Model-based Causal Bayesian Optimization

Rishit Dagli[1], Scott Sussex[2], Anastasia Makarova[2], Andreas Krause[2]

[1]University of Toronto [2]ETH Zürich

March 16, 2023

Table of Contents

Table of Contents

Bayesian Optimization

- Optimizing an unknown function that is expensive to evaluate (like hyperparameter tuning).

Bayesian Optimization

- Optimizing an unknown function that is expensive to evaluate (like hyperparameter tuning).
- If the function is not expensive to evaluate, just sample at many points via grid search, numeric gradient estimation, and more.

Bayesian Optimization

- Optimizing an unknown function that is expensive to evaluate (like hyperparameter tuning).
- If the function is not expensive to evaluate, just sample at many points via grid search, numeric gradient estimation, and more.
- Idea: find the global optimum in a minimum number of steps.

Bayesian Optimization

- Optimizing an unknown function that is expensive to evaluate (like hyperparameter tuning).
- If the function is not expensive to evaluate, just sample at many points via grid search, numeric gradient estimation, and more.
- Idea: find the global optimum in a minimum number of steps.
- Incorporate prior belief and update the prior with (some) samples to get a posterior that is better at approximating.

Current Methods

- Acquisition function: $x_t = \mathrm{argmax}_x \, u(x|\mathcal{D}_{1:t-1})$

Current Methods

- Acquisition function: $x_t = \text{argmax}_x \; u(x|\mathcal{D}_{1:t-1})$

Current Methods

- Acquisition function: $x_t = \text{argmax}_x \, u(x | \mathcal{D}_{1:t-1})$
- $\mathcal{D}_{1:t-1} = (\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{t-1}, y_{t-1})$, $t-1$ samples we already drew

Current Methods

- Acquisition function: $x_t = \text{argmax}_x \, u(x| \mathcal{D}_{1:t-1})$
- $\mathcal{D}_{1:t-1} = (\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{t-1}, y_{t-1})$, $t-1$ samples we already drew
- Obtain a noisy sample $y_t = f(\mathbf{x}_t) + \epsilon_t$ from objective

Current Methods

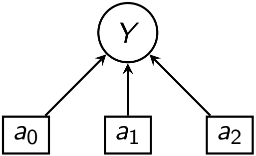
- Acquisition function: $x_t = \text{argmax}_x \, u(x| \mathcal{D}_{1:t-1})$
- $\mathcal{D}_{1:t-1} = (\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{t-1}, y_{t-1})$, $t-1$ samples we already drew
- Obtain a noisy sample $y_t = f(\mathbf{x}_t) + \epsilon_t$ from objective
- Add to previous samples and update GP, $\mathcal{D}_{1:t} = \mathcal{D}_{1:t-1}, (\mathbf{x}_t, y_t)$

Current Methods

- Acquisition function: $x_t = \text{argmax}_x\, u(x|\, \mathcal{D}_{1:t-1})$
- $\mathcal{D}_{1:t-1} = (\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{t-1}, y_{t-1})$, $t-1$ samples we already drew
- Obtain a noisy sample $y_t = f(\mathbf{x}_t) + \epsilon_t$ from objective
- Add to previous samples and update GP, $\mathcal{D}_{1:t} = \mathcal{D}_{1:t-1}, (\mathbf{x}_t, y_t)$
- Define $\text{EI}(\mathbf{x}) = \mathbb{E} \max(f(\mathbf{x}) - f(\mathbf{x}^+), 0)$ where $f(\mathbf{x}^+)$ is the best sample so far

Current Methods

- Acquisition function: $x_t = \text{argmax}_x\ u\ (x|\ \mathcal{D}_{1:t-1}\ )$
- $\mathcal{D}_{1:t-1} = (\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{t-1}, y_{t-1})$, $t-1$ samples we already drew
- Obtain a noisy sample $y_t = f(\mathbf{x}_t) + \epsilon_t$ from objective
- Add to previous samples and update GP, $\mathcal{D}_{1:t} = \mathcal{D}_{1:t-1}, (\mathbf{x}_t, y_t)$
- Define $\text{EI}(\mathbf{x}) = \mathbb{E}\max(f(\mathbf{x}) - f(\mathbf{x}^+), 0)$ where $f(\mathbf{x}^+)$ is the best sample so far
- Black Box Setup



Causal Bayesian Optimization

- Exploit structural knowledge in the form of a causal graph specified by a DAG, assuming that actions can be modeled as interventions on a structural causal model

Causal Bayesian Optimization

- Exploit structural knowledge in the form of a causal graph specified by a DAG, assuming that actions can be modeled as interventions on a structural causal model

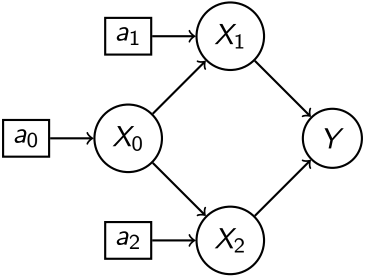


Table of Contents

Statistical Model

- $\mathcal{GP}$ for learning the RKHS($\mathcal{H}_{k_i}$) functions $(f_0, \ldots, f_m)$

Statistical Model

- $\mathcal{GP}$ for learning the RKHS($\mathcal{H}_{k_i}$) functions $(f_0, \ldots, f_m)$
- Computing posterior $\mathcal{GP}$ mean and variance:

Statistical Model

- $\mathcal{GP}$ for learning the RKHS($\mathcal{H}_{k_i}$) functions $(f_0, \ldots, f_m)$
- Computing posterior $\mathcal{GP}$ mean and variance:

$$\mu_{i,t}(z_i, a_i, l) = k_{i,t}(z_i, a_i, l)^\top (K_t + b_i^2 I)^{-1} \text{vec}(x_{i,1:t})$$

$$\begin{aligned}\sigma^2_{i,t}(z_i, a_i, l) =& k_i((z_i, a_i, l); (z_i, a_i, l)) \\ & - k_{i,t}(z_i, a_i, l)^\top (K_t + b_i^2 I)^{-1} k_{i,t}(z_i, a_i, l)\end{aligned}$$

Statistical Model

- $\mathcal{GP}$ for learning the RKHS($\mathcal{H}_{k_i}$) functions $(f_0, \ldots, f_m)$
- Computing posterior $\mathcal{GP}$ mean and variance:

$$\boxed{\mu_{i,t}(z_i, a_i, l)} = k_{i,t}(z_i, a_i, l)^\top (K_t + b_i^2 I)^{-1} \mathrm{vec}(x_{i,1:t})$$

$$\begin{aligned}\boxed{\sigma^2_{i,t}(z_i, a_i, l)} =& k_i((z_i, a_i, l); (z_i, a_i, l)) \\ & - k_{i,t}(z_i, a_i, l)^\top (K_t + b_i^2 I)^{-1} k_{i,t}(z_i, a_i, l)\end{aligned}$$

Statistical Model

- $\mathcal{GP}$ for learning the RKHS($\mathcal{H}_{k_i}$) functions $(f_0, \ldots, f_m)$
- Computing posterior $\mathcal{GP}$ mean and variance:

$$\mu_{i,t}(z_i, a_i, l) = k_{i,t}(z_i, a_i, l)^\top (K_t + b_i^2 I)^{-1} \, \mathrm{vec}(x_{i,1:t})$$

$$\begin{aligned}\sigma_{i,t}^2(z_i, a_i, l) =& k_i((z_i, a_i, l); (z_i, a_i, l)) \\ & - k_{i,t}(z_i, a_i, l)^\top (K_t + b_i^2 I)^{-1} k_{i,t}(z_i, a_i, l)\end{aligned}$$

Statistical Model

- $\mathcal{GP}$ for learning the RKHS($\mathcal{H}_{k_i}$) functions $(f_0, \ldots, f_m)$
- Computing posterior $\mathcal{GP}$ mean and variance:

$$\mu_{i,t}(z_i, a_i, l) = k_{i,t}(z_i, a_i, l)^\top (K_t + \underbrace{b_i}_{\text{variance proxy for w}}{}^2 I)^{-1} \mathrm{vec}(x_{i,1:t})$$

$$\begin{aligned}\sigma^2_{i,t}(z_i, a_i, l) =& k_i((z_i, a_i, l); (z_i, a_i, l)) \\ & - k_{i,t}(z_i, a_i, l)^\top (K_t + \underbrace{b_i}_{\text{variance proxy for w}}{}^2 I)^{-1} k_{i,t}(z_i, a_i, l)\end{aligned}$$

Statistical Model

- $\mathcal{GP}$ for learning the RKHS($\mathcal{H}_{k_i}$) functions $(f_0, \ldots, f_m)$
- Computing posterior $\mathcal{GP}$ mean and variance:

$$\mu_{i,t}(z_i, a_i, l) = \boxed{k_{i,t}}(z_i, a_i, l)^\top (K_t + b_i^2 I)^{-1} \mathrm{vec}(x_{i,1:t})$$

$$\sigma_{i,t}^2(z_i, a_i, l) = \boxed{k_i}((z_i, a_i, l); (z_i, a_i, l)) - \boxed{k_{i,t}}(z_i, a_i, l)^\top (K_t + b_i^2 I)^{-1} \boxed{k_{i,t}}(z_i, a_i, l)$$

Statistical Model

$$[K_t]_{(t_1,l),(t_2,l')} = k_i((z_{i,t_1,l}, a_{i,t_1,l}, l); (z_{i,t_2,l'}, a_{i,t_2,l'}, l'))$$

$$\begin{aligned} k_{i,t}(z_i, a_i, l)^\top =& [k_i((z_{i,1,1}, a_{i,1,1}, 1); (z_i, a_i, l)), \ldots, \\ & k_i((z_{i,t,d}, a_{i,t,d}, d); (z_i, a_i, l))]^\top \end{aligned}$$

Statistical Model

$$[K_t]_{(t_1,l),(t_2,l')} = k_i((z_{i,t_1,l}, a_{i,t_1,l}, l); (z_{i,t_2,l'}, a_{i,t_2,l'}, l'))$$

$$\begin{aligned} k_{i,t}(z_i, a_i, l)^\top =& [k_i((z_{i,1,1}, a_{i,1,1}, 1); (z_i, a_i, l)), \ldots, \\ & k_i((z_{i,t,d}, a_{i,t,d}, d); (z_i, a_i, l))]^\top \end{aligned}$$

- Single scalar-output $\mathcal{GP}$ with kernel k for modeling all output components, but introduce the component index as part of the input space

Assumptions

Assumptions on f_i

Comes with the assumption that $f_i(\cdot)$ belongs to an RKHS space of smooth functions, $\mathcal{S} = \mathcal{Z}_i \times \mathcal{A}_i$.

Assumptions

Assumptions on f_i

Comes with the assumption that $f_i(\cdot)$ belongs to an RKHS space of smooth functions, $\mathcal{S} = \mathcal{Z}_i \times \mathcal{A}_i$.

Assumptions on the Norm

Comes with the assumption that the RKHS norm of $f_i(\cdot)$ is bounded $||f_i||_{k_i} \leq \mathcal{B}_i > 0$. Also $\implies$ L_f-Lipschitz continous.

Acquisition function

Optimistically pick interventions that yield the highest expected return among all system models that are still plausible given past observations.

$$a_{:,t} = \arg\max_{a \in \mathcal{A}} \max_{\tilde{F} \in \mathcal{M}_t} \mathbb{E}_w \left[y \mid \tilde{F}, a \right]$$

Acquisition function

Optimistically pick interventions that yield the highest expected return among all system models that are still plausible given past observations.

$$a_{:,t} = \arg\max_{a \in \mathcal{A}} \max_{\tilde{F} \in \mathcal{M}_t} \mathbb{E}_w \left[y \mid \tilde{F}, a \right]$$

set of functions with bounded RKHS norm

Acquisition function

Optimistically pick interventions that yield the highest expected return among all system models that are still plausible given past observations.

$$a_{:,t} = \arg\max_{a \in \mathcal{A}} \max_{\tilde{F} \in \mathcal{M}_t} \mathbb{E}_w \left[y \mid \tilde{F}, a \right]$$

Requires reparameterization tricks

Acquisition function

Optimistically pick interventions that yield the highest expected return among all system models that are still plausible given past observations.

$$a_{:,t} = \arg\max_{a \in \mathcal{A}} \max_{\tilde{F} \in \mathcal{M}_t} \mathbb{E}_w \left[y \mid \tilde{F}, a \right]$$

Requires reparameterization tricks

$$a_{:,t} = \arg\max_{a \in \mathcal{A}} \max_{\eta(\cdot)} \mathbb{E}_w \left[y \mid \tilde{F}, a \right]$$

Acquisition function

Optimistically pick interventions that yield the highest expected return among all system models that are still plausible given past observations.

$$a_{:,t} = \arg\max_{a \in \mathcal{A}} \max_{\tilde{F} \in \mathcal{M}_t} \mathbb{E}_w \left[y \mid \tilde{F}, a \right]$$

Requires reparameterization tricks

$$a_{:,t} = \arg\max_{a \in \mathcal{A}} \max_{\eta(\cdot)} \mathbb{E}_w \left[y \mid \tilde{F}, a \right]$$

choosing optimistic but plausible models given the confidence bounds

Implementing Soft Interventions

Require: Parameters $\{\beta_t\}_{t\geq 1}, \Omega$, prior means $\mu_{i,0} = 0$, kernel functions $k_{i,0}\ \forall i \in [0, \ldots, m]$

1: **for** $t = 1, 2, \ldots$ **do**

2: Construct confidence bounds

3: Select $a_t \in \arg\max_{a \in \mathcal{A}} \max_{\eta(\cdot)} \mathbb{E}[y \mid \{\tilde{f}\}, a]$

4: Observe samples $\{z_{i,t}, x_{i,t}\}_{i=0}^{m}$

5: Use $\mathcal{D}_t$ to update posterior $\{\mu_{i,t}(\cdot), \sigma_{i,t}^2(\cdot)\}_{i=0}^{m}$

6: **end for**

Hard Interventions?

- Naturally generalizes to hard interventions, perform the combinatorial optimization over the set of nodes $\mathcal{I}$

Hard Interventions?

- Naturally generalizes to hard interventions, perform the combinatorial optimization over the set of nodes $\mathcal{I}$
- $|\mathcal{I}|$ being large is a problem but for many such use cases this is not the case

Hard Interventions?

- Naturally generalizes to hard interventions, perform the combinatorial optimization over the set of nodes $\mathcal{I}$
- $|\mathcal{I}|$ being large is a problem but for many such use cases this is not the case

For practical use-cases with large $|\mathcal{I}|$

Minimal intervention set (Lee et al., 2019) to prune sets of intervention targets that contain redundant interventions.

Implementing Hard Interventions

Require: Parameters $\{\beta_t\}_{t\geq 1}, \Omega$, prior means $\mu_{i,0} = 0$, kernel functions $k_{i,0}\ \forall i \in [0, \dots, m]$

1: **for** $t = 1, 2, \dots$ **do**

2: Construct confidence bounds

3: Select $I, a_I \in \arg\max_{I,a_I} \max_{\eta} \mathbb{E}[y \mid \{\tilde{f}\}, do(X_I = a_I)]$

4: Observe samples $\{z_{i,t}, x_{i,t}\}_{i=0}^{m}$

5: Use $\mathcal{D}_t$ to update posterior $\{\mu_{i,t}(\cdot), \sigma^2_{i,t}(\cdot)\}_{i=0}^{m}$

6: **end for**

Table of Contents

Main Results

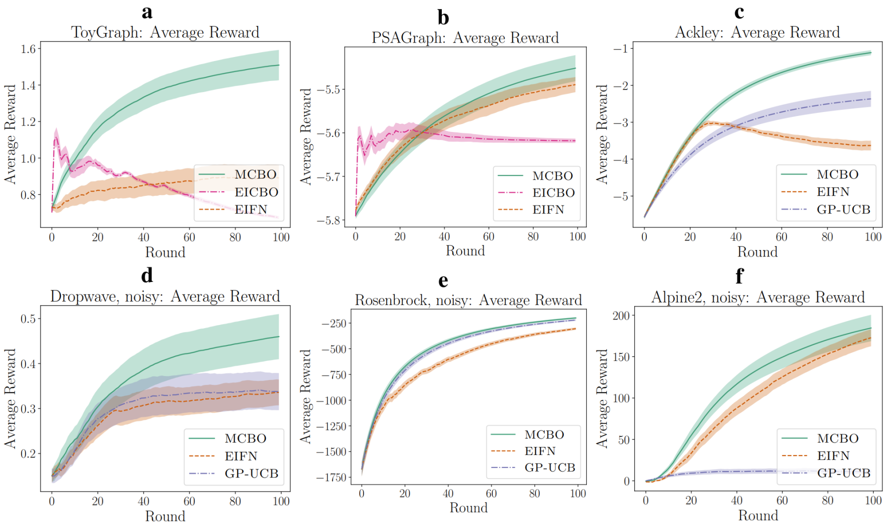


Table of Contents

Summary

- MCBO: a model-based algorithm for CBO that can be applied with very generic classes of interventions

Summary

- MCBO: a model-based algorithm for CBO that can be applied with very generic classes of interventions
- Combines models in two lines of literature, causal BayesOpt that considered "hard interventions", BO for function networks that considered "soft interventions

Summary

- MCBO: a model-based algorithm for CBO that can be applied with very generic classes of interventions
- Combines models in two lines of literature, causal BayesOpt that considered "hard interventions", BO for function networks that considered "soft interventions
- Can be efficiently implemented with popular gradient-based optimizers

Summary

- MCBO: a model-based algorithm for CBO that can be applied with very generic classes of interventions
- Combines models in two lines of literature, causal BayesOpt that considered "hard interventions", BO for function networks that considered "soft interventions
- Can be efficiently implemented with popular gradient-based optimizers
- Potentially exponential improvement in cumulative regret, with respect to the number of actions, compared to standard BO, first sublinear cumulative regret bound for CBO

Table of Contents

Next Steps

- Assumes no unobserved confounding
- More real-world applications
- Heuristics or choices for how modeling η changes performance

Thank You