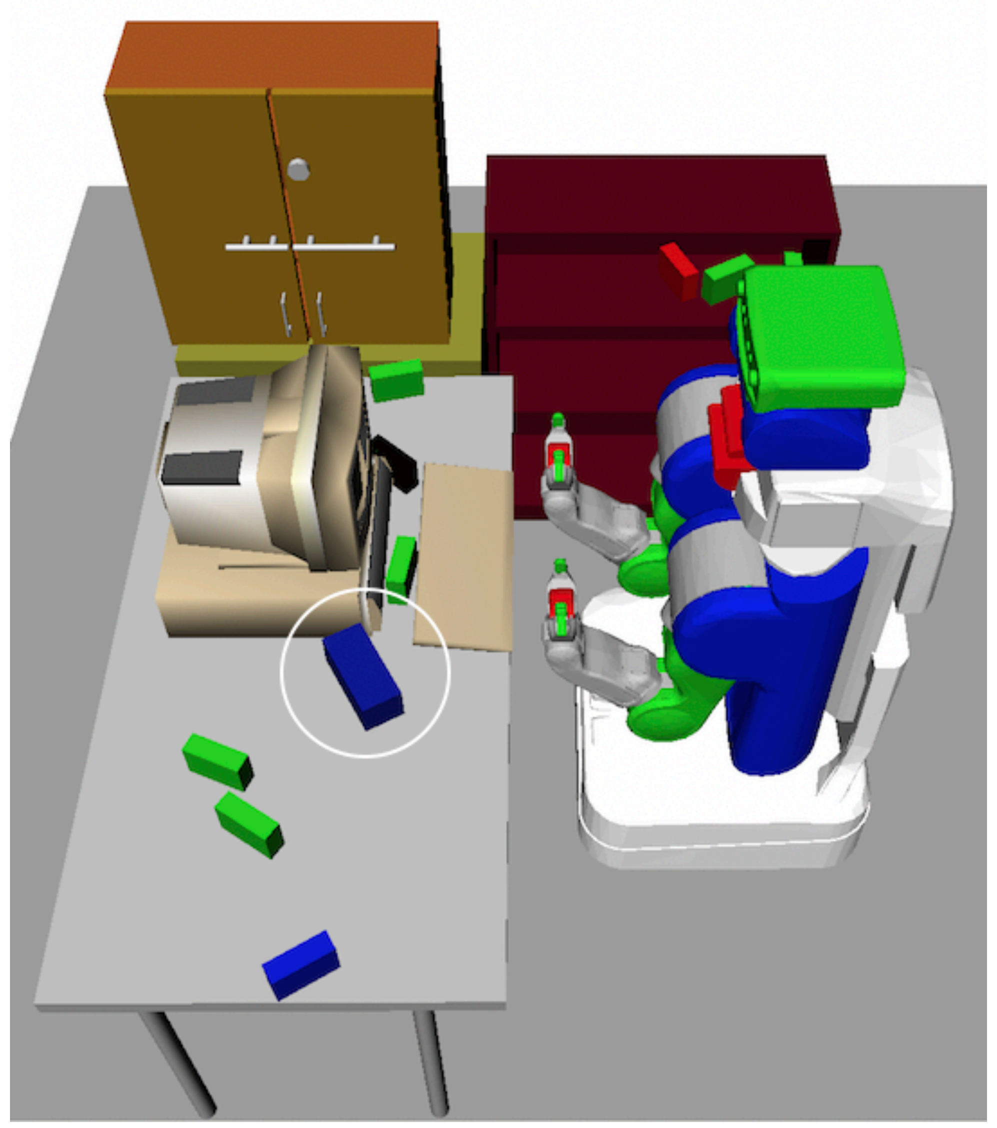
- f is expensive to evaluate
- f is multi-peak
- no gradient information
- evaluations can be noisy

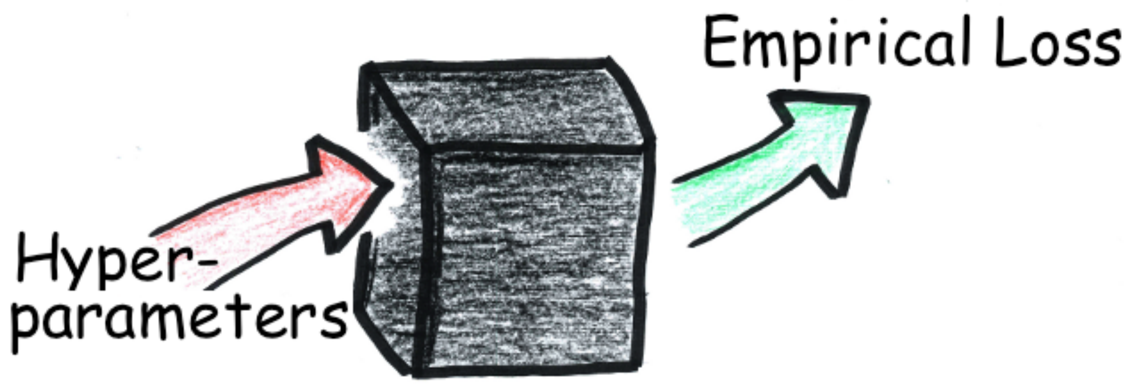
Assume a GP prior $f \sim GP(\mu, k)$

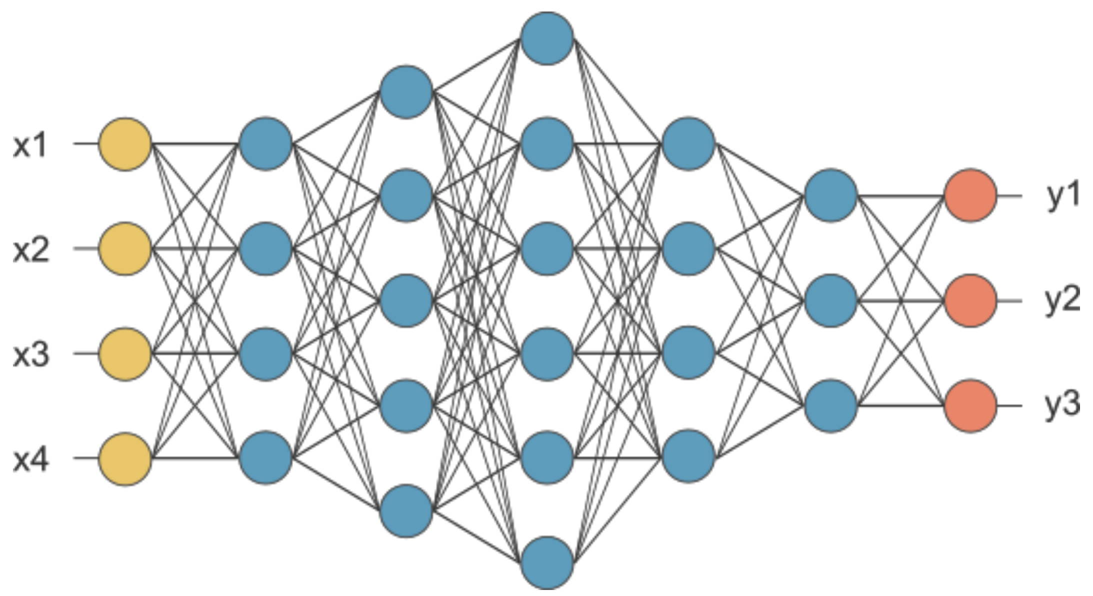
LOOP

- choose new query point(s) to evaluate
- compute the posterior GP model
- *re-estimate the prior parameters*

  e.g. by maximizing marginal data likelihood every few iterations





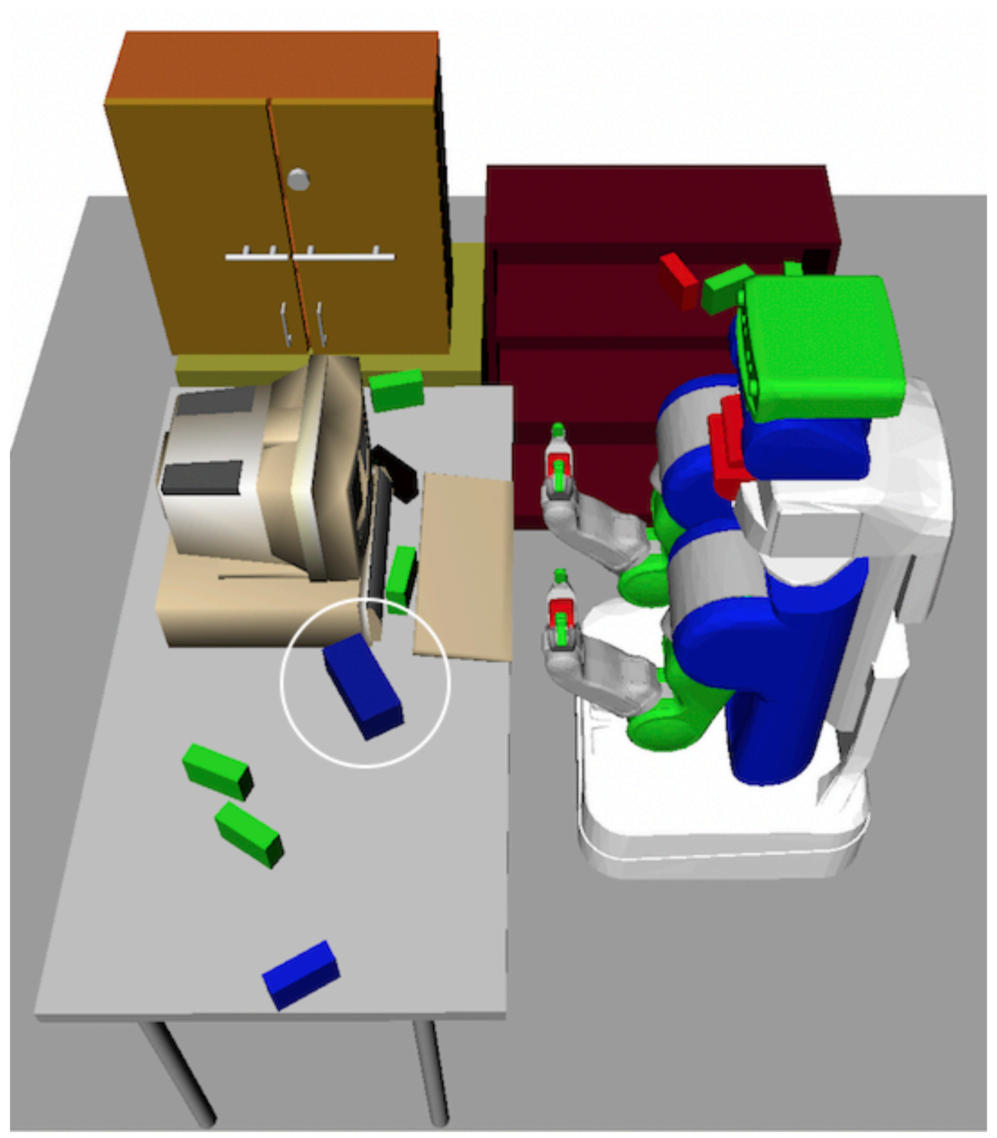


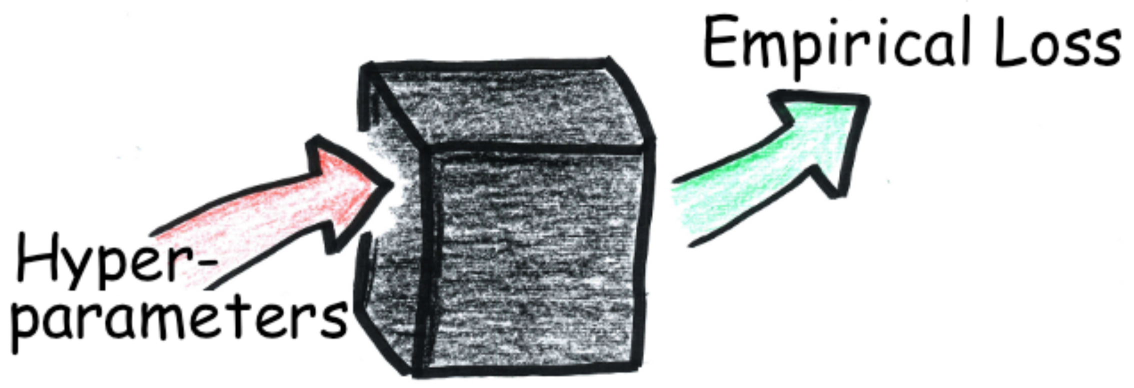
Bayesian optimization

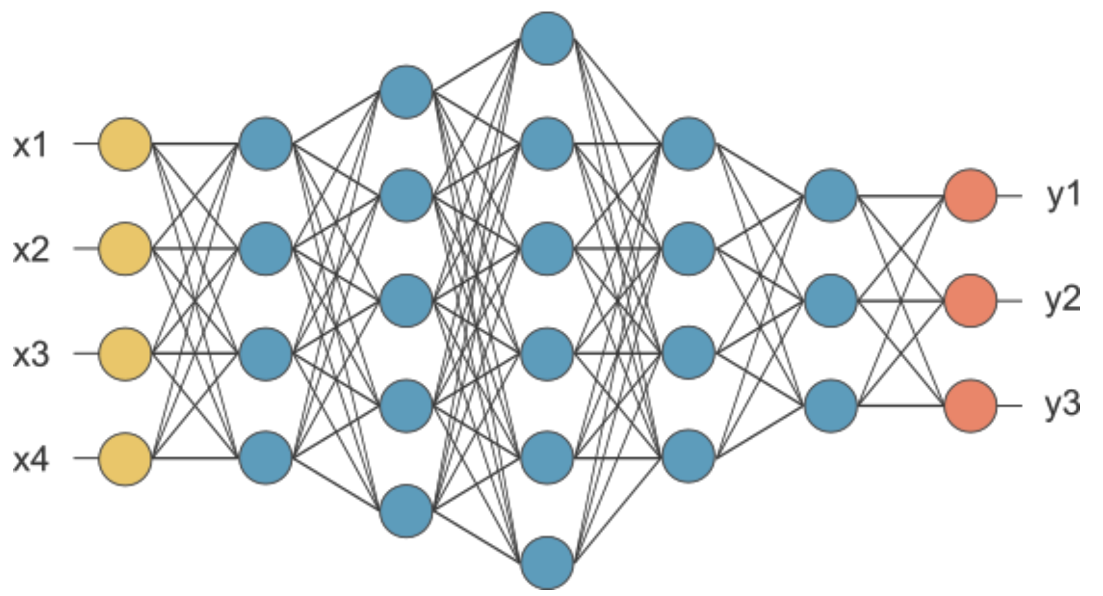
Goal: $x^* = \underset{x \in \mathfrak{X}}{\operatorname{argmax}} f(x)$

Challenges:

- f is expensive to evaluate
- f is multi-peak
- no gradient information
- evaluations can be noisy







Which comes first?
Data or prior?

Assume a GP prior $f \sim GP(\mu, k)$

LOOP

- choose new query point(s) to evaluate
- compute the posterior GP model
- *re-estimate the prior parameters*

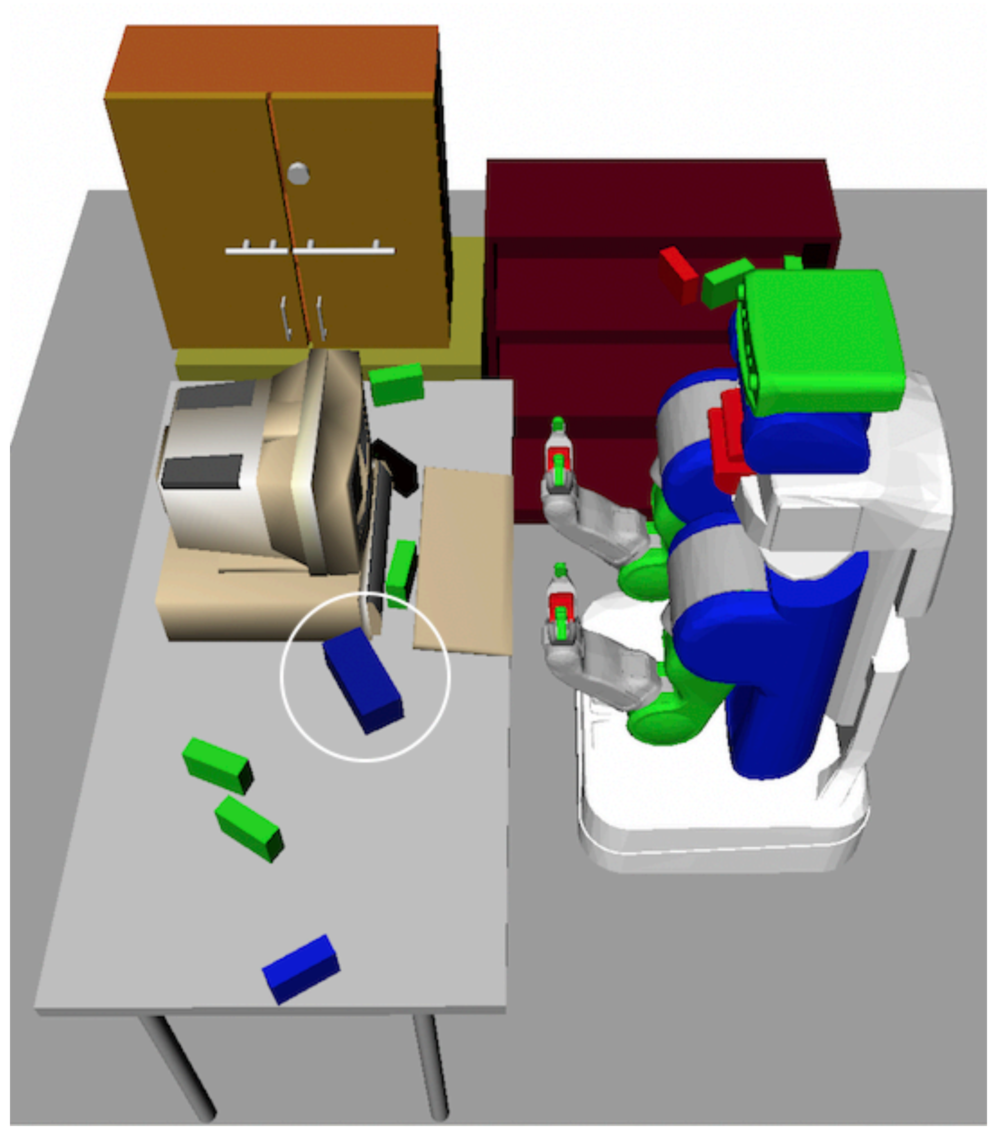
  e.g. by maximizing marginal data likelihood
  every few iterations

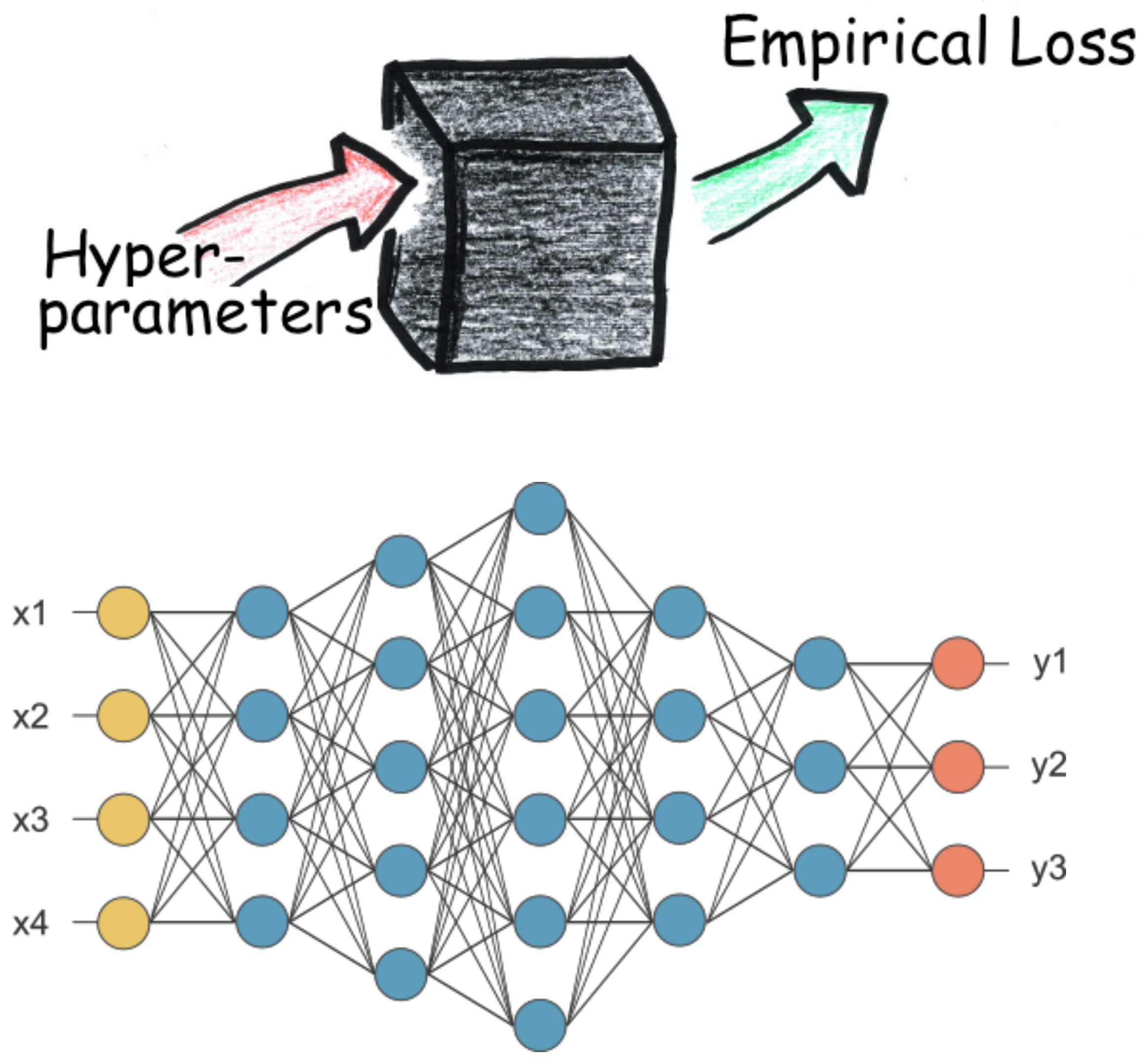
Bayesian optimization

Goal: $x^* = \underset{x \in \mathfrak{X}}{\operatorname{argmax}} f(x)$

Challenges:

- f is expensive to evaluate
- f is multi-peak
- no gradient information
- evaluations can be noisy





Which comes first?
Data or prior?

Assume a GP prior $f \sim GP(\mu, k)$

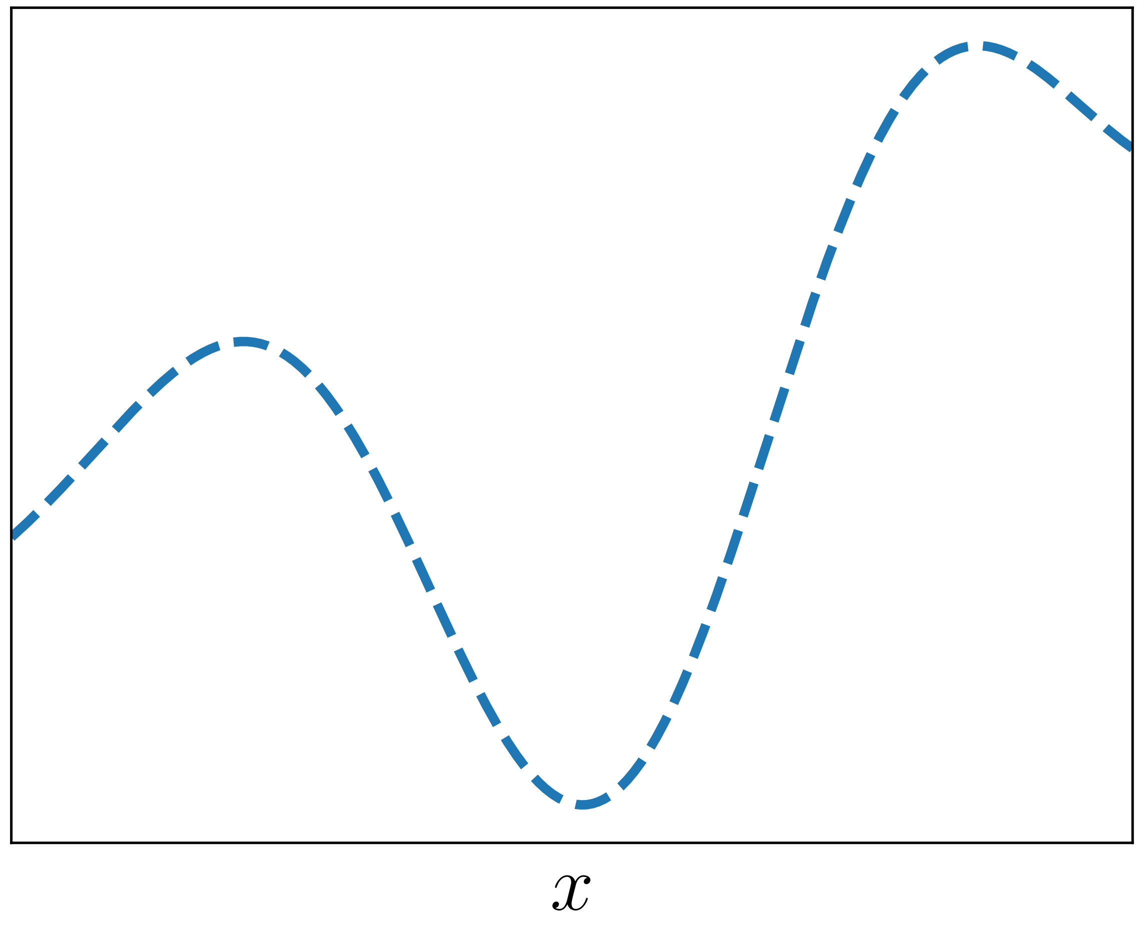
LOOP
- choose new query point(s) to evaluate
- compute the posterior GP model
- *re-estimate the prior parameters*
  e.g. by maximizing marginal data likelihood every few iterations

Hard to analyze.

Bayesian optimization with an unknown GP prior

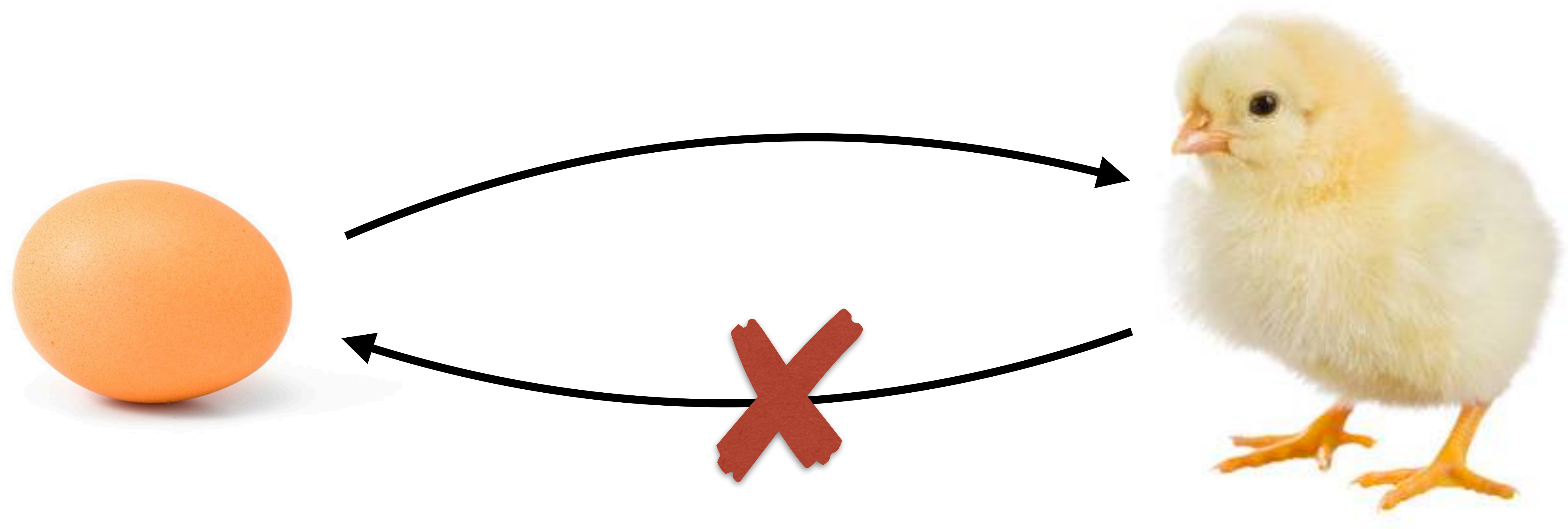
prior model

data collected on f

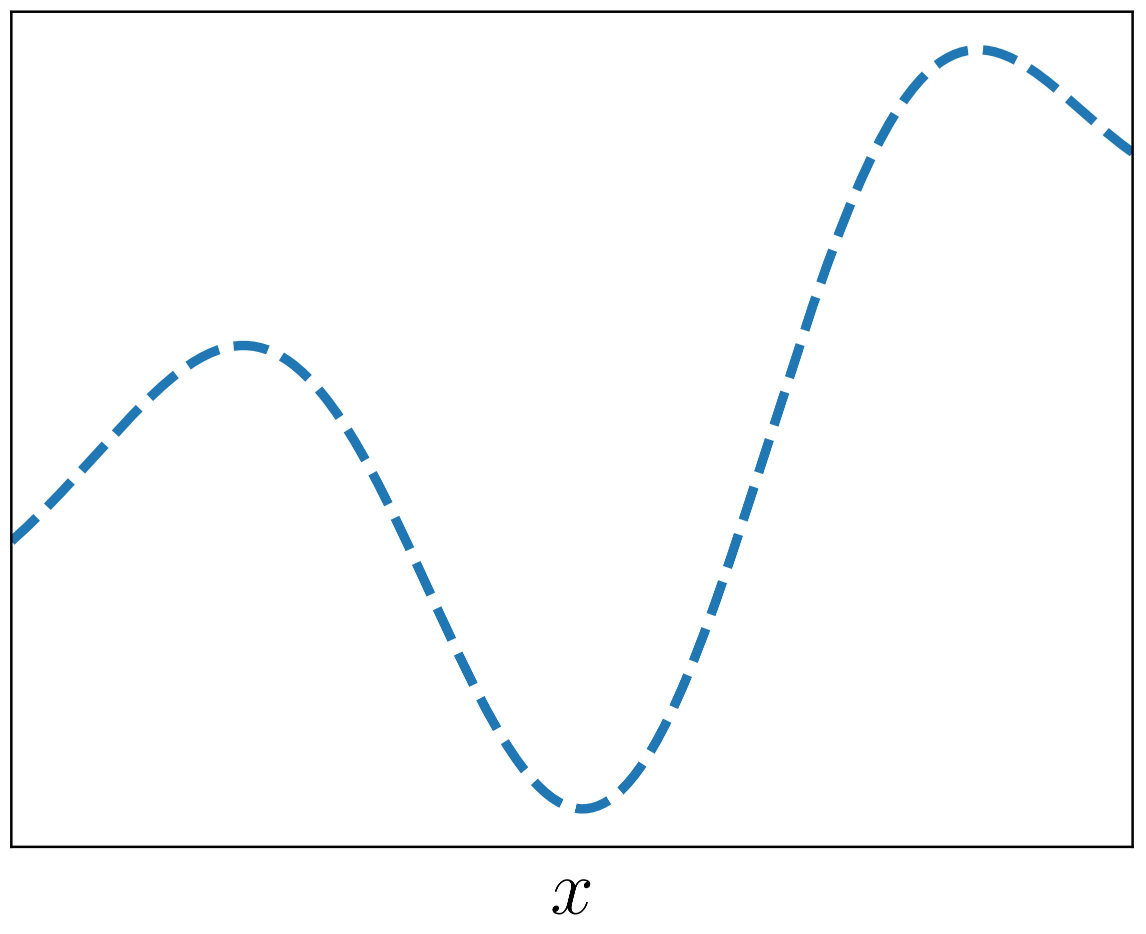


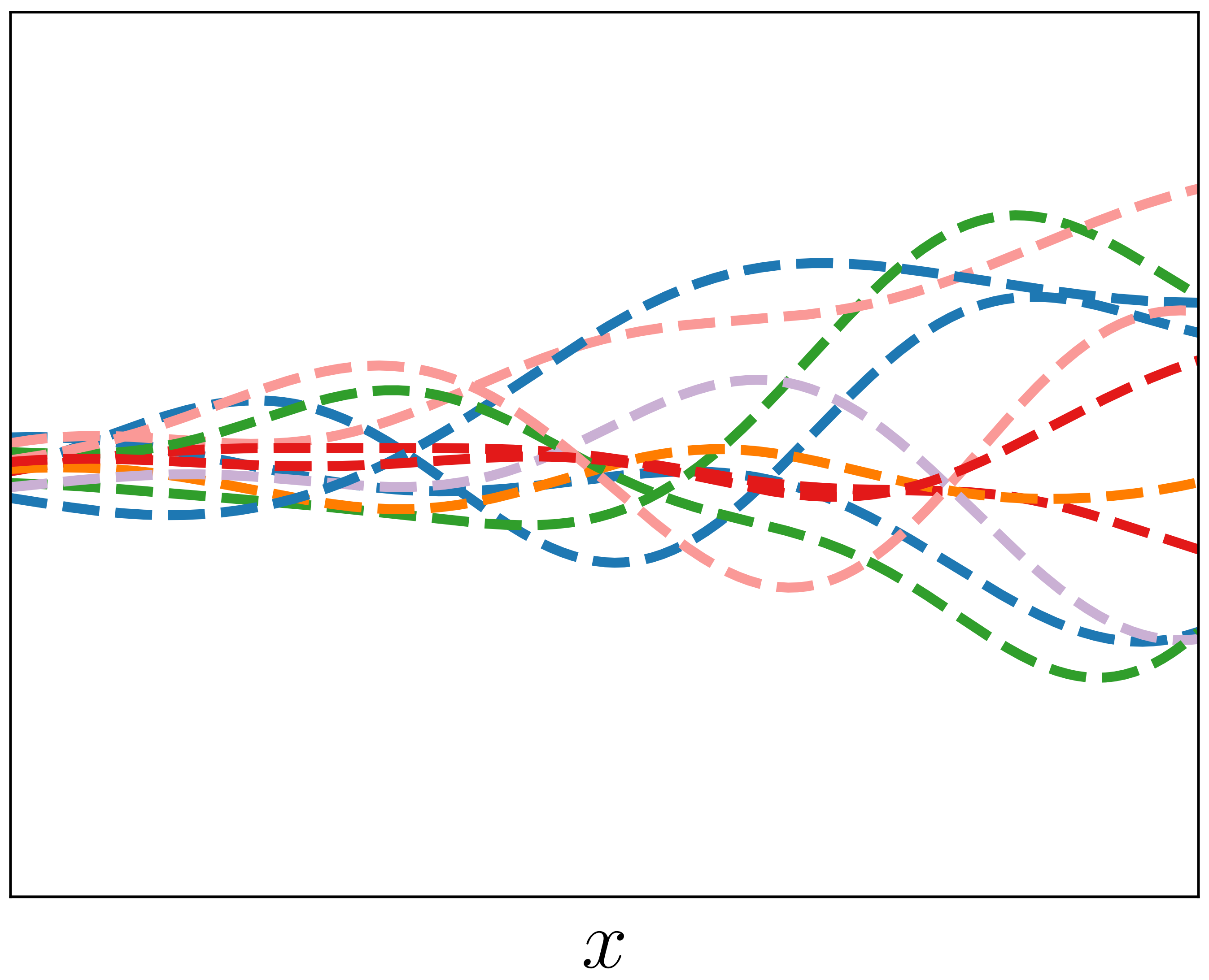
Bayesian optimization with an unknown GP prior

prior model



data collected on f

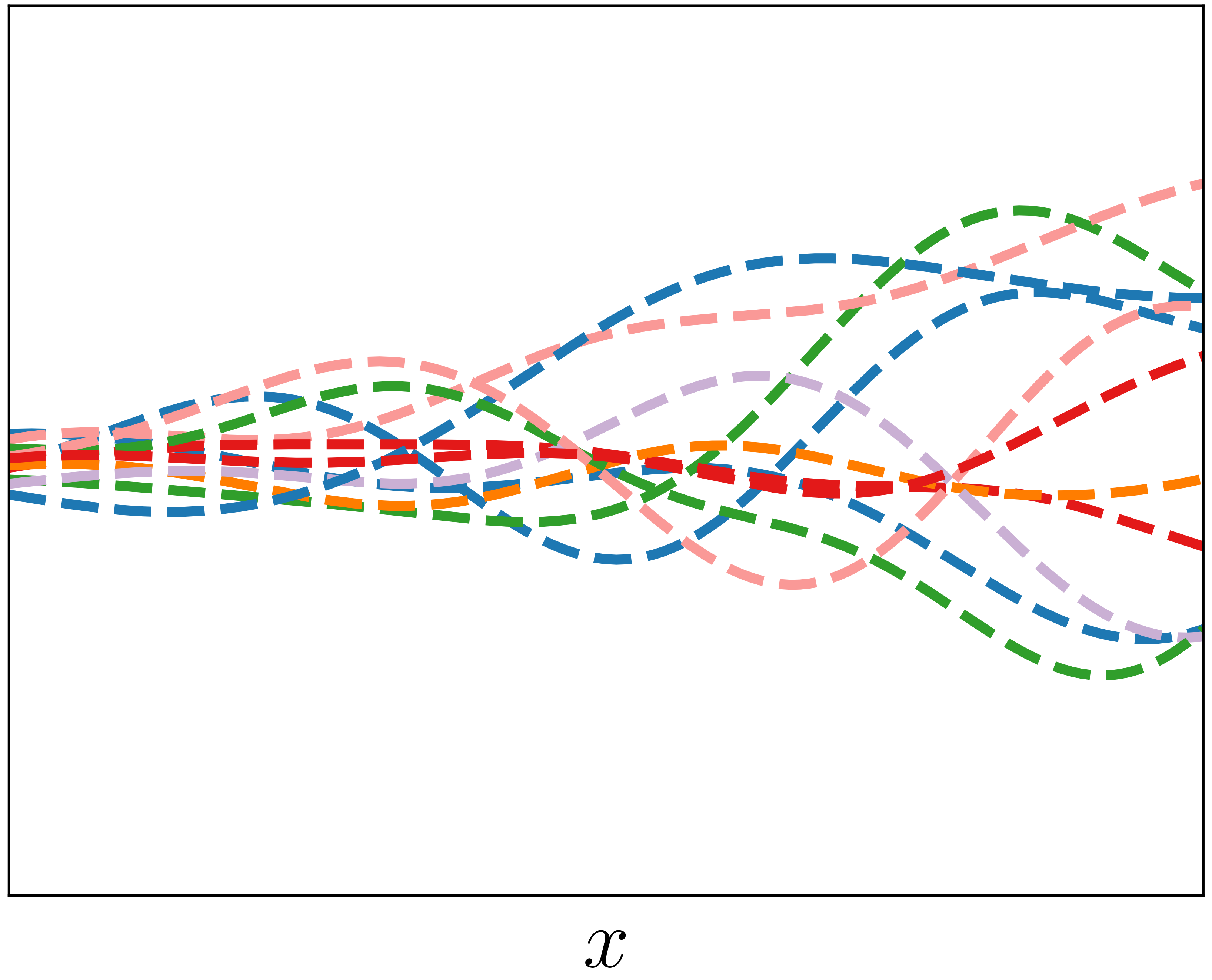




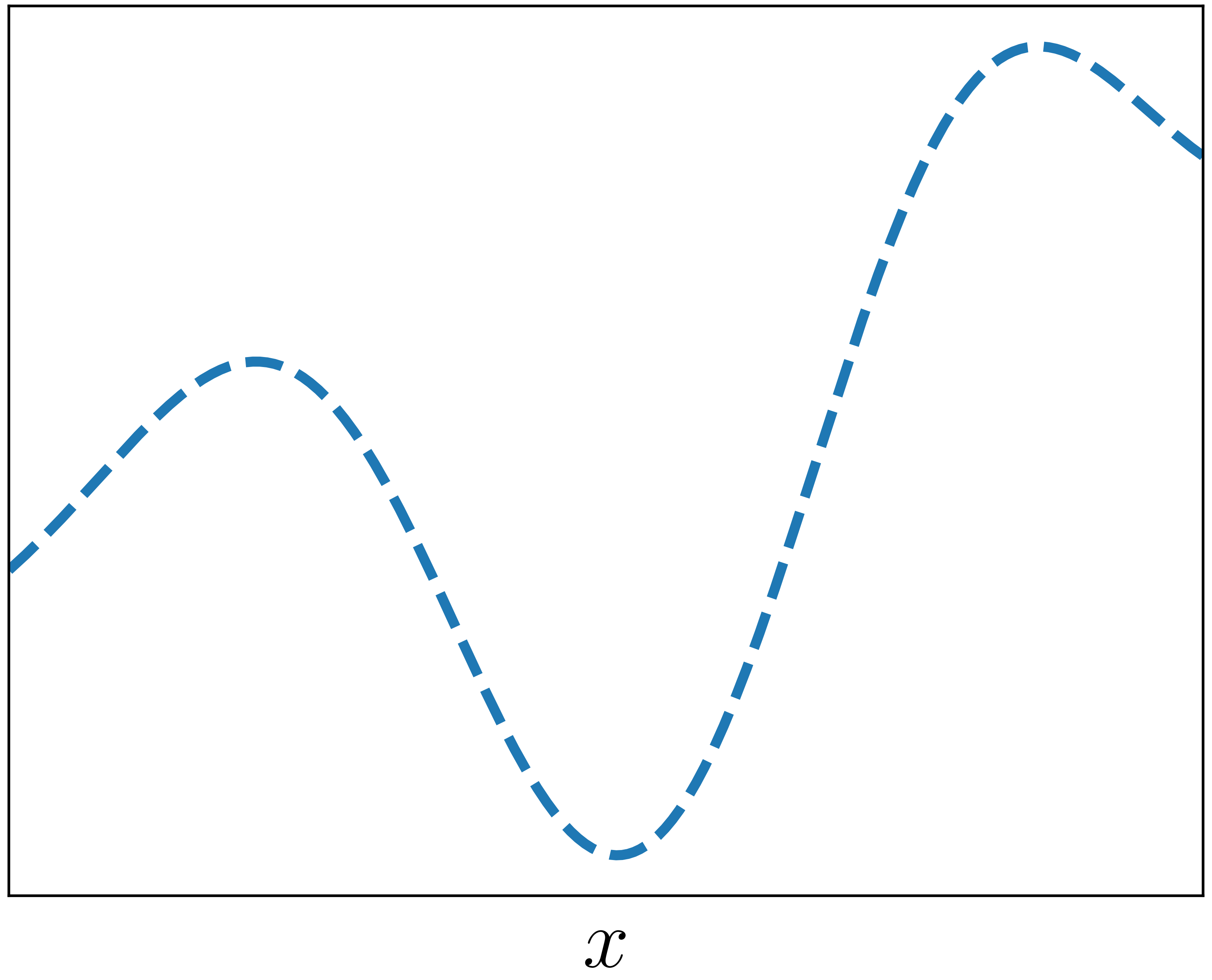
Our problem setup:
use past experience with similar functions as the meta training data to break the circular dependencies

Meta Bayesian optimization with an unknown GP prior

Offline phase

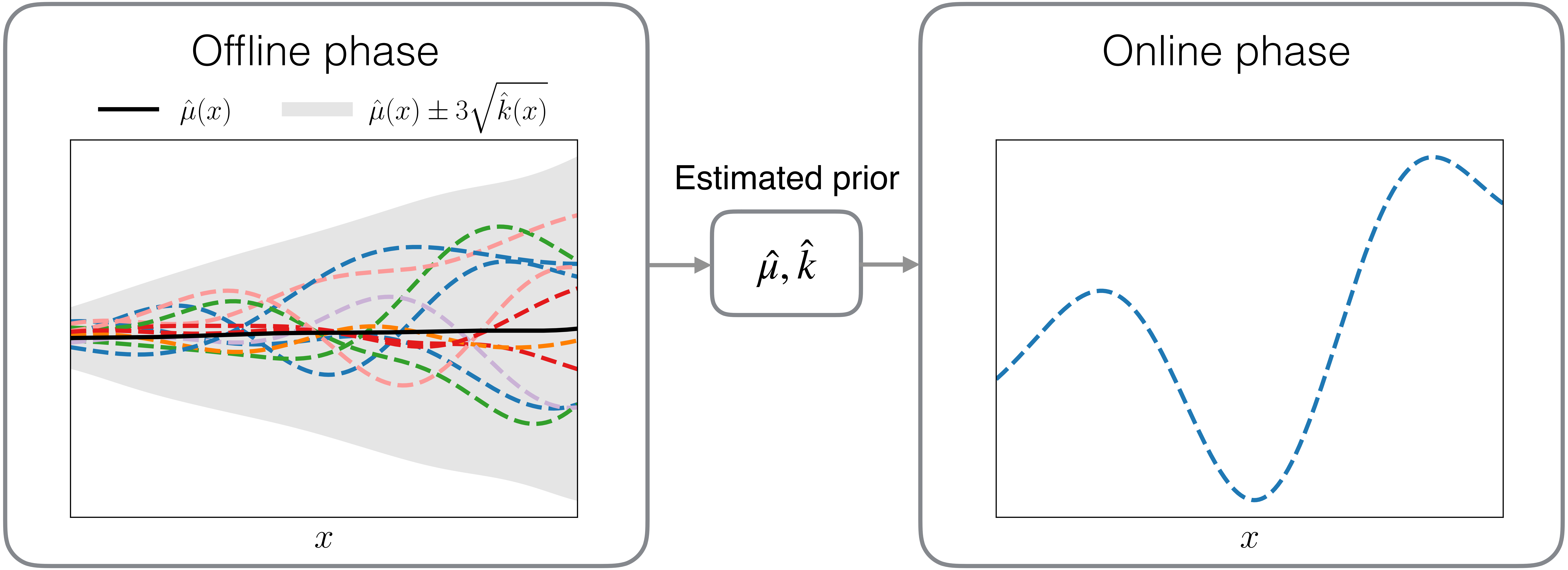


Online phase



Meta Bayesian optimization with an unknown GP prior

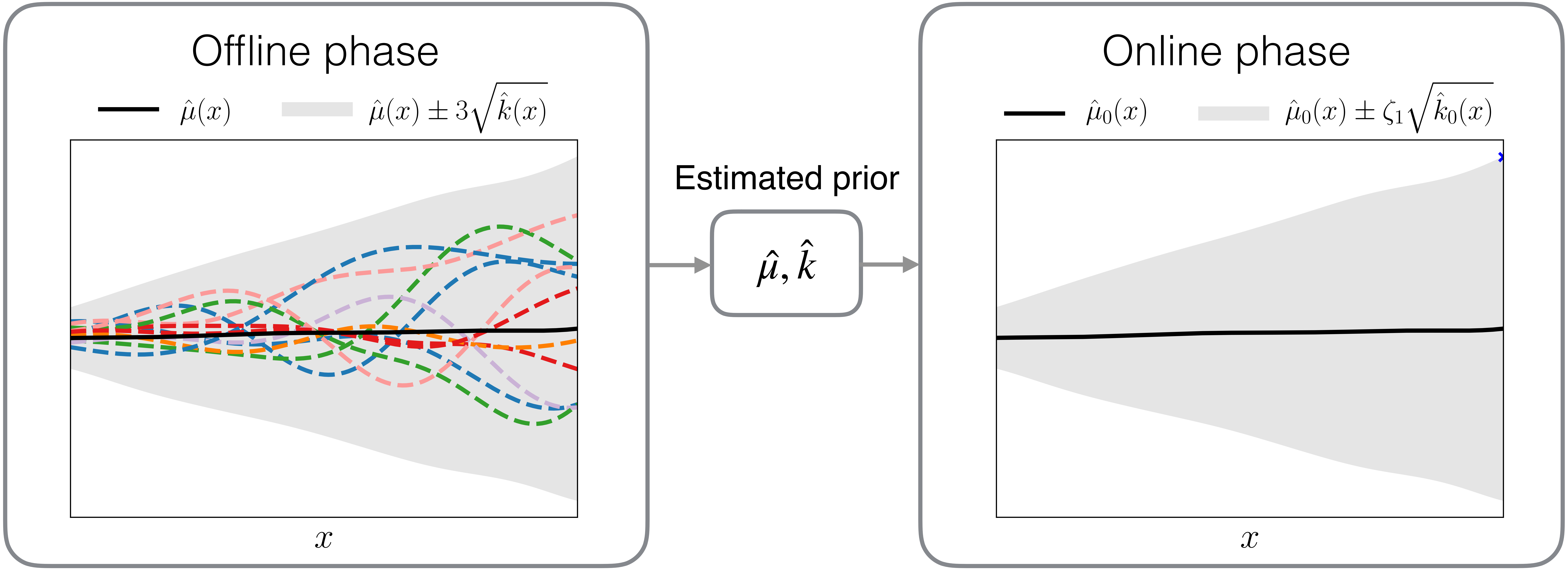
Estimate the GP prior from offline data
sampled from the same prior



Meta Bayesian optimization with an unknown GP prior

Estimate the GP prior from offline data sampled from the same prior

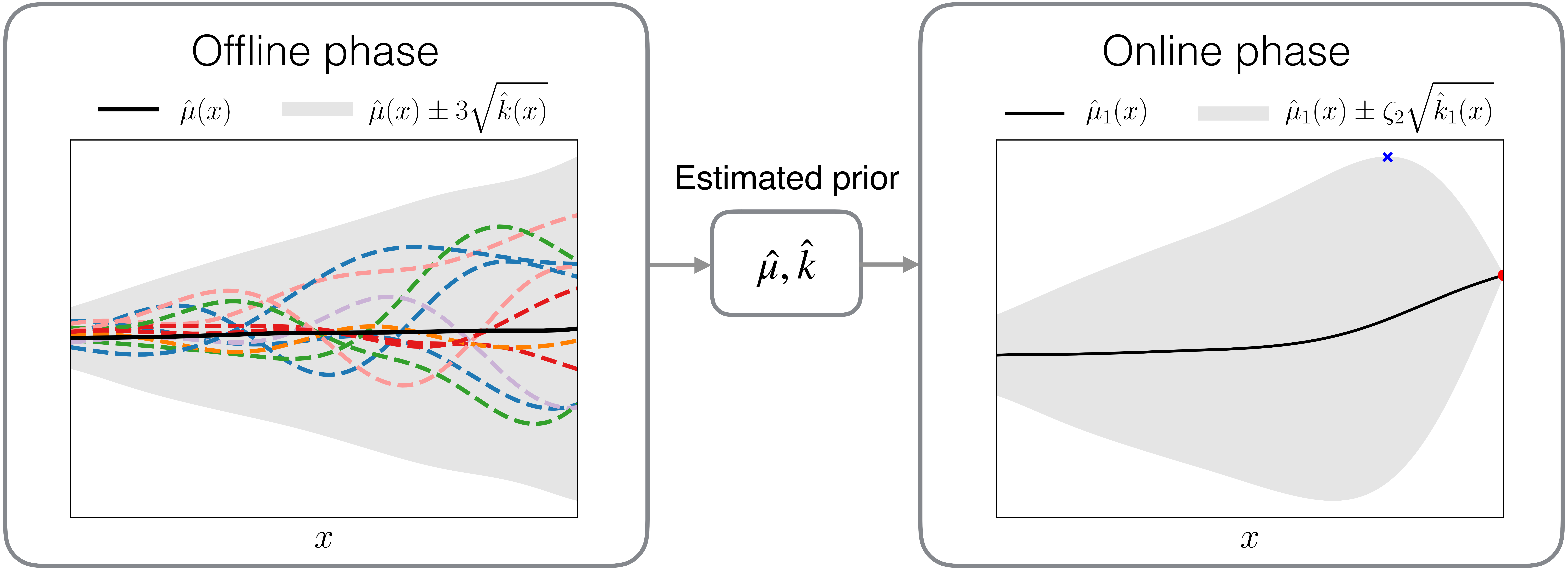
Construct unbiased estimators of the posterior and use a variant of GP-UCB



Meta Bayesian optimization with an unknown GP prior

Estimate the GP prior from offline data sampled from the same prior

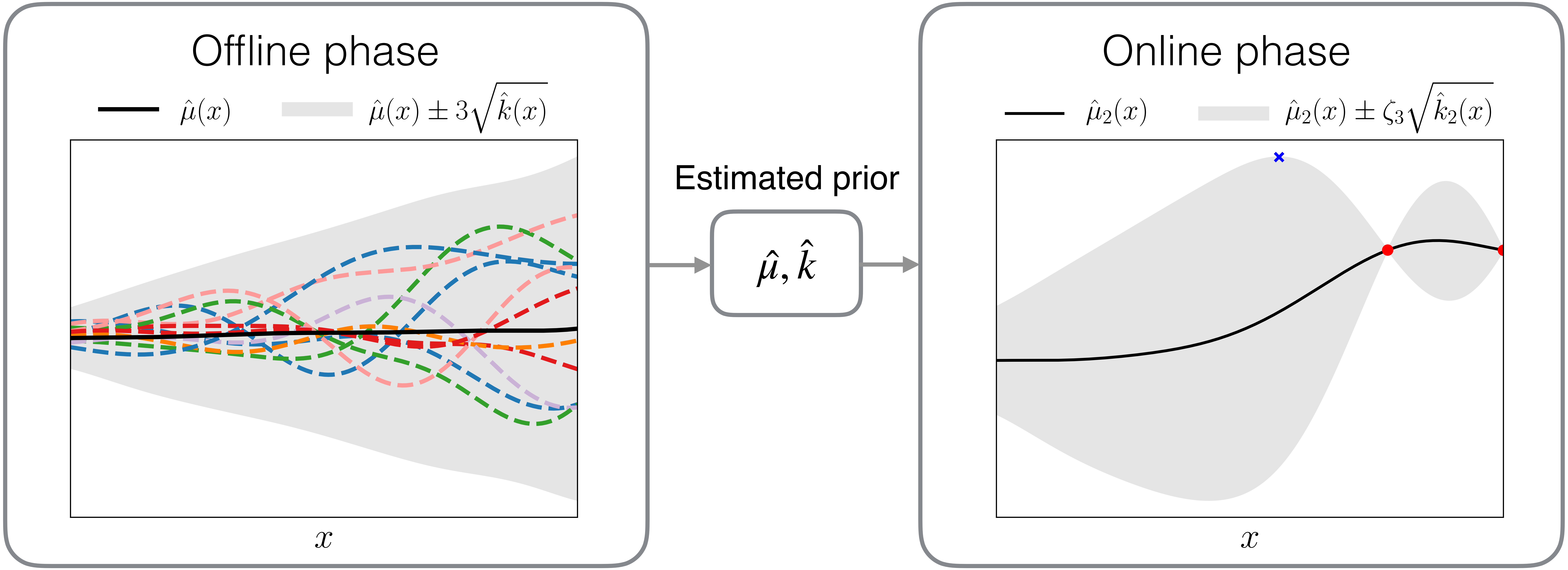
Construct unbiased estimators of the posterior and use a variant of GP-UCB



Meta Bayesian optimization with an unknown GP prior

Estimate the GP prior from offline data sampled from the same prior

Construct unbiased estimators of the posterior and use a variant of GP-UCB

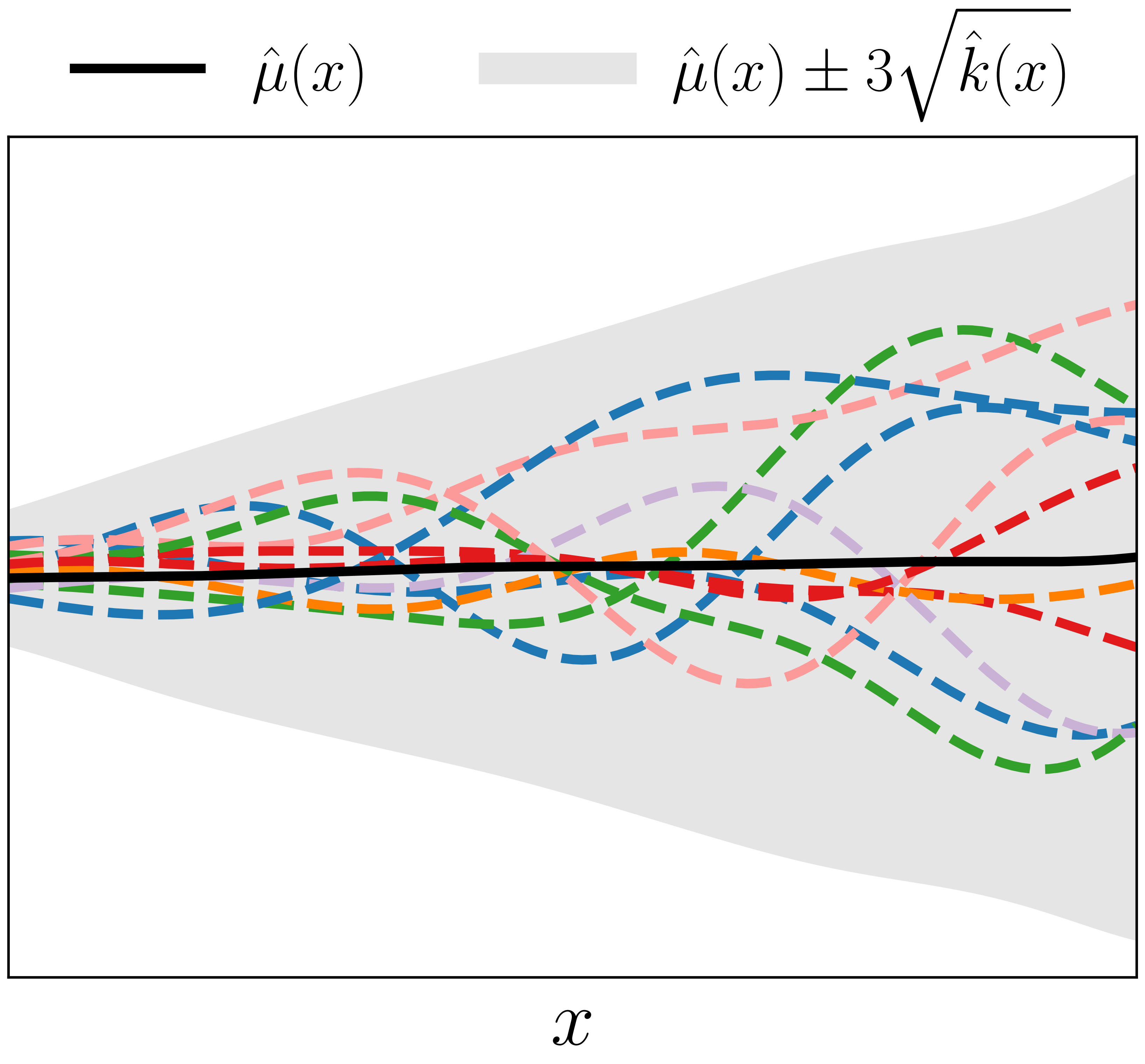


Meta Bayesian optimization with an unknown GP prior

Estimate the GP prior from offline data sampled from the same prior

Construct unbiased estimators of the posterior and use a variant of GP-UCB

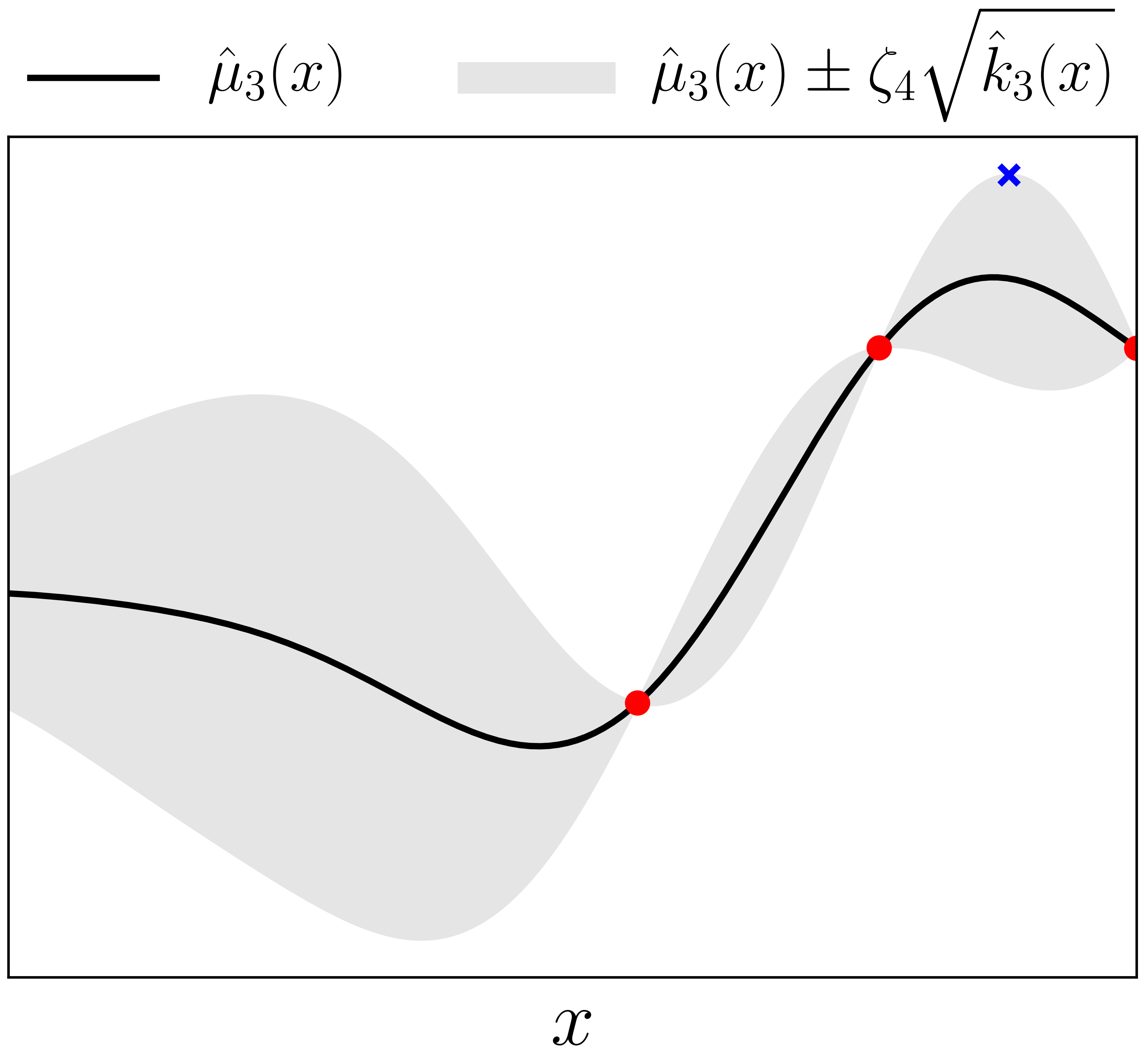
Offline phase



Estimated prior

$\hat{\mu}, \hat{k}$

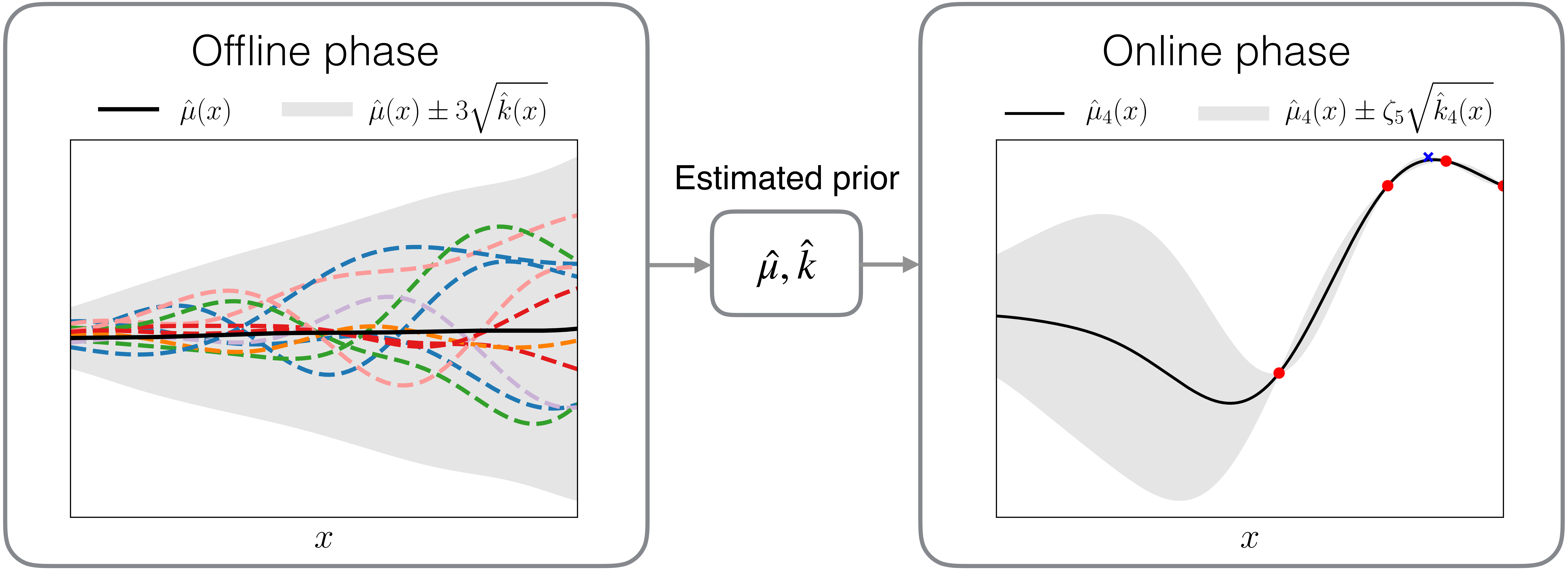
Online phase



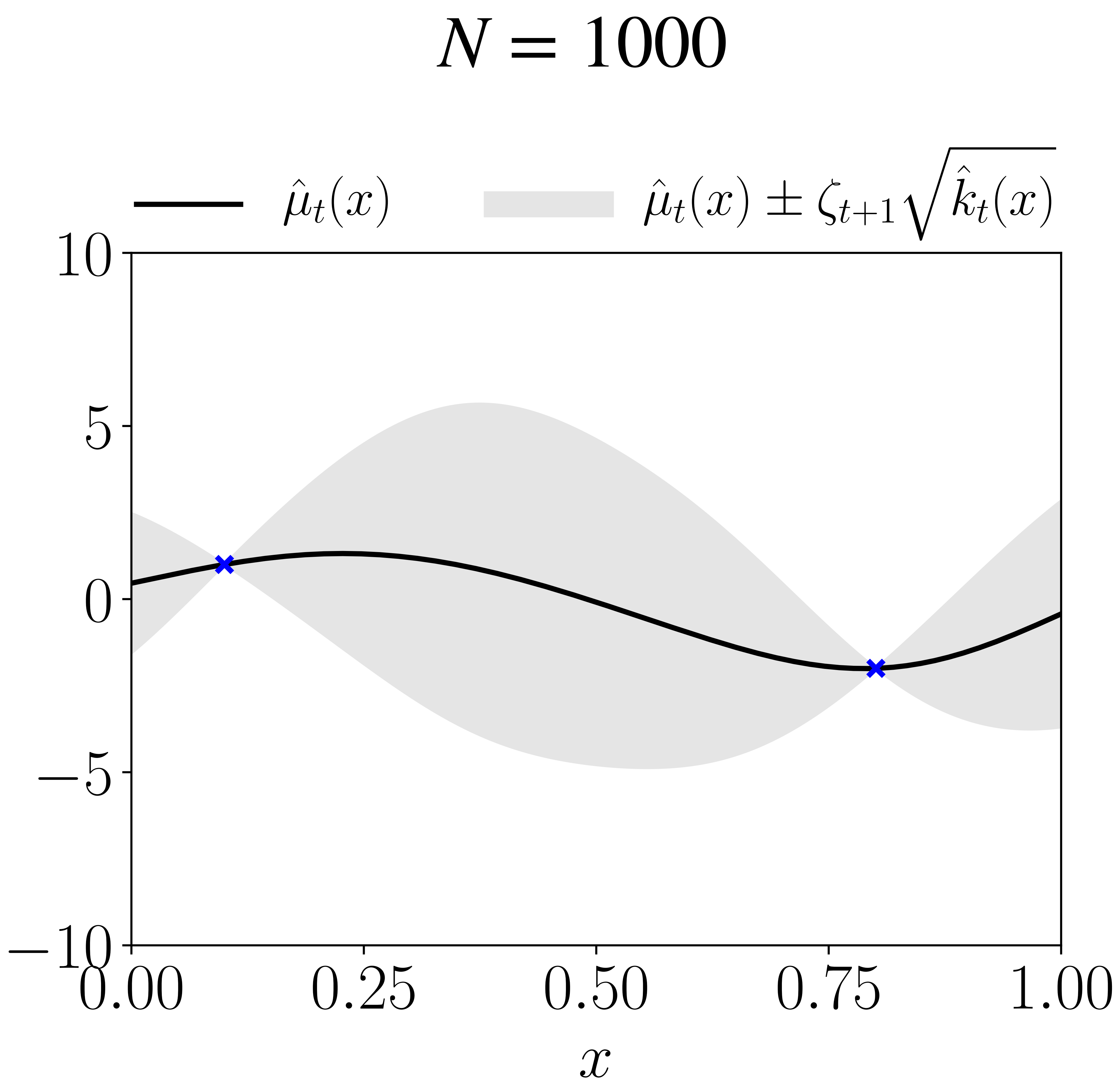
Meta Bayesian optimization with an unknown GP prior

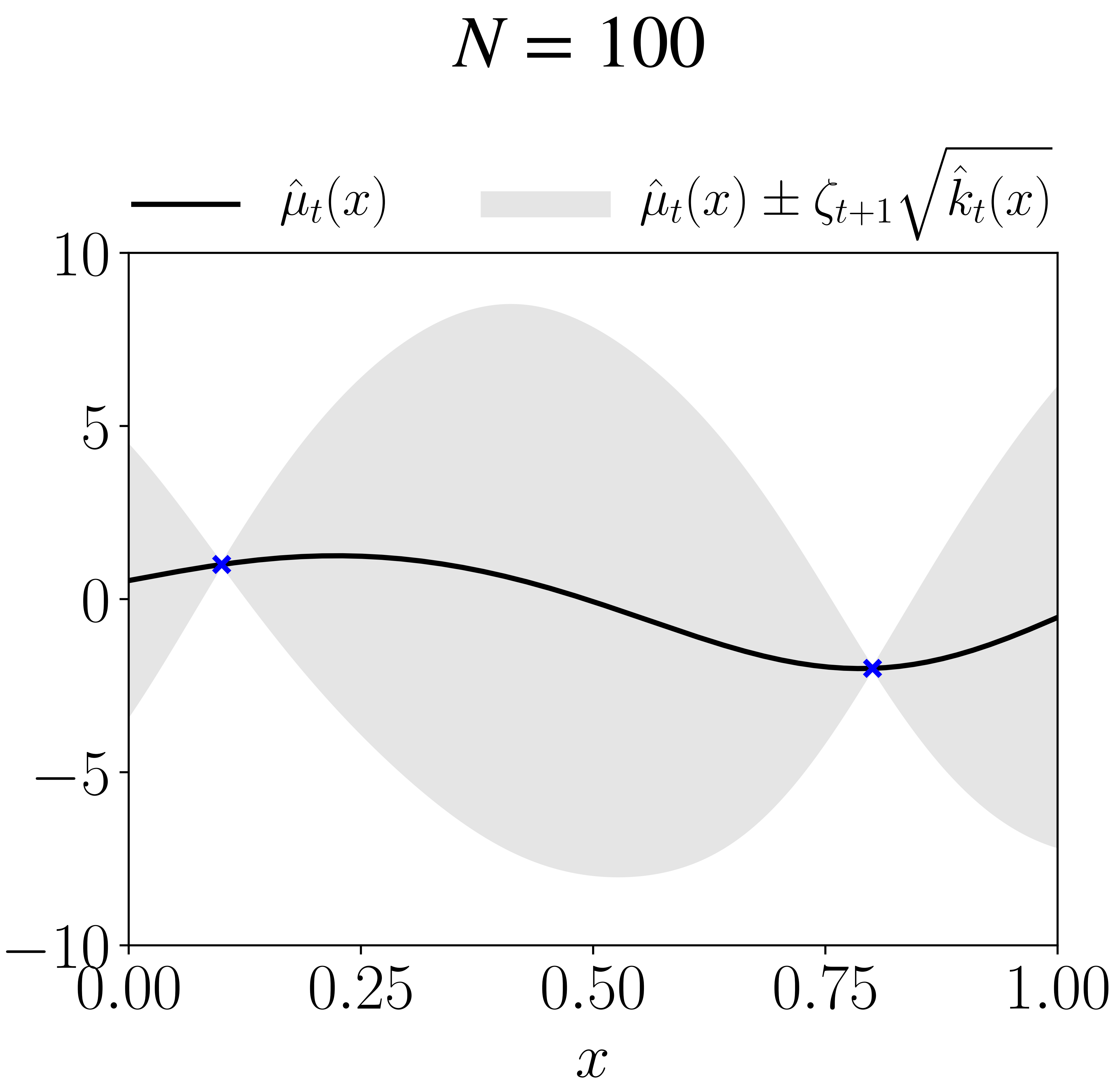
Estimate the GP prior from offline data sampled from the same prior

Construct unbiased estimators of the posterior and use a variant of GP-UCB



Effect of N, the number of meta training functions





Bounding the regret of meta BO with an unknown GP prior

Results for continuous input space @ poster #22

Theorem (finite input space)

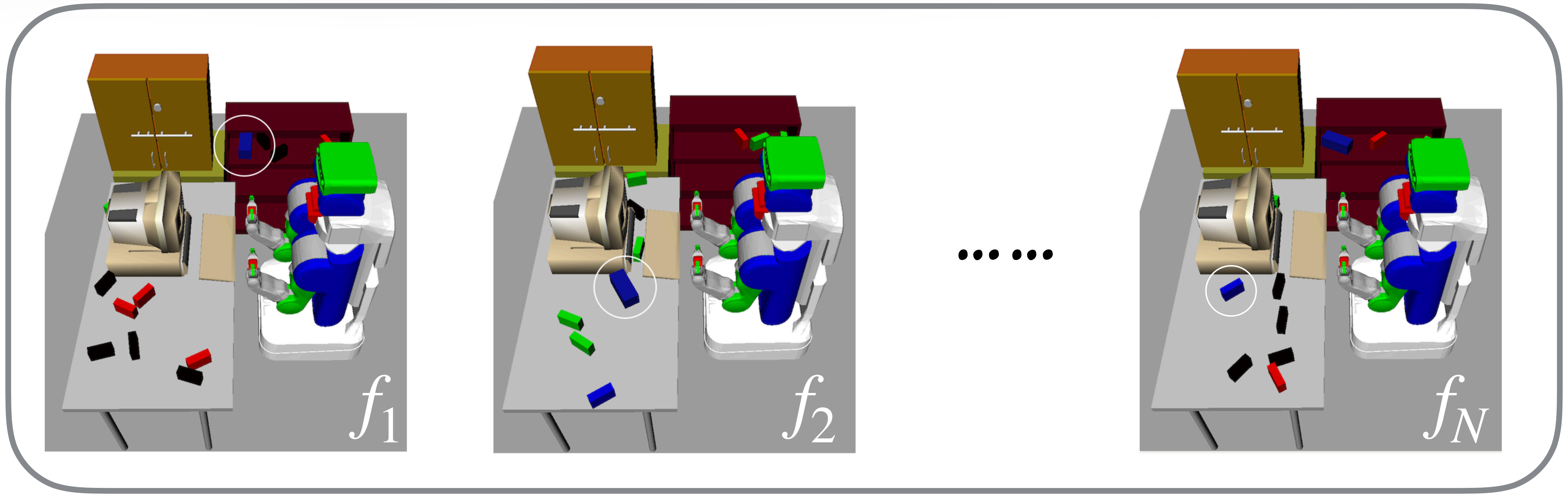
Important assumptions:

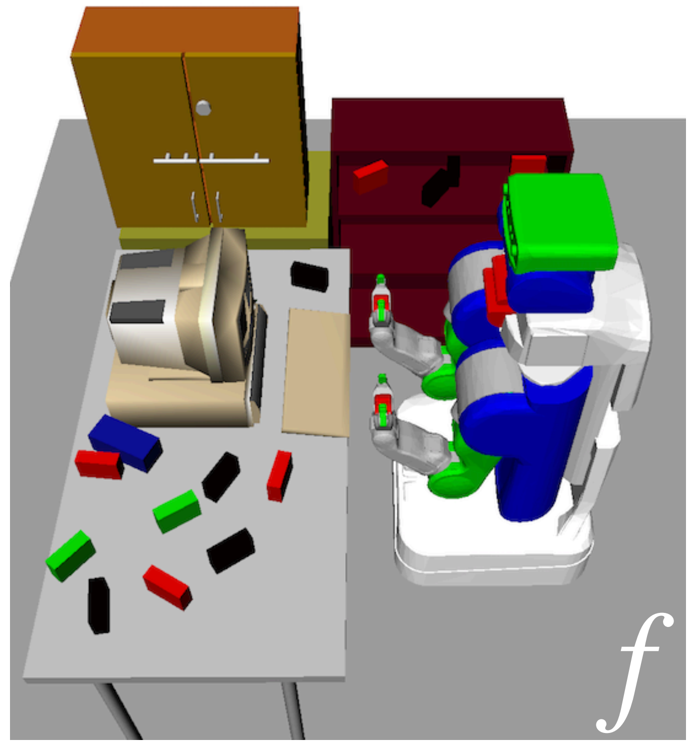
- meta-training functions come from the same prior
- enough number of meta-training functions $N \gtrsim T + 20$

Given T observations on the test function f, with high probability,

$$\text{simple regret} \quad R_T \lesssim \left(O\left(\sqrt{\frac{1}{N-T}}\right) + \underbrace{C}_{\text{constant} \approx 10} \right) \left(\sqrt{\underbrace{O\left(\frac{\log T}{T}\right)}_{\text{linear kernel}} + \underbrace{\sigma^2}_{\text{observation noise}}} \right) \to C\sigma$$

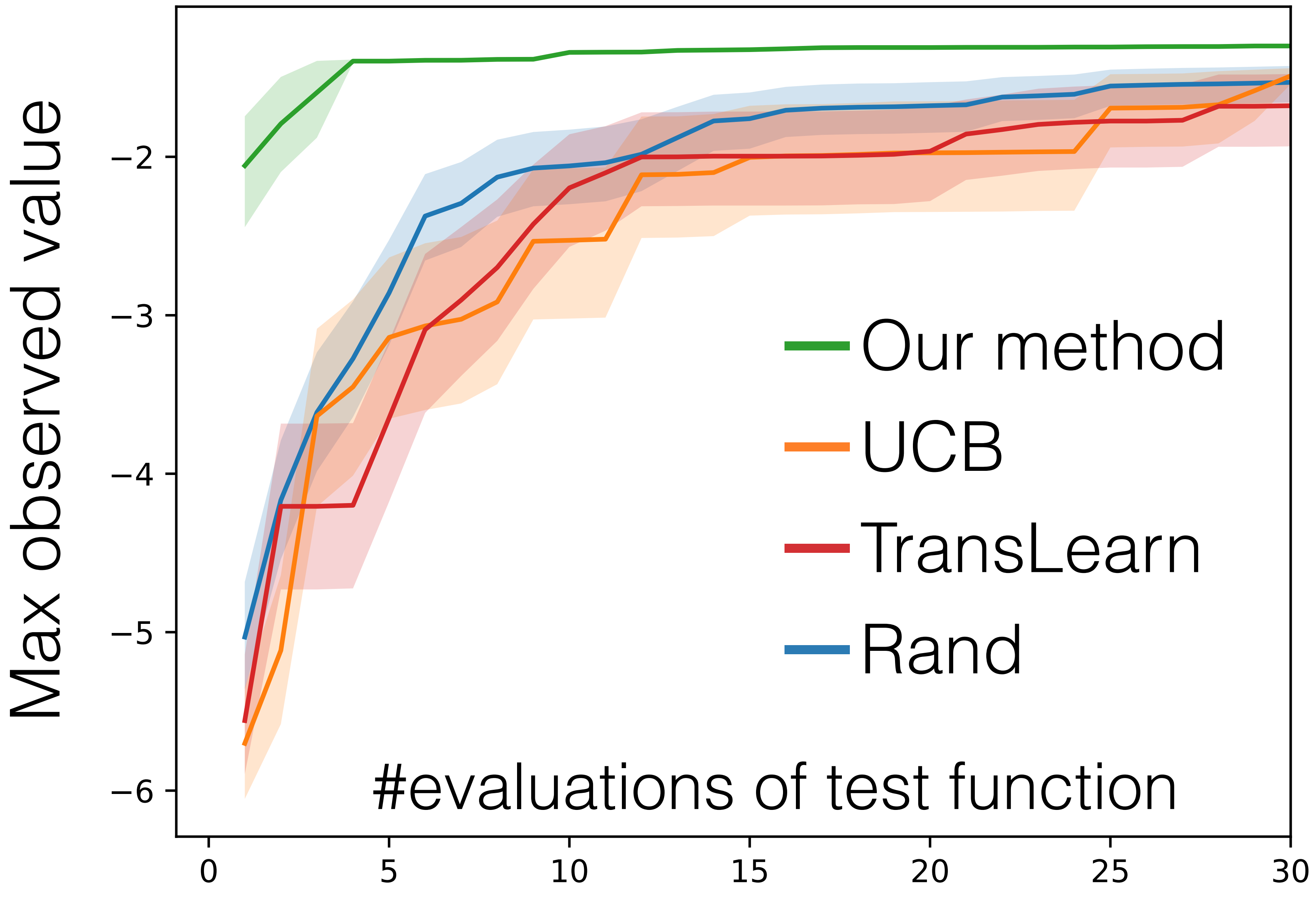
Empirical results on block picking and placing

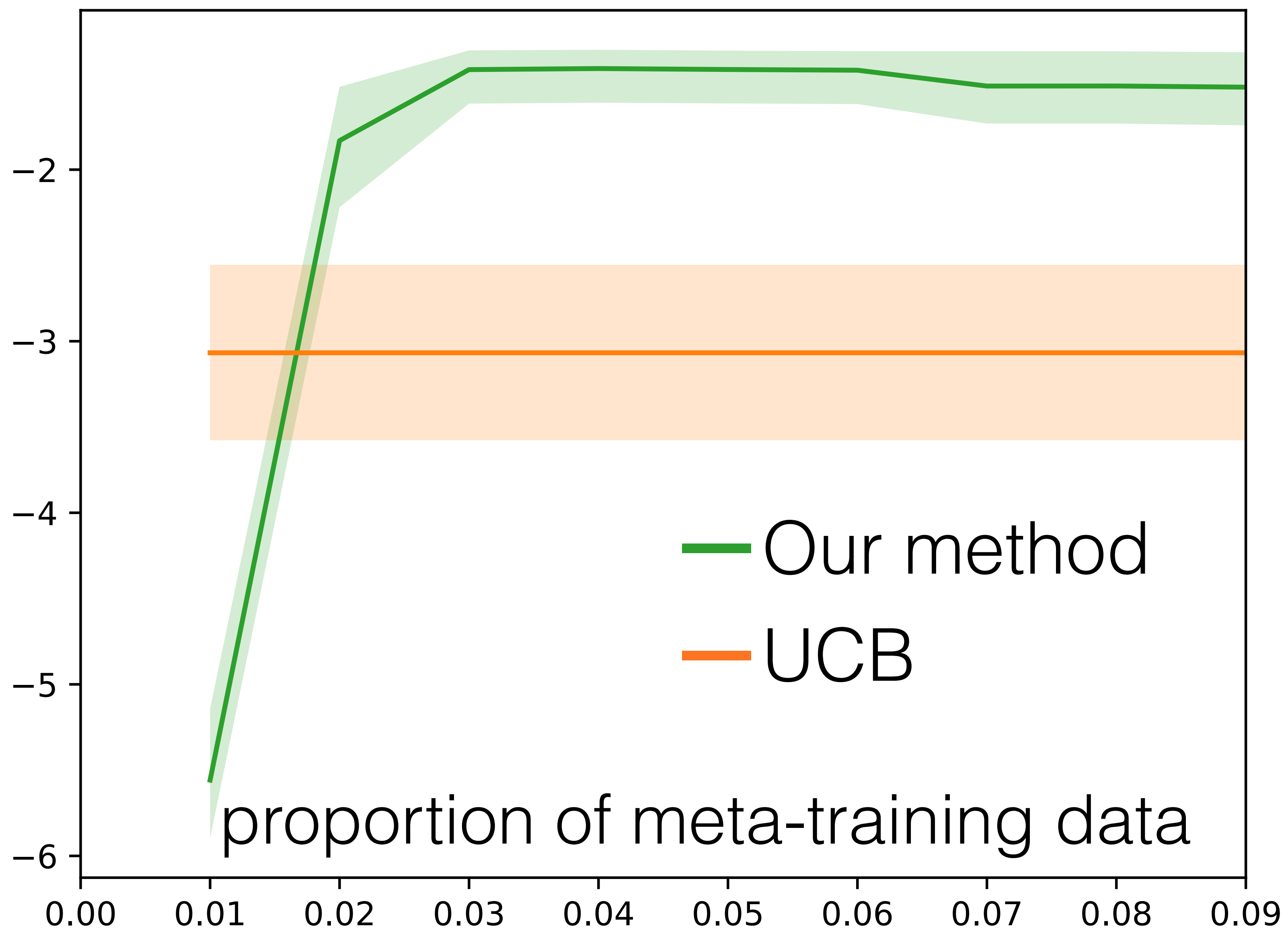




meta-training data $N = 1500$

test function





Regret bounds for meta Bayesian optimization with an **unknown** Gaussian process prior

Poster #22

More results on:

- estimation details for discrete and continuous input spaces
- regret bounds for compact input space in R^d
- regret bounds for *probability of improvement* in the meta learning setting
- empirical results on robotics tasks

https://ziw.mit.edu/meta_bo