Bayesian Quadrature for Parametric Expectations

Zonghao (Hudson) Chen

Department of Computer Science
University College London

Next-Generational Extrapolation Methods

Topic of this talk

Conditional Bayesian Quadrature

Recently appeared at **UAI 2024**!

Nested Expectations with Kernel Quadrature

Ongoing work

Background: Quadrature

Quantity of interest: $$I = \mathbb{E}_{X \sim \pi}[f(X)] = \int_{\mathcal{X}} f(x)\pi(x)dx$$

Background: Quadrature

Quantity of interest: $$I = \mathbb{E}_{X\sim\pi}[f(X)] = \int_{\mathcal{X}} f(x)\pi(x)dx$$

Samples

$$x_{1:N} := \left[x_1, \cdots, x_N\right]^\top \in \mathbb{R}^{N\times d_x}$$

Function evaluations

$$f(x_{1:N}) := \left[f(x_1), \cdots, f(x_N)\right]^\top \in \mathbb{R}^{N},$$

Estimator: $$I \approx \hat{I} = \sum_{i=1}^{N} w_i f(x_i)$$

How to choose weights?

Background: Quadrature

Quantity of interest: $$I = \mathbb{E}_{X \sim \pi}[f(X)] = \int_{\mathcal{X}} f(x)\pi(x)dx$$

Monte Carlo : $$\hat{I}_{MC} = \sum_{i=1}^{N} \frac{1}{N} f(x_i)$$ **Uniform weights -> Sub-optimal**

Bayesian Quadrature (BQ): $$\hat{I}_{BQ} = \sum_{i=1}^{N} w_i f(x_i)$$ **“Smart” weights**

Background: Quadrature

Bayesian Quadrature (BQ): $\hat{I}_{BQ} = \sum_{i=1}^{N} w_i f(x_i)$ **"Smart" weights**

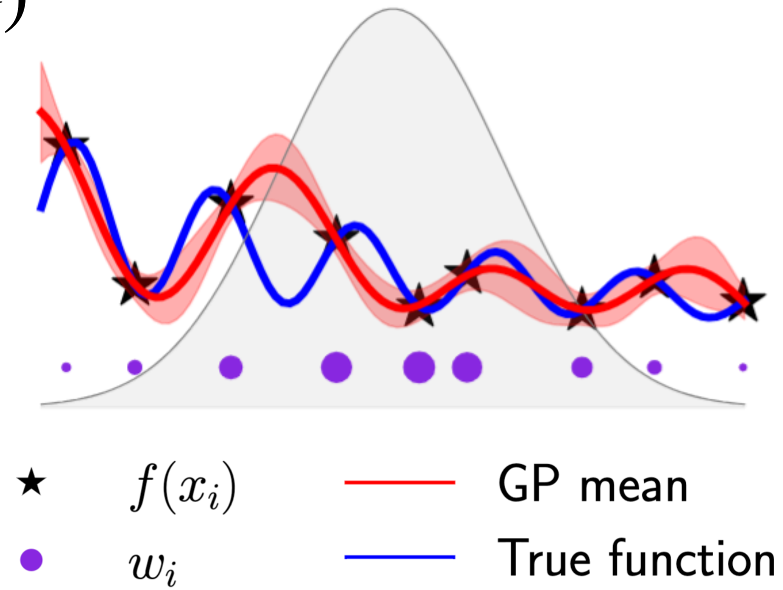
- Posit a prior $f \sim GP(0, k)$ **Smoothness**
- Conditioned on function evaluations $f(x_1), \ldots, f(x_N)$

$$f \mid f(x_1), \ldots, f(x_N) \sim GP(\bar{m}, \bar{k}),$$

$$\bar{m}(x) = [k(x, x_1), \ldots, k(x, x_N)]\mathbf{K}^{-1}[f(x_1), \ldots, f(x_N)]^\top$$

- Define $\mu(x) = \mathbb{E}_{X \sim \pi}[k(X, x)]$. The BQ weights

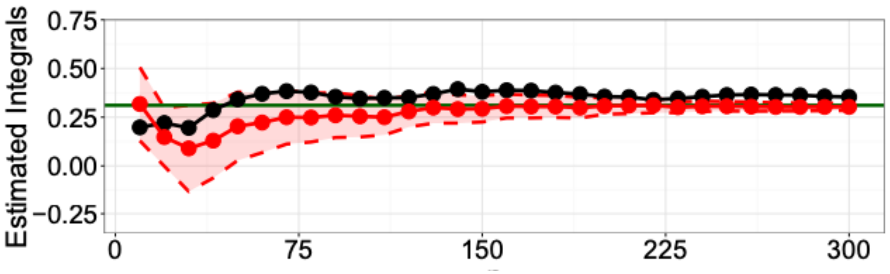
$$[w_1, \ldots, w_N] = [\mu(x_1), \ldots, \mu(x_N)]\mathbf{K}^{-1}$$



Background: Quadrature

Bayesian Quadrature (BQ): $\hat{I}_{BQ} = \sum_{i=1}^{N} w_i f(x_i)$ **"Smart" weights**

- What is good about BQ?
  - "Smarter" weights ===> Faster convergence **Smoothness**
  - Finite sample uncertainty about $\hat{I}_{BQ}$: $\hat{\sigma}^2_{BQ}$
- What is bad about BQ?
  - Inversion of Gram matrix $\mathcal{O}(N^3)$
  - Closed-form $\mu(x) = \mathbb{E}_{X\sim\pi}[k(X,x)]$ **Reparameterization "trick" (!)**

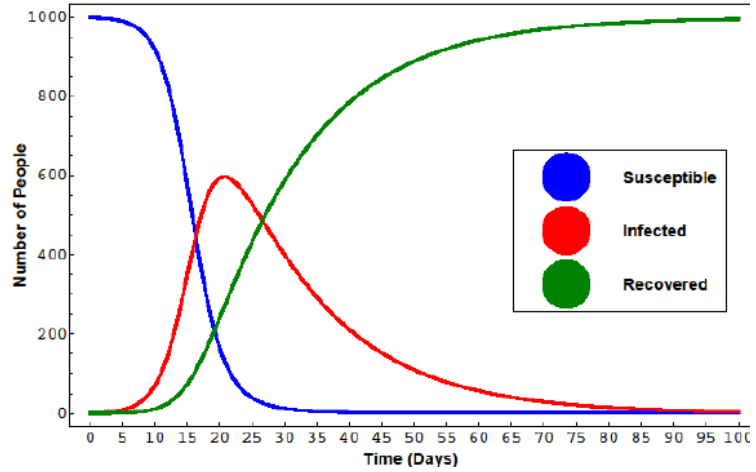


Black: Monte Carlo Red: BQ

Today: Parametric expectations

$$I(\theta) = \mathbb{E}_{X\sim\pi_\theta}[f(X,\theta)] = \int_{\mathcal{X}} f(x,\theta)\pi(x;\theta)dx$$

- Conditional Expectation: $\mathbb{E}_{X\sim\pi(X|\theta)}[f(X)]$
- Example: Susceptible-Infectious-Recovered (SIR)
  - x is the infection rate.
  - A prior belief about the distribution of x : $\pi(x;\theta)$.
  - $f(x,\theta)$ represents the peak number of infections. **Expensive!**
  - $I(\theta)$ represents the expected peak number of infections.



Given $\theta_1, \ldots, \theta_T$ would be "sufficient" for $I(\theta^*)$, provided that I is smooth enough.

The Setting

Goal: We want to approximate $I(\theta)$ over some region of the parameter space Θ:

$$I(\theta) = \mathbb{E}_{X\sim\pi_\theta}[f(X,\theta)] = \int_{\mathcal{X}} f(x,\theta)\pi(x;\theta)dx$$

Data: We have the following "data" available:

$$\theta_{1:T} := [\theta_1, \cdots, \theta_T]^\top \in \Theta^T$$

$$\forall t \in \{1,\ldots,T\}, \quad x_{1:N}^{(t)} := [x_1^{(t)}, \cdots, x_N^{(t)}]^\top \in \mathcal{X}^N$$

N samples per t

$$\forall t \in \{1,\ldots,T\}, \quad f(x_{1:N}^{(t)}, \theta_t) := [f(x_1^{(t)}, \theta_t), \cdots, f(x_N^{(t)}, \theta_t)]^\top \in \mathbb{R}^N$$

Conditional Bayesian Quadrature

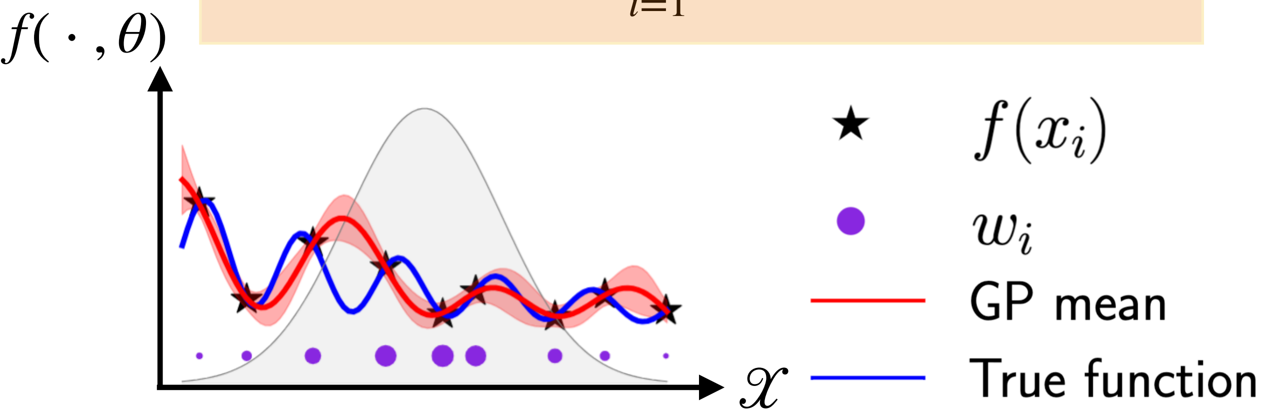
$$I(\theta) = \int_{\mathcal{X}} f(x,\theta)\pi(x;\theta)dx$$

$$x^{(t)}_{1:N} := [x^{(t)}_1, \cdots, x^{(t)}_N]^\top \in \mathcal{X}^N$$
$$\theta_{1:T} := [\theta_1, \cdots, \theta_T]^\top \in \Theta^T$$
$$f(x^{(t)}_{1:N}, \theta_t) := [f(x^{(t)}_1, \theta_t), \cdots, f(x^{(t)}_N, \theta_t)]^\top \in \mathbb{R}^N$$

Stage I: Compute T BQ posteriors:

$$\hat{I}_{\text{BQ}}(\theta_1), \sigma^2_{\text{BQ}}(\theta_1), \ldots, \hat{I}_{\text{BQ}}(\theta_T), \sigma^2_{\text{BQ}}(\theta_T),$$

$$\hat{I}_{BQ}(\theta_t) = \sum_{i=1}^{N} w_{i,t} f(x^{(t)}_i, \theta_t)$$



Conditional Bayesian Quadrature

$$I(\theta) = \int_{\mathcal{X}} f(x,\theta)\pi(x;\theta)dx$$

$$x^t_{1:N} := [x^t_1, \cdots, x^t_N]^\top \in \mathcal{X}^N$$

$$\theta_{1:T} := [\theta_1, \cdots, \theta_T]^\top \in \Theta^T$$

$$f(x^t_{1:N}, \theta_t) := [f(x^t_1, \theta_t), \cdots, f(x^t_N, \theta_t)]^\top \in \mathbb{R}^N$$

Stage I: Compute T BQ posteriors:

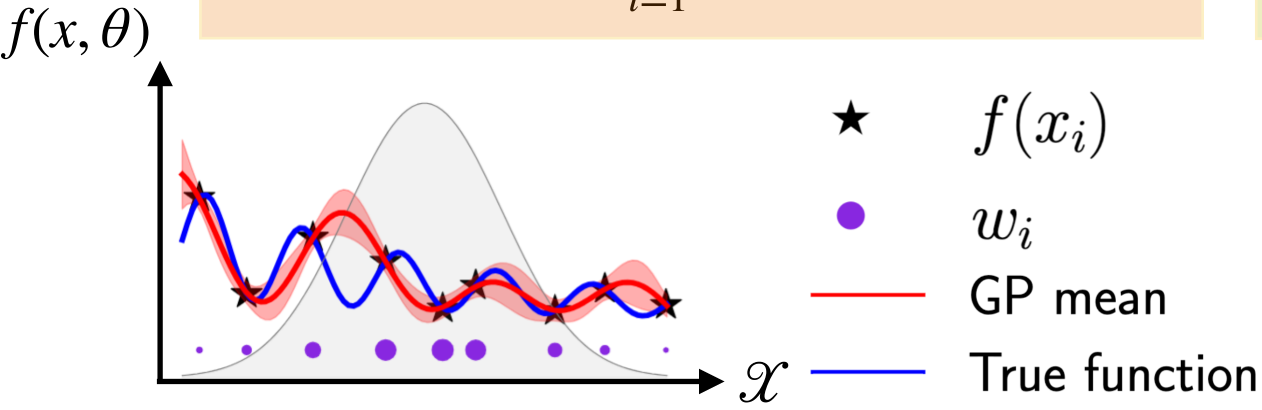
$\hat{I}_{\text{BQ}}(\theta_1), \sigma^2_{\text{BQ}}(\theta_1), \ldots, \hat{I}_{\text{BQ}}(\theta_T), \sigma^2_{\text{BQ}}(\theta_T)$,

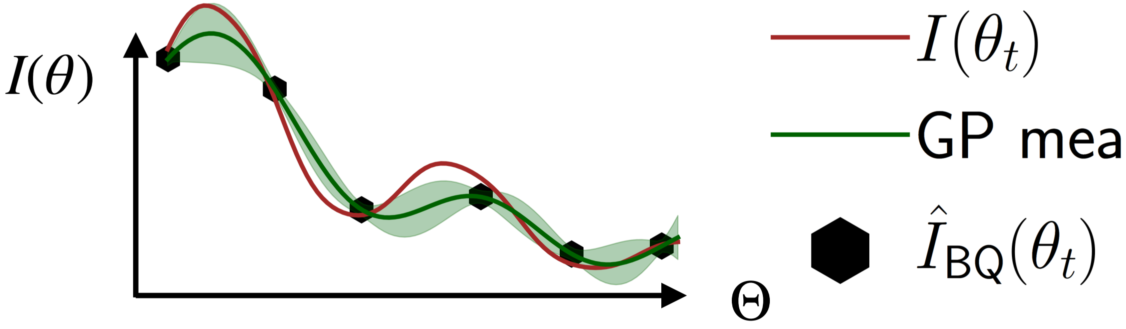
$$\hat{I}_{BQ}(\theta_t) = \sum_{i=1}^{N} w_{i,t} f(x_i^{(t)}, \theta_t)$$

Stage II: Heteroscedastic GP regression over $I(\theta)$ with outputs from Stage I

$$\hat{I}_{\text{CBQ}}(\theta) := k_\Theta\left(\theta, \theta_{1:T}\right)\left(\mathbf{K}_\Theta + \text{diag}\left(\sigma^2_{\text{BQ}}\left(\theta_{1:T}\right)\right)\right)^{-1} \hat{I}_{\text{BQ}}(\theta_{1:T})$$

$$\hat{\sigma}_{\text{CBQ}}(\theta)^2 := \ldots$$





Convergence guarantees

- **Theorem (informal):** Under regularity assumptions including
  - The samples $\{x_i^{(t)}\}_{i=1}^N$ are iid from $\pi(x;\theta_t)$. $\theta_1, \ldots, \theta_T$ are iid from $\mathbb{Q}$.
  - $f(\,\cdot\,,\theta)$ has smoothness $s_x > d_x/2$ and $f(x,\,\cdot\,)$ has smoothness $s_\theta > d_\theta/2$.
  - The kernels $k_\mathcal{X}$ and k_Θ have smoothness s_x and s_θ respectively.

Convergence guarantees

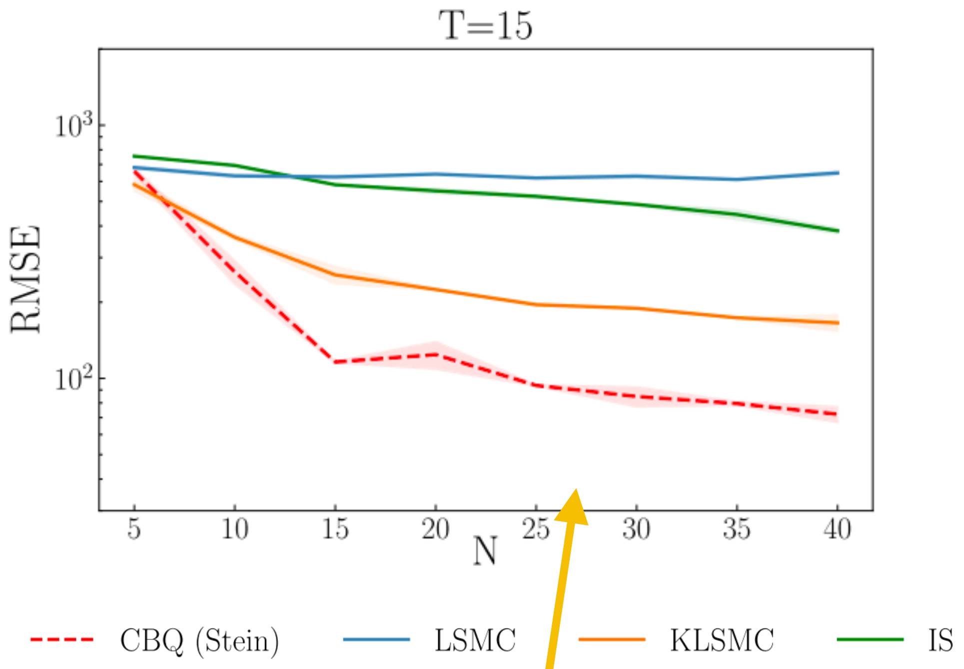
- **Theorem (informal):** Under regularity assumptions including
  - The samples $\{x_i^{(t)}\}_{i=1}^N$ are iid from $\pi(x;\theta_t)$. $\theta_1,\ldots,\theta_T$ are iid from $\mathbb{Q}$.
  - $f(\cdot\,,\theta)$ has smoothness $s_x > d_x/2$ and $f(x,\cdot\,)$ has smoothness $s_\theta > d_\theta/2$.
  - The kernels $k_{\mathcal{X}}$ and k_Θ have smoothness s_x and s_θ respectively.

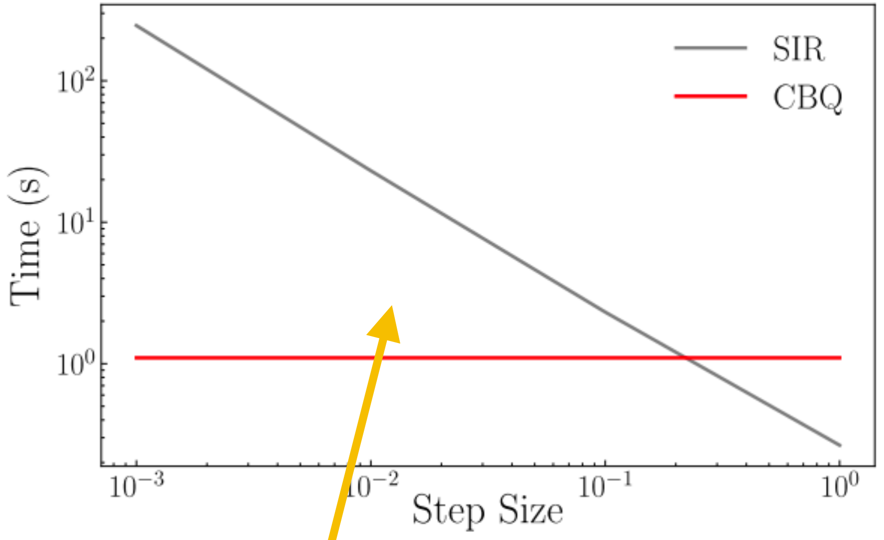
$$\left\|\hat{I}_{CBQ} - I\right\|_{L^2(\Theta)} = \mathcal{O}_P\left(N^{-\frac{s_x}{d_x}} + T^{-\frac{1}{4}}\right)$$

BQ rate in N, but non-parametric rate in T?

Change the algorithm slightly, we obtain $\mathcal{O}_P\left(N^{-\frac{s_x}{d_x}} + T^{-\frac{s_\theta}{d_\theta}}\right)$ **(!)**

Experiment: SIR model

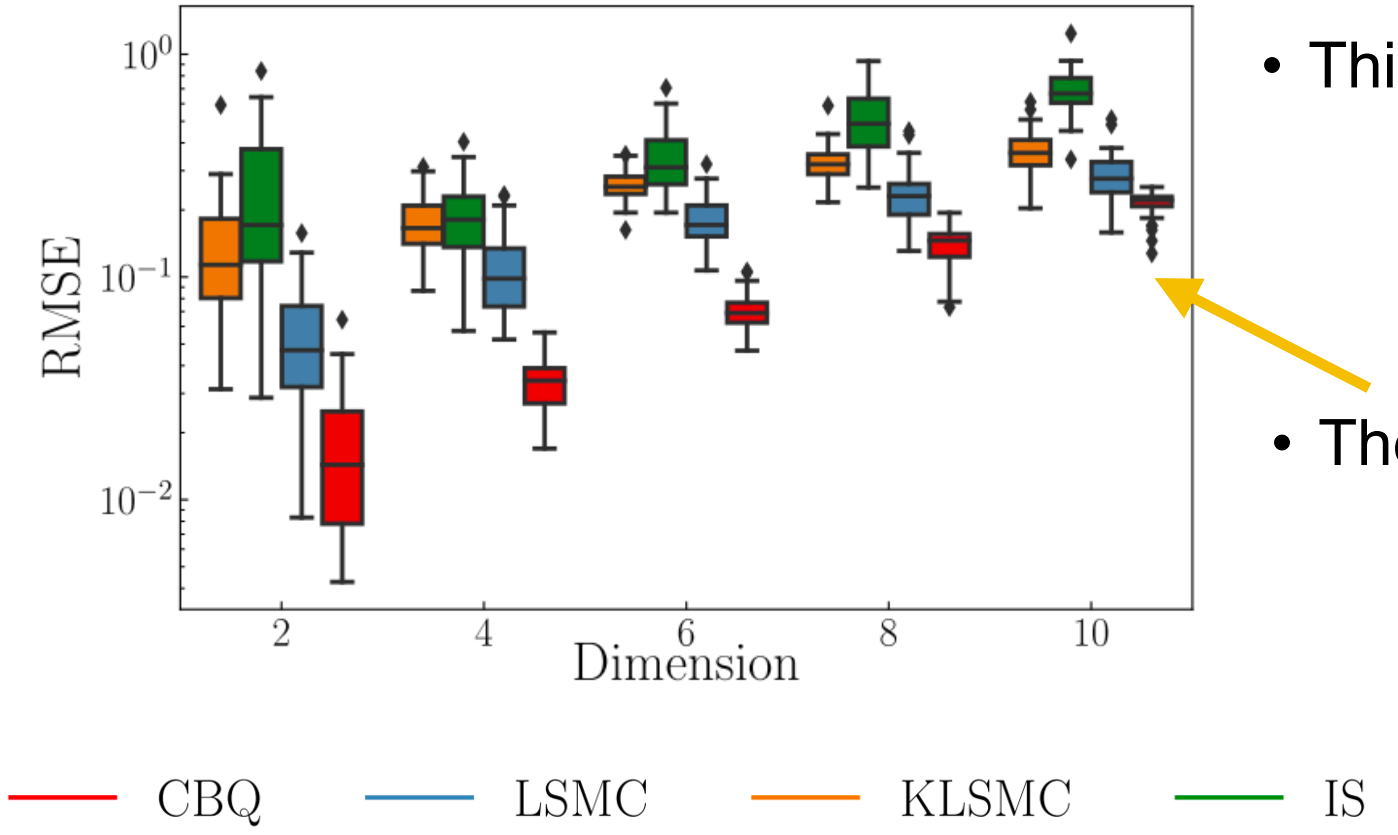




We get much faster convergence than alternatives!

The cost of doing CBQ is negligible compared to simulation cost from the SIR model.

Experiment: Curse of dimension

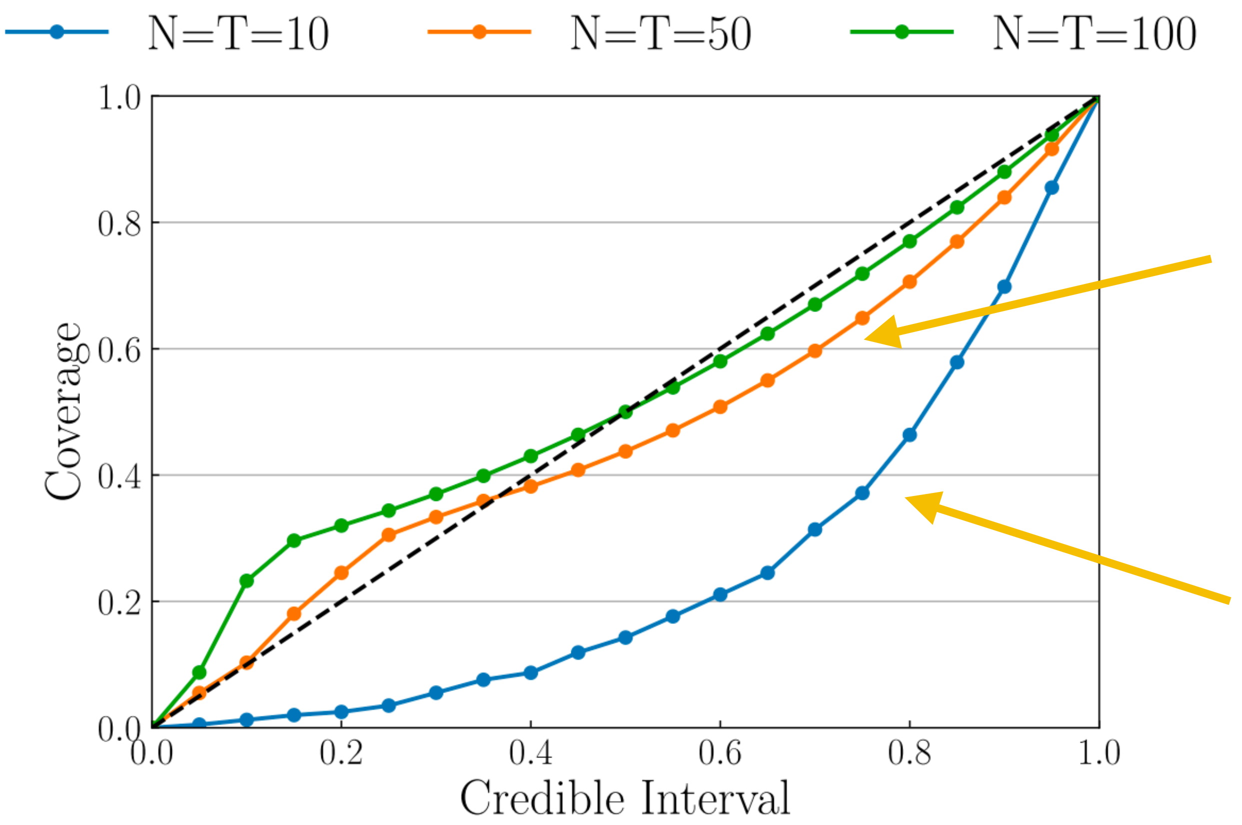


- This shows in our convergence rate…

$$\mathcal{O}_P\left(N^{-\frac{s_x}{d_x}} + T^{-\frac{1}{4}}\right)$$

- The rate bears out in practice

Calibration of the CBQ posterior



- But things get better for large N, T (although we didn't study this theoretically…)
- The CBQ posterior tends to be poorly calibrated when the number of data points is extremely small

Connection to Extrapolation

- The target of interest is $I(0) = \mathbb{E}_{X \sim \pi_0}[f(X)]$
    - We are given estimate $\hat{I}(t) \approx I(t) = \mathbb{E}_{X \sim \pi_t}[f(X)]$
    - Example 1: π_t is the power posterior in Bayesian inference.
- CBQ: BQ to estimate $\hat{I}_{BQ}(t)$ for $t \neq 0$. GP to estimate $\hat{I}_{CBQ}(0)$.
- For the CBQ rate to hold, $t_1, \cdots, t_T$ is iid. **Fill distance?**

Conclusion and future work

- We proposed CBQ to approximate parametric expectations.
  - Fast rate of convergence.
  - Finite-sample Bayesian uncertainty quantification.

$$I(\theta) = \int_{\mathcal{X}} f(x,\theta)\pi(x;\theta)dx$$

- Plenty of work remaining including:
  - Active learning for sequential sample selection.

Any Questions?

@Hudson19990518

hudsonchen.github.io

Reparameterization "trick"

- Two major bottlenecks of BQ / CBQ are:
  - The closed-form kernel mean embedding $\mu(x) = \mathbb{E}_{X\sim\pi}[k(X,x)]$.
  - The $\mathcal{O}(N^3)$ computational cost of inverting the Gram matrix.
- $U \sim \nu$ is another random variable with density q which is easy to sample from.
- Suppose we can find an invertible transformation Φ such that $X = \Phi(U)$.

$$I = \int f(x)\pi(x)dx = \int f(\Phi(u))q(u)du$$

$$\hat{I}_{BQ} = \mathbb{E}_{U\sim\nu}[k(U, u_{1:N})]k(u_{1:N}, u_{1:N})^{-1}(f \circ \Phi)(u_{1:N})$$

  - The closed-form kernel mean embedding $\mu(u) = \mathbb{E}_{U\sim\nu}[k(U,u)]$.
  - $\mathbb{E}_{U\sim\nu}[k(U, u_{1:N})]k(u_{1:N}, u_{1:N})^{-1}$ does not depend on f so can be precomputed.