Bayesian quadrature for parametric expectations

Dr François-Xavier Briol
Department of Statistical Science
University College London



Topic of this talk

Conditional Bayesian Quadrature

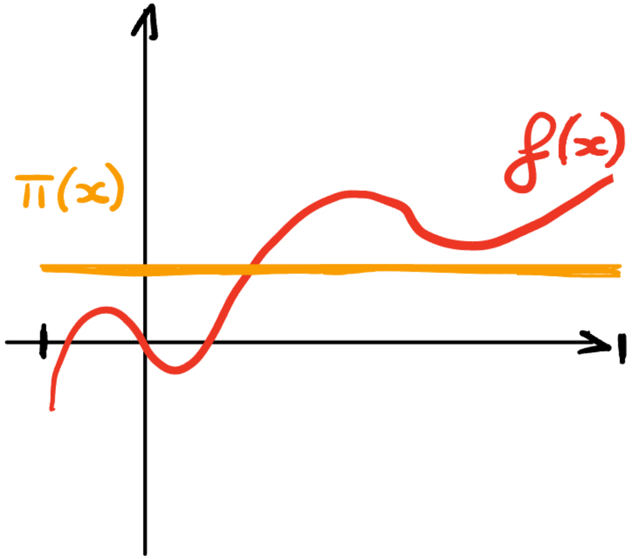
Zonghao Chen[1,*] **Masha Naslidnyk**[1,*] **Arthur Gretton**[2] **François-Xavier Briol**[3]

[1]Department of Computer Science, University College London, London, UK
[2]Gatsby Computational Neuroscience Unit, University College London, London, UK
[3]Department of Statistical Science, University College London, London, UK

Recently appeared at **UAI 2024**!

Numerical integration

Quantity of interest: $$I = \int_{\mathcal{X}} f(x)\pi(x)dx$$



Numerical integration

Quantity of interest:

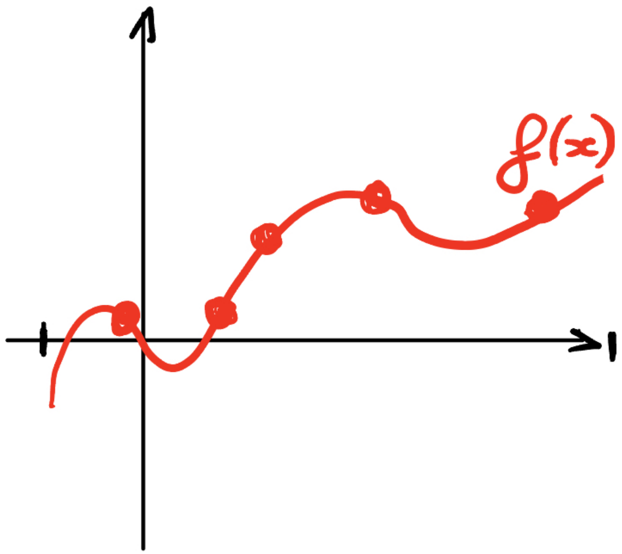
$$I = \int_{\mathscr{X}} f(x)\pi(x)dx$$

Data:

(Expensive??)

$$x_{1:N} := \left[x_1, \cdots, x_N\right]^\top \in \mathscr{X}^N,$$

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



Numerical integration

Quantity of interest:

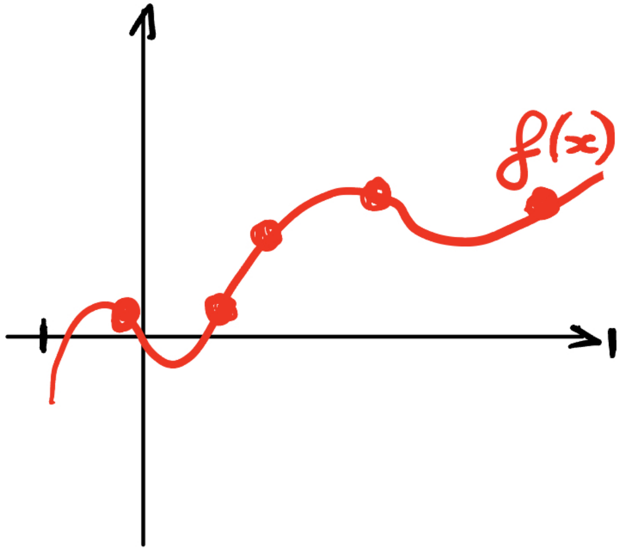
$$I = \int_{\mathcal{X}} f(x)\pi(x)dx$$

Data:

(Expensive??)

$$x_{1:N} := \left[x_1, \cdots, x_N\right]^\top \in \mathcal{X}^N,$$

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



Estimator:

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

Numerical integration

Quantity of interest:

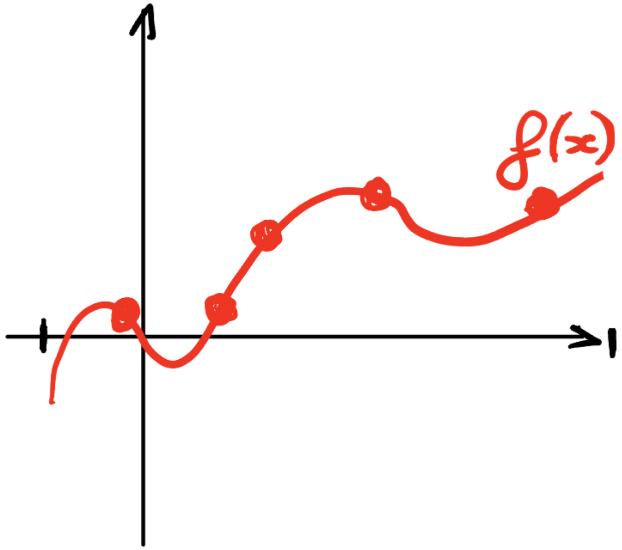
$$I = \int_{\mathcal{X}} f(x)\pi(x)dx$$

Data:

(Expensive??)

$$x_{1:N} := \left[x_1, \cdots, x_N\right]^\top \in \mathcal{X}^N,$$

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



Estimator:

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

Most of the literature:

$\{x_i, w_i\}_{i=1}^N$?????

Related Numerical Integration Tasks

An interesting setting which requires more attention:

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Related Numerical Integration Tasks

An interesting setting which requires more attention:

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

- Rather than brute-forcing each I_t with our favourite algorithm, we can **share information across integration tasks**!

Related Numerical Integration Tasks

An interesting setting which requires more attention:

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

- Rather than brute-forcing each I_t with our favourite algorithm, we can **share information across integration tasks**!
- This can be particularly helpful if the tasks are **"related"**; i.e. we can converge faster!

Related Numerical Integration Tasks

An interesting setting which requires more attention:

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

- Rather than brute-forcing each I_t with our favourite algorithm, we can **share information across integration tasks**!
- This can be particularly helpful if the tasks are **"related"**; i.e. we can converge faster!

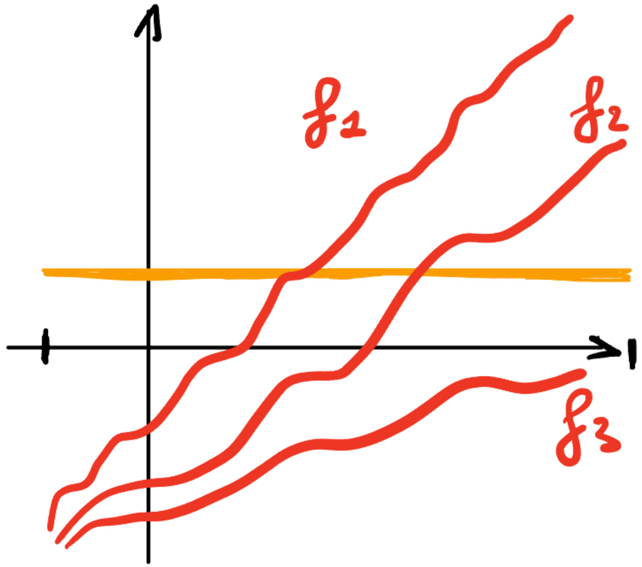
→ **Key question:** What does "related" mean, and how do we take advantage of it?

Related Numerical Integration Tasks

An interesting setting which requires more attention:

$$I_t = \int_{\mathcal{X}} f_t(x)\pi(x)dx \qquad t \in \{1,\ldots,T\}$$

Example 1:
Related integrands
$f_1, \ldots, f_T$

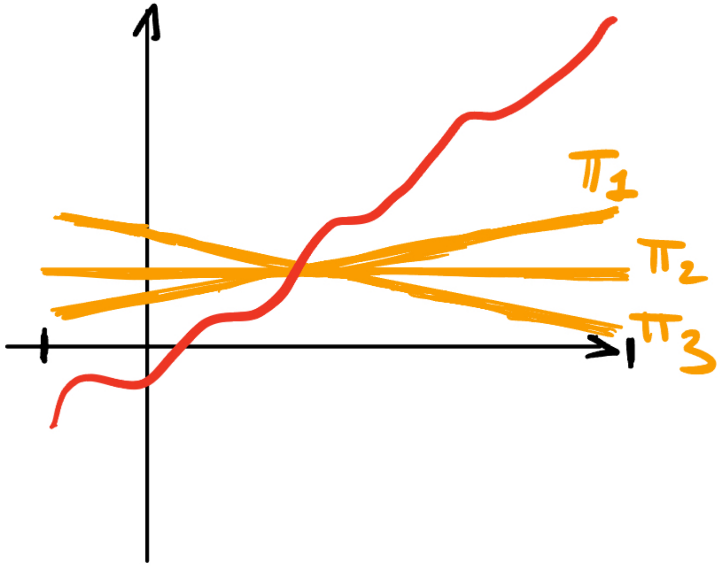


Related Numerical Integration Tasks

An interesting setting which requires more attention:

$$I_t = \int_{\mathcal{X}} f(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Example 2:
Related densities
$\pi_1, \ldots, \pi_T$



Existing work

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Existing work

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Importance sampling: Sample $x_1, \ldots, x_N$ from some π, then reweight the samples:

$$w_i = \frac{\pi_t(x_i)}{\pi(x_i)} \qquad I \approx \hat{I} = \sum_{i=1}^{N} w_i f_t(x_i)$$

Existing work

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Importance sampling: Sample $x_1, \ldots, x_N$ from some π, then reweight the samples:

$$w_i = \frac{\pi_t(x_i)}{\pi(x_i)} \qquad I \approx \hat{I} = \sum_{i=1}^{N} w_i f_t(x_i)$$

This doesn't take into account the relationship between $f_1, \ldots, f_T$ and only works under relatively strong conditions on these weights (variance often infinite!!).

Existing work

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Importance sampling: Sample $x_1, \ldots, x_N$ from some π, then reweight the samples:

$$w_i = \frac{\pi_t(x_i)}{\pi(x_i)} \qquad I \approx \hat{I} = \sum_{i=1}^{N} w_i f_t(x_i)$$

This doesn't take into account the relationship between $f_1, \ldots, f_T$ and only works under relatively strong conditions on these weights (variance often infinite!!).

Madras, N., & Piccioni, M. (1999). Importance sampling for families of distributions. *The Annals of Applied Probability*, *9*(4), 1202–1225.

Tang, X. (2013). *Importance sampling for efficient parametric simulation*. Boston University.

Demange-Chryst, J., Bachoc, F., & Morio, J. (2022). Efficient estimation of multiple expectations with the same sample by adaptive importance sampling and control variates. *arXiv:2212.00568*.

Existing work

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Existing work

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Multilevel methods: $f_1 \approx f_2 \approx \ldots \approx f_T$ are increasingly more accurate approximations of f.

Existing work

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Multilevel methods: $f_1 \approx f_2 \approx \ldots \approx f_T$ are increasingly more accurate approximations of f.

Giles, M. B. (2015). Multilevel Monte Carlo methods. *Acta Numerica*, *24*, 259–328.

Existing work

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Multilevel methods: $f_1 \approx f_2 \approx \ldots \approx f_T$ are increasingly more accurate approximations of f.

Giles, M. B. (2015). Multilevel Monte Carlo methods. *Acta Numerica*, *24*, 259–328.

Multi-task learning: $f = [f_1, \ldots, f_T]^\top$ is a vector-valued func. + we model correlations across outputs

Existing work

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Multilevel methods: $f_1 \approx f_2 \approx \ldots \approx f_T$ are increasingly more accurate approximations of f.

Giles, M. B. (2015). Multilevel Monte Carlo methods. *Acta Numerica*, *24*, 259–328.

Multi-task learning: $f = [f_1, \ldots, f_T]^\top$ is a vector-valued func. + we model correlations across outputs

Xi, X., Briol, F.-X., & Girolami, M. (2018). Bayesian quadrature for multiple related integrals. *ICML*, 8533–8564.

Gessner, A., Gonzalez, J., & Mahsereci, M. (2019). Active multi-information source Bayesian quadrature. *UAI*.

Existing work

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Multilevel methods: $f_1 \approx f_2 \approx \ldots \approx f_T$ are increasingly more accurate approximations of f.

Giles, M. B. (2015). Multilevel Monte Carlo methods. *Acta Numerica*, *24*, 259–328.

Multi-task learning: $f = [f_1, \ldots, f_T]^\top$ is a vector-valued func. + we model correlations across outputs

Xi, X., Briol, F.-X., & Girolami, M. (2018). Bayesian quadrature for multiple related integrals. *ICML*, 8533–8564.

Gessner, A., Gonzalez, J., & Mahsereci, M. (2019). Active multi-information source Bayesian quadrature. *UAI*.

Meta-learning: $f_1, \ldots, f_T$ and $\pi_1, \ldots, \pi_T$ are iid draws from a distribution over tasks.

Existing work

$$I_t = \int_{\mathcal{X}_t} f_t(x)\pi_t(x)dx \qquad t \in \{1,\ldots,T\}$$

Multilevel methods: $f_1 \approx f_2 \approx \ldots \approx f_T$ are increasingly more accurate approximations of f.

Giles, M. B. (2015). Multilevel Monte Carlo methods. *Acta Numerica*, *24*, 259–328.

Multi-task learning: $f = [f_1, \ldots, f_T]^\top$ is a vector-valued func. + we model correlations across outputs

Xi, X., Briol, F.-X., & Girolami, M. (2018). Bayesian quadrature for multiple related integrals. *ICML*, 8533–8564.

Gessner, A., Gonzalez, J., & Mahsereci, M. (2019). Active multi-information source Bayesian quadrature. *UAI*.

Meta-learning: $f_1, \ldots, f_T$ and $\pi_1, \ldots, \pi_T$ are iid draws from a distribution over tasks.

Sun, Z., Oates, C. J., & Briol, F.-X. (2023). Meta-learning control variates: Variance reduction with limited data. *UAI (oral)*, 2047–2057.

Today: Parametric expectations

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

Today: Parametric expectations

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

We integrate over dummy variable x, not parameter θ

Today: Parametric expectations

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

We integrate over dummy variable x, not parameter θ

- Closely related to multiple task setting if we fix some $\theta_1, \ldots, \theta_T$, in which case $f_t(x) = f(x;\theta_t)$ and $\pi_t(x) = \pi(x;\theta_t)$.

Today: Parametric expectations

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

We integrate over dummy variable x, not parameter θ

- Closely related to multiple task setting if we fix some $\theta_1, \ldots, \theta_T$, in which case $f_t(x) = f(x;\theta_t)$ and $\pi_t(x) = \pi(x;\theta_t)$.
- We additionally will assume some smoothness in θ. That is, we know tasks given by $\theta_t, \theta_{t'}$ are going to be similar if $\theta_t \approx \theta_{t'}$.

Today: Parametric expectations

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

We integrate over dummy variable x, not parameter θ

- Closely related to multiple task setting if we fix some $\theta_1, \ldots, \theta_T$, in which case $f_t(x) = f(x;\theta_t)$ and $\pi_t(x) = \pi(x;\theta_t)$.
- We additionally will assume some smoothness in θ. That is, we know tasks given by $\theta_t, \theta_{t'}$ are going to be similar if $\theta_t \approx \theta_{t'}$.

→ We can take advantage of this assumption by encoding it in our algorithm/model!

Today: Parametric expectations

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

We integrate over dummy variable x, not parameter θ

- Closely related to multiple task setting if we fix some $\theta_1, \ldots, \theta_T$, in which case $f_t(x) = f(x;\theta_t)$ and $\pi_t(x) = \pi(x;\theta_t)$.
- We additionally will assume some smoothness in θ. That is, we know tasks given by $\theta_t, \theta_{t'}$ are going to be similar if $\theta_t \approx \theta_{t'}$.

→ We can take advantage of this assumption by encoding it in our algorithm/model!

[Several other talks at MCQMC, or papers from this community!]

The Setting

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

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

The Setting

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

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

Data: We have the following "data" available:

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

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

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

The Setting

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

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

Data: We have the following "data" available:

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

T tasks

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

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

The Setting

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

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

Data: We have the following "data" available:

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

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

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

The Setting

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

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

Data: We have the following "data" available:

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

T tasks

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

N samples per task

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

Function values at each sample for each task

Example: Bayesian sensitivity analysis

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

Example: Bayesian sensitivity analysis

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

Bayesian posterior

Example: Bayesian sensitivity analysis

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

Hyperparameters in
the prior or likelihood

Bayesian posterior

Example: Bayesian sensitivity analysis

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

QoI;
e.g. moments

Bayesian posterior

Hyperparameters in
the prior or likelihood

Example: Bayesian sensitivity analysis

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

Posterior expectation as function of hyper parameters

QoI;
e.g. moments

Bayesian posterior

Hyperparameters in the prior or likelihood

Example: Bayesian sensitivity analysis

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

Posterior expectation as function of hyper parameters

QoI; e.g. moments

Bayesian posterior

Hyperparameters in the prior or likelihood

Bornn, L., Doucet, A., & Gottardo, R. (2010). An efficient computational approach for prior sensitivity analysis and cross-validation. *Canadian Journal of Statistics*, *38*(1), 47–64.

Kallioinen, N., Paananen, T., Bürkner, P. C., & Vehtari, A. (2024). Detecting and diagnosing prior and likelihood sensitivity with power-scaling. *Statistics and Computing*, *34*(1), 1–27.

Example: Bayesian sensitivity analysis

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

Posterior expectation as function of hyper parameters

QoI; e.g. moments

Bayesian posterior

Hyperparameters in the prior or likelihood

Bornn, L., Doucet, A., & Gottardo, R. (2010). An efficient computational approach for prior sensitivity analysis and cross-validation. *Canadian Journal of Statistics*, *38*(1), 47–64.

Kallioinen, N., Paananen, T., Bürkner, P. C., & Vehtari, A. (2024). Detecting and diagnosing prior and likelihood sensitivity with power-scaling. *Statistics and Computing*, *34*(1), 1–27.

Most of the existing work is based on some form of importance sampling…

Example: Nested expectations

$$\int_{\theta} \phi\left(I(\theta)\right) q(\theta) d\theta = \int_{\theta} \phi\left(\int_{\mathcal{X}} f(x;\theta)\pi(x;\theta)dx \right) q(\theta) d\theta$$

Example: Nested expectations

$$\int_{\theta} \phi\left(I(\theta)\right) q(\theta) d\theta = \int_{\theta} \phi\left(\int_{\mathcal{X}} f(x;\theta)\pi(x;\theta)dx\right) q(\theta) d\theta$$

Health economics: The expected value of perfect information is a nested expectation telling us whether it is worth going to do some (**potentially expensive**) tests on patients.

Example: Nested expectations

$$\int_{\theta} \phi\left(I(\theta)\right) q(\theta) d\theta = \int_{\theta} \phi\left(\int_{\mathcal{X}} f(x;\theta)\pi(x;\theta)dx\right) q(\theta) d\theta$$

Health economics: The expected value of perfect information is a nested expectation telling us whether it is worth going to do some (**potentially expensive**) tests on patients.

Active learning/Bayesian optimisation: This comes up in acquisition functions when you want to select points for multiple function evaluations at a time.

Example: Nested expectations

$$\int_\theta \phi\left(I(\theta)\right) q(\theta)d\theta = \int_\theta \phi\left(\int_{\mathcal{X}} f(x;\theta)\pi(x;\theta)dx\right) q(\theta)d\theta$$

- **Health economics:** The expected value of perfect information is a nested expectation telling us whether it is worth going to do some (**potentially expensive**) tests on patients.
- **Active learning/Bayesian optimisation:** This comes up in acquisition functions when you want to select points for multiple function evaluations at a time.
- **Many others…** Bayesian experimental design, statistical divergences for conditional distributions, etc.. etc..

Least-squares Monte Carlo

Longstaff, F. A., & Schwartz, E. S. (2001). Valuing American options by simulation: A simple least-squares approach. *Review of Financial Studies*, *14*(1), 113–147.

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

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

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

Stage I: Compute Monte Carlo estimators for $I(\theta_1), \ldots, I(\theta_T)$:

$$\hat{I}_{\mathrm{MC}}(\theta_t) = \frac{1}{N}\sum_{i=1}^{N} f(x^t_i; \theta_t)$$

Least-squares Monte Carlo

Longstaff, F. A., & Schwartz, E. S. (2001). Valuing American options by simulation: A simple least-squares approach. *Review of Financial Studies*, *14*(1), 113–147.

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

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

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

Stage I: Compute Monte Carlo estimators for $I(\theta_1), \ldots, I(\theta_T)$:

$$\hat{I}_{\text{MC}}(\theta_t) = \frac{1}{N} \sum_{i=1}^{N} f(x^t_i; \theta_t)$$

Stage II: Perform linear regression over Θ using estimators from Stage I:

$$\hat{I}_{\text{LSMC}}(\theta) = \hat{\beta}_0 + \hat{\beta}_1 \theta_1 + \ldots + \hat{\beta}_d \theta_d$$

Least-squares Monte Carlo

Longstaff, F. A., & Schwartz, E. S. (2001). Valuing American options by simulation: A simple least-squares approach. *Review of Financial Studies*, *14*(1), 113–147.

$$x_{1:N}^t := [x_1^t, \cdots, x_N^t]^\top \in \mathcal{X}^N$$

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

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

Stage I: Compute Monte Carlo estimators for $I(\theta_1), \ldots, I(\theta_T)$:

$$\hat{I}_{\mathrm{MC}}(\theta_t) = \frac{1}{N} \sum_{i=1}^{N} f(x_i^t; \theta_t)$$

Slow convergence

Stage II: Perform linear regression over Θ using estimators from Stage I:

$$\hat{I}_{\mathrm{LSMC}}(\theta) = \hat{\beta}_0 + \hat{\beta}_1 \theta_1 + \ldots + \hat{\beta}_d \theta_d$$

Least-squares Monte Carlo

Longstaff, F. A., & Schwartz, E. S. (2001). Valuing American options by simulation: A simple least-squares approach. *Review of Financial Studies*, *14*(1), 113–147.

$$x^t_{1:N} := [x^t_1, \cdots, x^t_N]^\top \in \mathcal{X}^N$$
$$\theta_{1:T} := [\theta_1, \cdots, \theta_T]^\top \in \Theta^T$$
$$f(x^t_{1:N}, \theta_t) := [f(x^t_1, \theta_t), \cdots, f(x^t_N, \theta_t)]^\top \in \mathbb{R}^N$$

Stage I: Compute Monte Carlo estimators for $I(\theta_1), \ldots, I(\theta_T)$:

$$\hat{I}_{\mathrm{MC}}(\theta_t) = \frac{1}{N} \sum_{i=1}^{N} f(x^t_i; \theta_t)$$

Slow convergence

Stage II: Perform linear regression over Θ using estimators from Stage I:

$$\hat{I}_{\mathrm{LSMC}}(\theta) = \hat{\beta}_0 + \hat{\beta}_1 \theta_1 + \ldots + \hat{\beta}_d \theta_d$$

Linear model might be poor

Least-squares Monte Carlo

Longstaff, F. A., & Schwartz, E. S. (2001). Valuing American options by simulation: A simple least-squares approach. *Review of Financial Studies*, *14*(1), 113–147.

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

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

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

Stage I: Compute Monte Carlo estimators for $I(\theta_1), \ldots, I(\theta_T)$:

$$\hat{I}_{\mathrm{MC}}(\theta_t) = \frac{1}{N}\sum_{i=1}^{N} f(x^t_i; \theta_t)$$

Slow convergence

Stage II: Perform linear regression over Θ using estimators from Stage I:

$$\hat{I}_{\mathrm{LSMC}}(\theta) = \hat{\beta}_0 + \hat{\beta}_1\theta_1 + \ldots + \hat{\beta}_d\theta_d$$

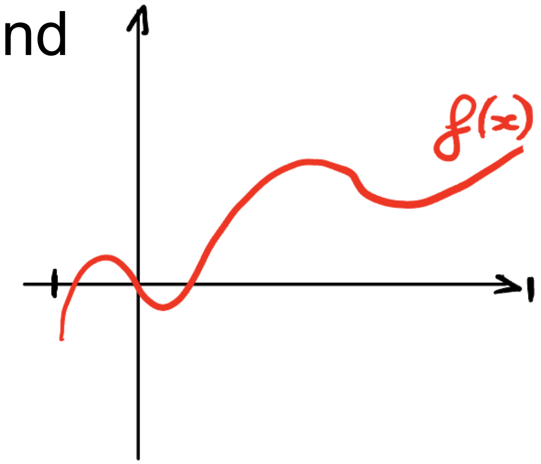
Linear model might be poor

We will try to improve on this with GPs…

Bayesian quadrature

Consider a single task: $I = \int_{\mathcal{X}} f(x)\pi(x)dx$

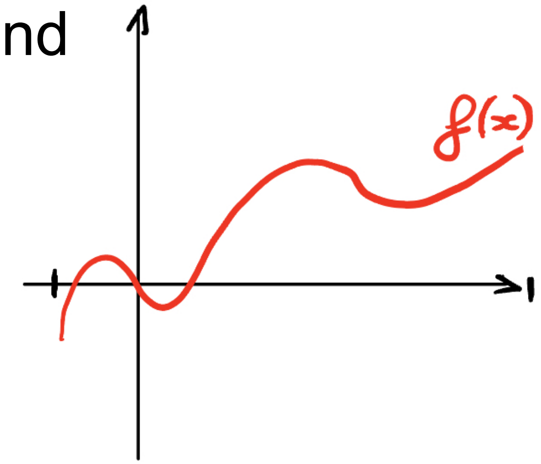
Integrand



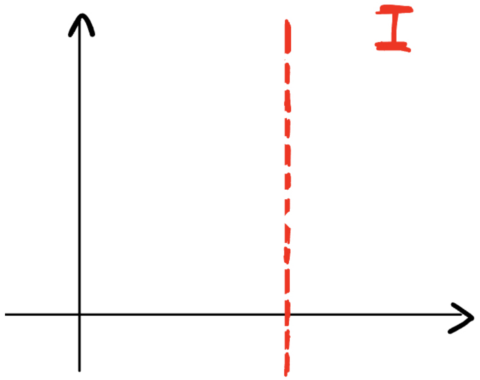
Bayesian quadrature

Consider a single task: $I = \int_{\mathcal{X}} f(x)\pi(x)dx$

Integrand



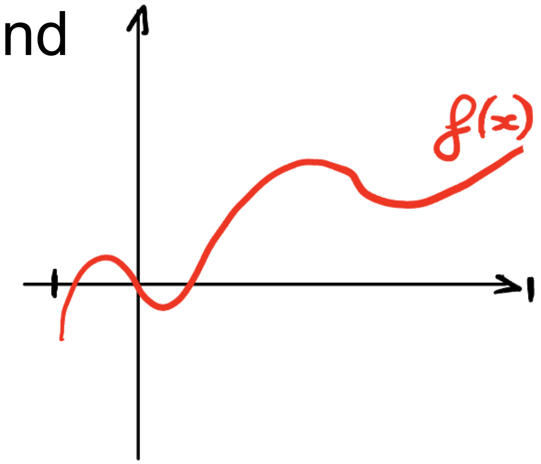
Integral

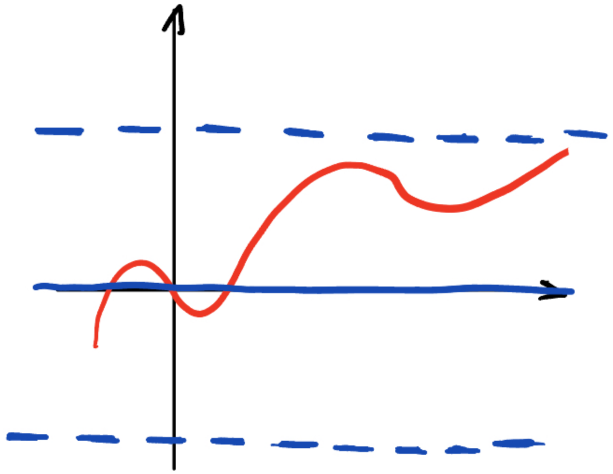


Bayesian quadrature

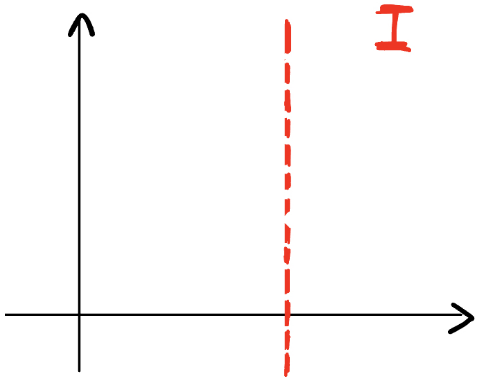
Consider a single task: $I = \int_{\mathcal{X}} f(x)\pi(x)dx$

Integrand





Integral

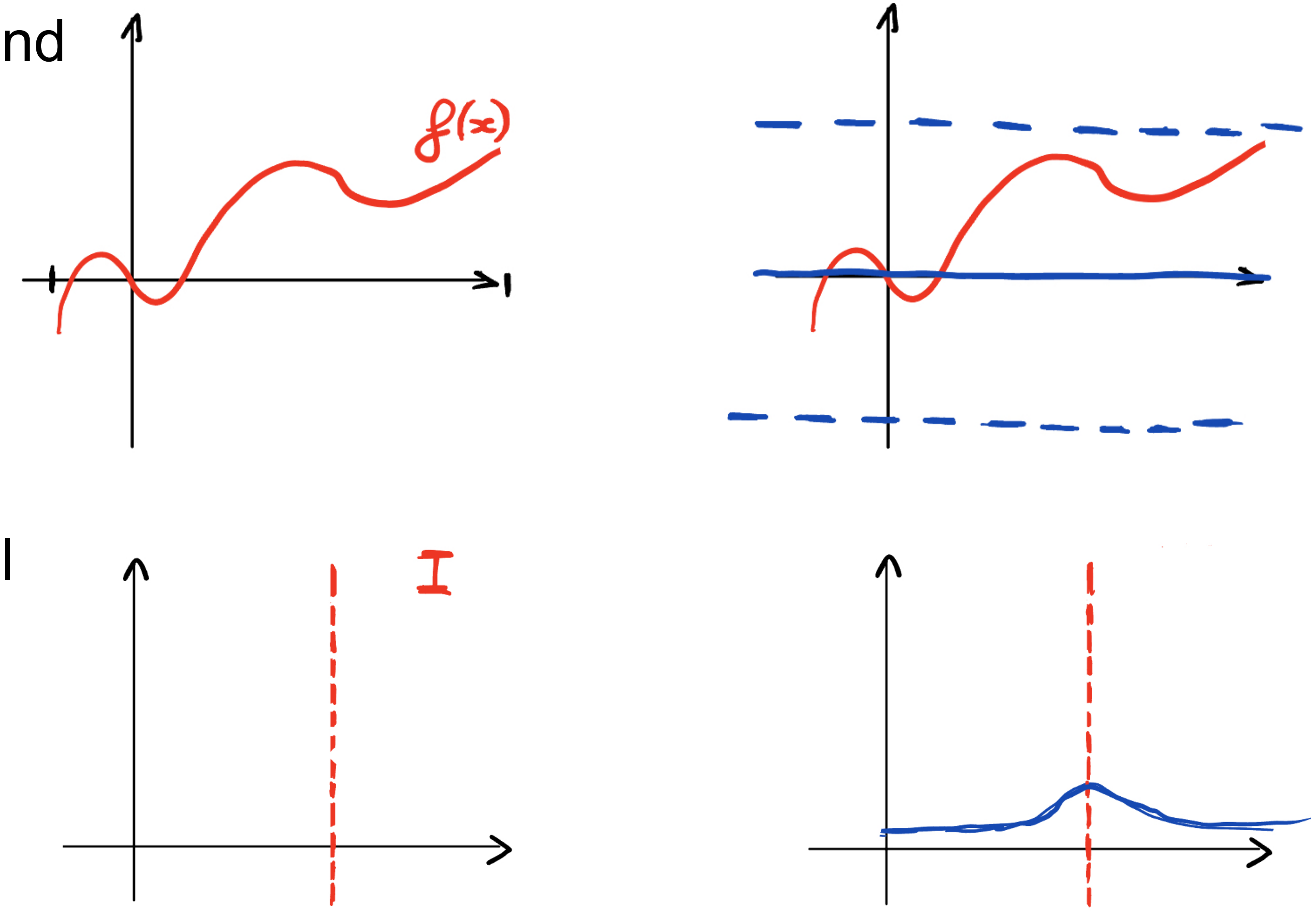


Bayesian quadrature

Consider a single task: $I = \int_{\mathcal{X}} f(x)\pi(x)dx$

Integrand

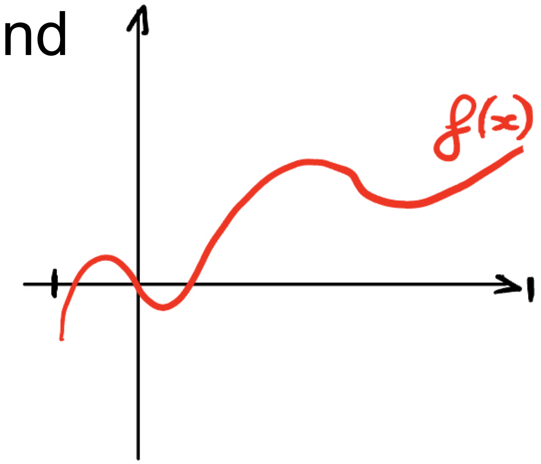
Integral

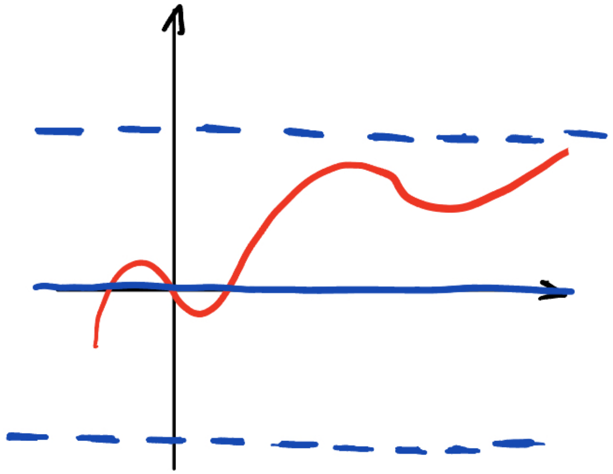


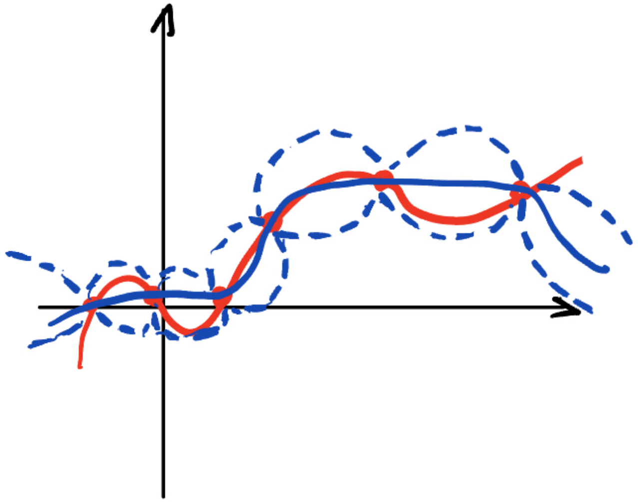
Bayesian quadrature

Consider a single task: $I = \int_{\mathcal{X}} f(x)\pi(x)dx$

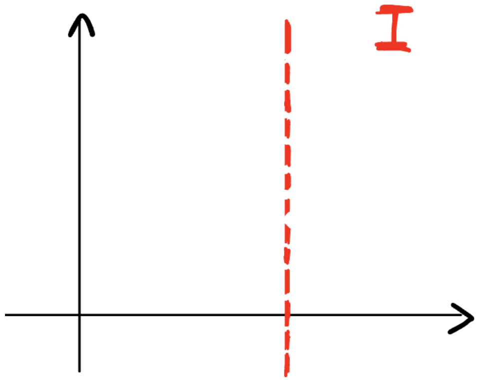
Integrand

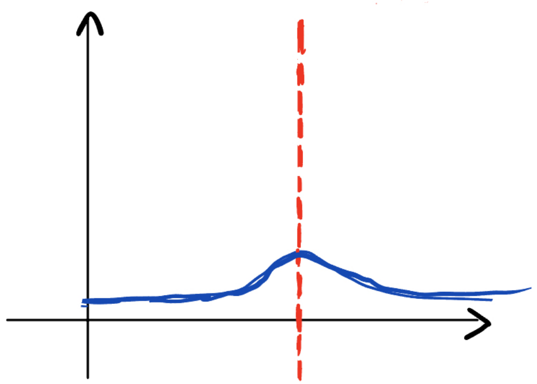






Integral



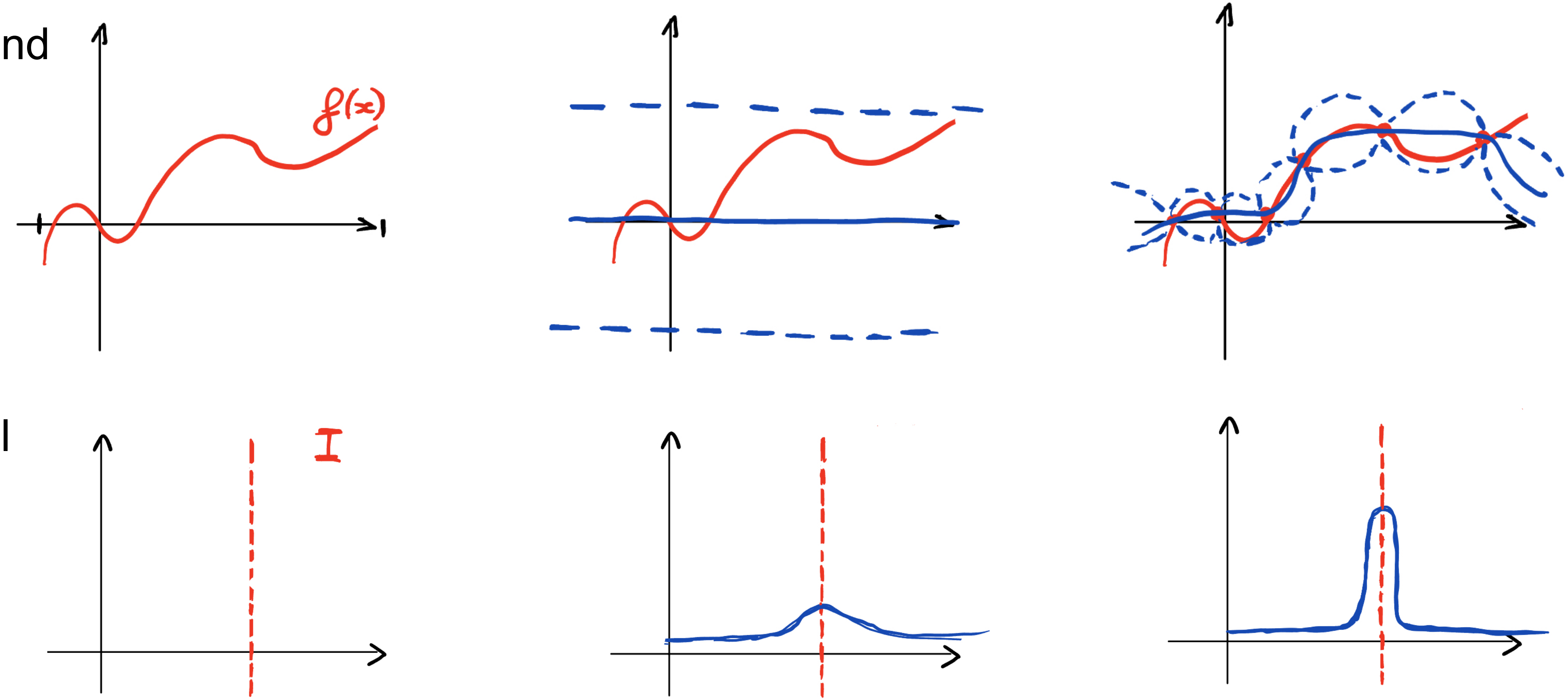


Bayesian quadrature

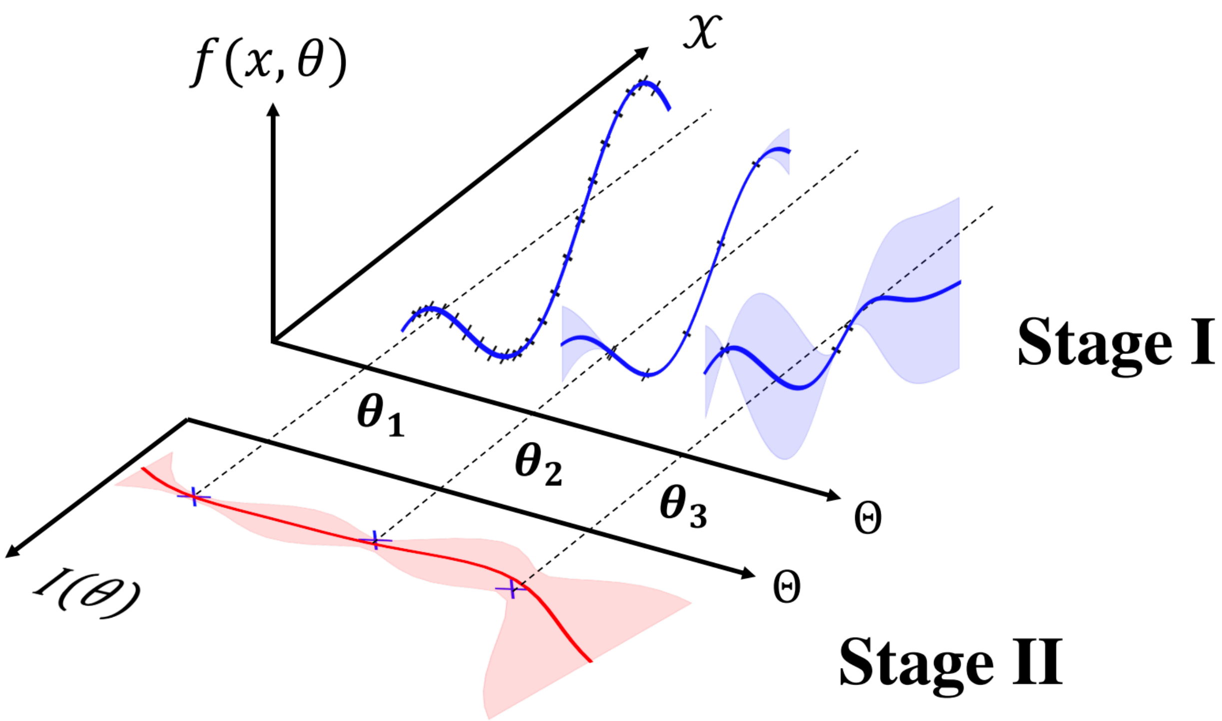
Consider a single task: $I = \int_{\mathcal{X}} f(x)\pi(x)dx$

Integrand

Integral



Conditional Bayesian Quadrature

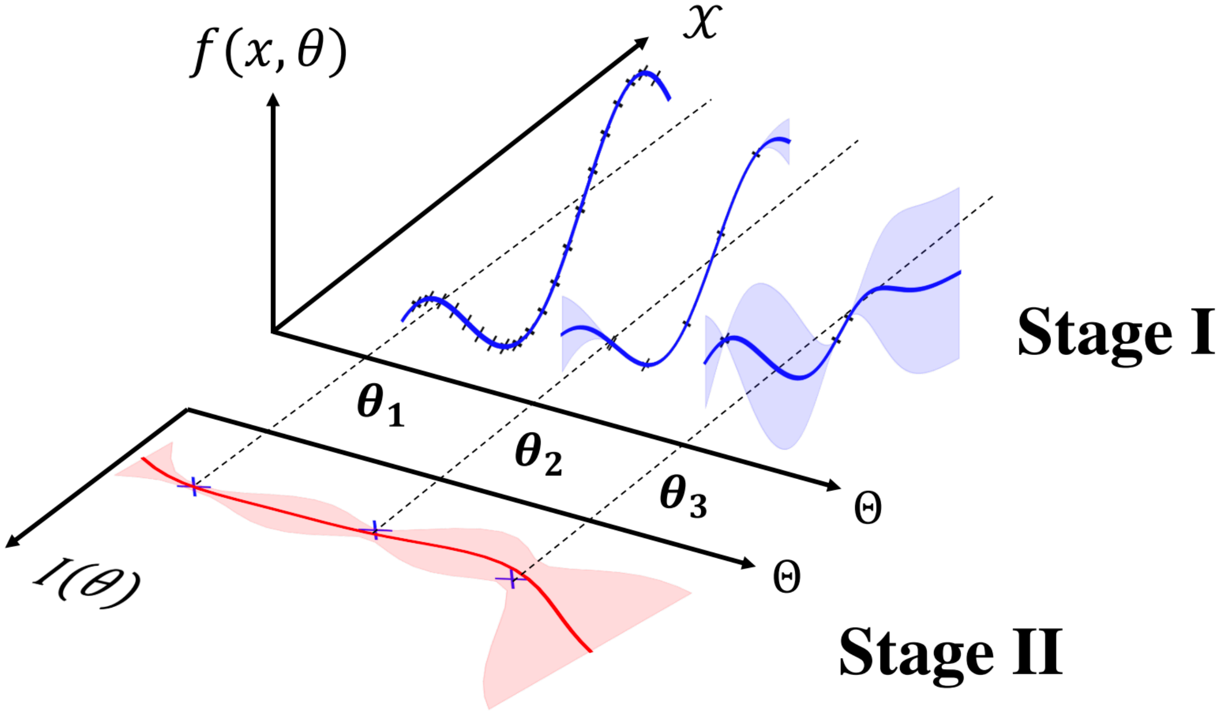


$$x_{1:N}^t := [x_1^t, \cdots, x_N^t]^\top \in \mathcal{X}^N$$

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

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

Conditional Bayesian Quadrature



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

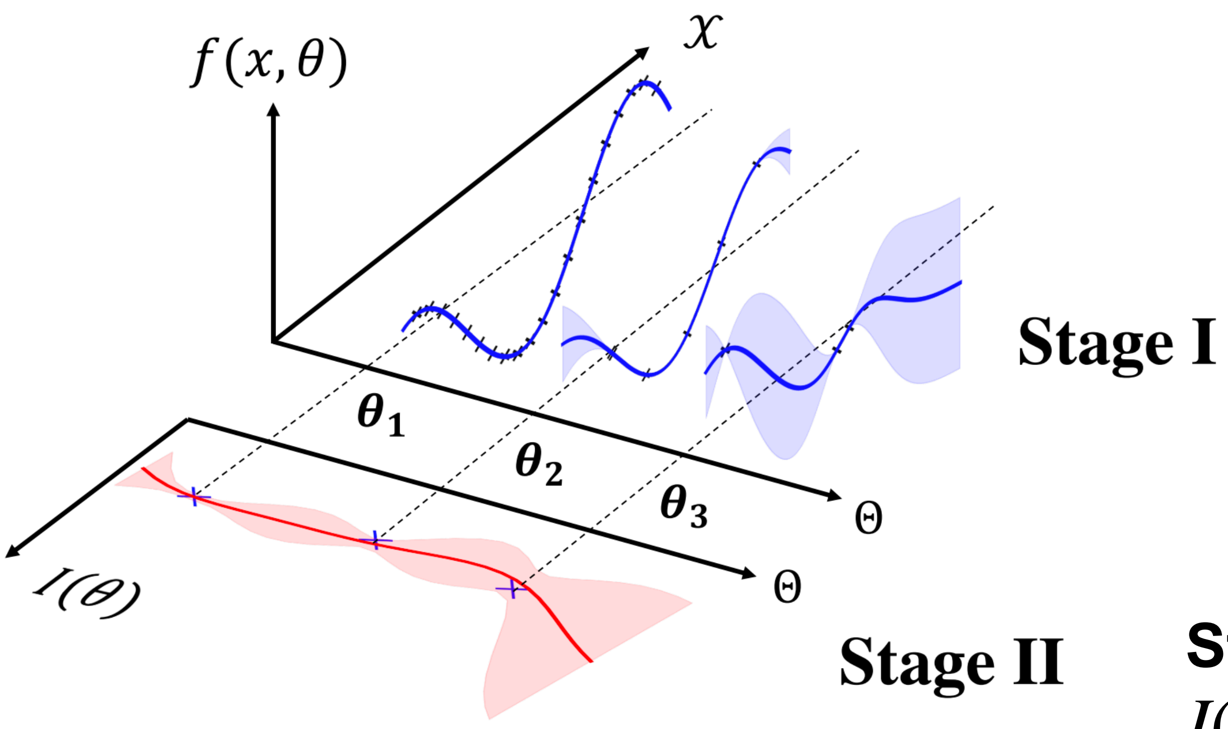
$$\theta_{1:T} := [\theta_1, \cdots, \theta_T]^\top \in \Theta^T$$

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

Stage I: Compute T BQ posteriors:

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

Conditional Bayesian Quadrature



$$x^t_{1:N} := [x^t_1, \cdots, x^t_N]^\top \in \mathcal{X}^N$$
$$\theta_{1:T} := [\theta_1, \cdots, \theta_T]^\top \in \Theta^T$$
$$f(x^t_{1:N}, \theta_t) := [f(x^t_1, \theta_t), \cdots, f(x^t_N, \theta_t)]^\top \in \mathbb{R}^N$$

Stage I: Compute T BQ posteriors:

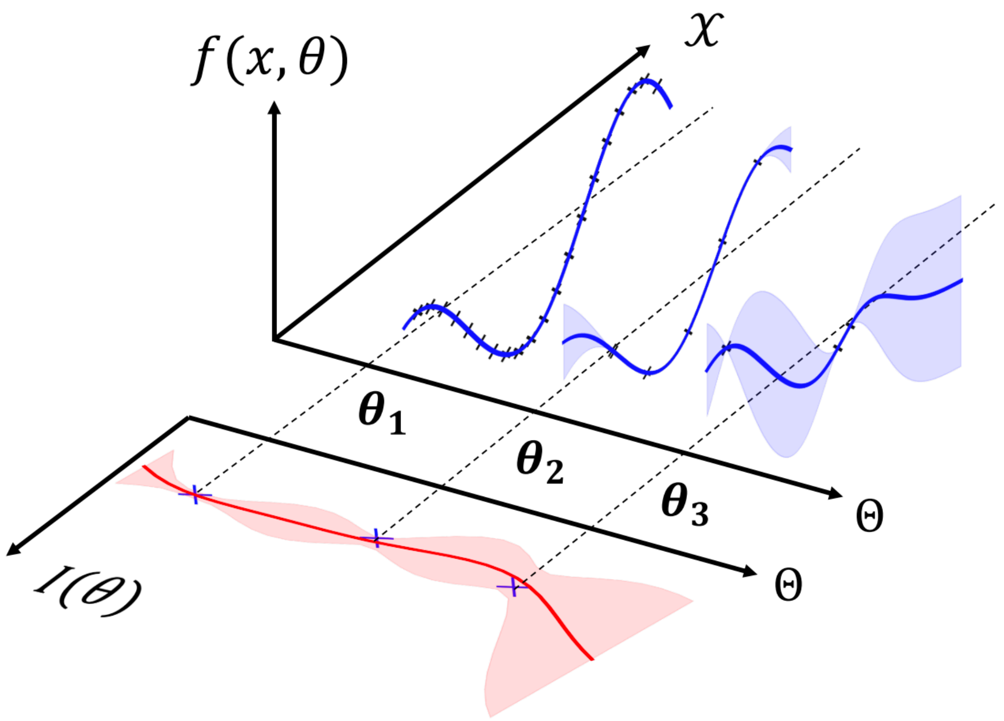
$$\hat{I}_{\text{BQ}}(\theta_1), \sigma^2_{\text{BQ}}(\theta_1), \ldots, \hat{I}_{\text{BQ}}(\theta_T), \sigma^2_{\text{BQ}}(\theta_T),$$

Stage II: Heteroscedastic GP regression over $I(\theta)$ with data from Stage I and likelihood

$$\hat{I}_{\text{BQ}}(\theta_t) = I(\theta_t) + \epsilon_t, \quad \epsilon_t \sim N\left(0, \sigma^2_{\text{BQ}}(\theta_t)\right)$$

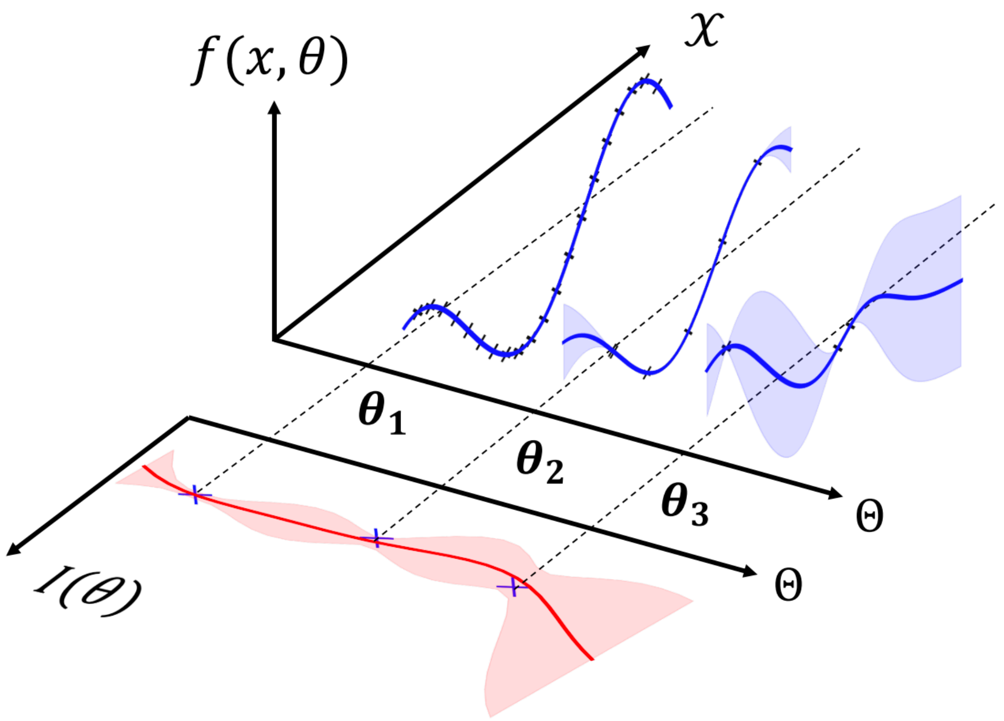
Some remarks on CBQ

- Computational cost is $O(TN^3 + T^3)$.



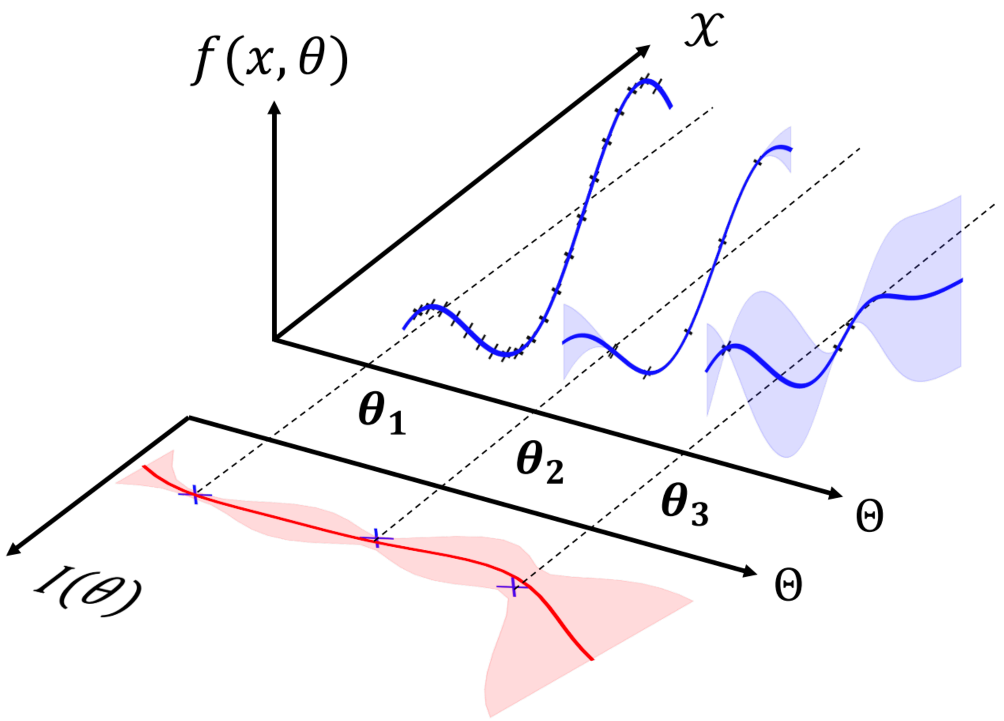
Some remarks on CBQ

- Computational cost is $O(TN^3 + T^3)$.
- Need to pick one GP prior for $x \mapsto f(x; \theta_t)$, and one GP prior for $\theta \mapsto I(\theta)$. Can **encode any prior knowledge**!



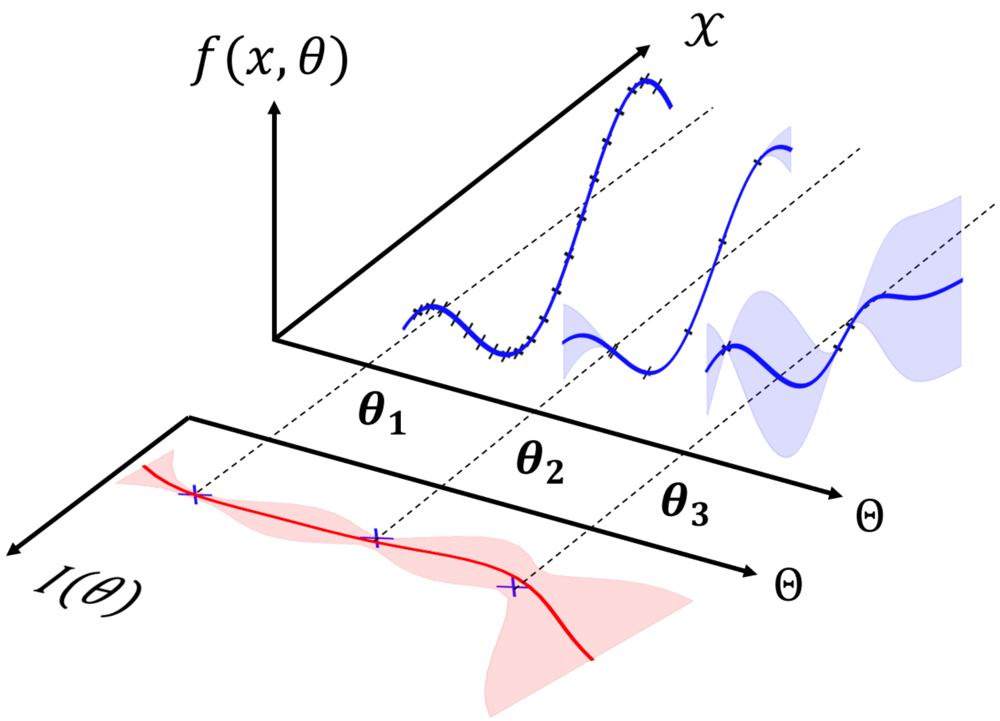
Some remarks on CBQ

- Computational cost is $O(TN^3 + T^3)$.
- Need to pick one GP prior for $x \mapsto f(x; \theta_t)$, and one GP prior for $\theta \mapsto I(\theta)$. Can **encode any prior knowledge**!
- Stage II likelihood accounts for the fact that some BQ estimators might be more accurate than others…



Some remarks on CBQ

- Computational cost is $O(TN^3 + T^3)$.
- Need to pick one GP prior for $x \mapsto f(x; \theta_t)$, and one GP prior for $\theta \mapsto I(\theta)$. Can **encode any prior knowledge**!
- Stage II likelihood accounts for the fact that some BQ estimators might be more accurate than others…
- We end up with a full (Gaussian process) posterior **quantifying our uncertainty** on $I(\theta)$!



Convergence guarantees

- **Theorem (informal):** Under regularity assumptions including

Convergence guarantees

- **Theorem (informal):** Under regularity assumptions including
  - The samples $\{x_i^t\}_{i=1}^n$ and $\theta_1, \ldots, \theta_T$ are iid from $\mathbb{P}_{\theta_t}$ and $\mathbb{Q}$ respectively.

Convergence guarantees

- **Theorem (informal):** Under regularity assumptions including
  - The samples $\{x_i^t\}_{i=1}^n$ and $\theta_1, \ldots, \theta_T$ are iid from $\mathbb{P}_{\theta_t}$ and $\mathbb{Q}$ respectively.
  - $f(\cdot\,;\theta)$ has smoothness $s_f > d/2$ and $f(x;\cdot\,)$ has smoothness $s_I > p/2$.

Convergence guarantees

- **Theorem (informal):** Under regularity assumptions including
  - The samples $\{x_i^t\}_{i=1}^n$ and $\theta_1, \ldots, \theta_T$ are iid from $\mathbb{P}_{\theta_t}$ and $\mathbb{Q}$ respectively.
  - $f(\,\cdot\,;\theta)$ has smoothness $s_f > d/2$ and $f(x;\,\cdot\,)$ has smoothness $s_I > p/2$.
  - The kernels $k_{\mathcal{X}}$ and k_Θ have smoothness $s_{\mathcal{X}} \in (d/2, s_f]$ and $s_\theta \in (p/2, s_I]$ respectively.

Convergence guarantees

- **Theorem (informal):** Under regularity assumptions including
  - The samples $\{x_i^t\}_{i=1}^n$ and $\theta_1, \ldots, \theta_T$ are iid from $\mathbb{P}_{\theta_t}$ and $\mathbb{Q}$ respectively.
  - $f(\cdot\,; \theta)$ has smoothness $s_f > d/2$ and $f(x; \cdot\,)$ has smoothness $s_I > p/2$.
  - The kernels $k_{\mathcal{X}}$ and k_Θ have smoothness $s_{\mathcal{X}} \in (d/2, s_f]$ and $s_\theta \in (p/2, s_I]$ respectively.

Then, with probability $1-\delta$ and for N, T large enough:

$$\left\| \hat{I}_{\text{CBQ}} - I \right\|_{L^2(\Theta)} \leq C_0(\delta) N^{-\frac{s_{\mathcal{X}}}{d}+\varepsilon} + C_1(\delta) T^{-\frac{1}{4}}$$

Convergence guarantees

- **Theorem (informal):** Under regularity assumptions including
  - The samples $\{x_i^t\}_{i=1}^n$ and $\theta_1, \ldots, \theta_T$ are iid from $\mathbb{P}_{\theta_t}$ and $\mathbb{Q}$ respectively.
  - $f(\,\cdot\,;\theta)$ has smoothness $s_f > d/2$ and $f(x;\,\cdot\,)$ has smoothness $s_I > p/2$.
  - The kernels $k_{\mathcal{X}}$ and k_{Θ} have smoothness $s_{\mathcal{X}} \in (d/2, s_f]$ and $s_\theta \in (p/2, s_I]$ respectively.

Then, with probability $1-\delta$ and for N, T large enough:

$$\left\| \hat{I}_{\text{CBQ}} - I \right\|_{L^2(\Theta)} \leq C_0(\delta) N^{-\frac{s_{\mathcal{X}}}{d}+\varepsilon} + C_1(\delta) T^{-\frac{1}{4}}$$

Fast BQ rate!

Convergence guarantees

- **Theorem (informal):** Under regularity assumptions including
  - The samples $\{x_i^t\}_{i=1}^n$ and $\theta_1, \ldots, \theta_T$ are iid from $\mathbb{P}_{\theta_t}$ and $\mathbb{Q}$ respectively.
  - $f(\cdot\,;\theta)$ has smoothness $s_f > d/2$ and $f(x;\cdot\,)$ has smoothness $s_I > p/2$.
  - The kernels $k_{\mathcal{X}}$ and k_Θ have smoothness $s_{\mathcal{X}} \in (d/2, s_f]$ and $s_\theta \in (p/2, s_I]$ respectively.

Then, with probability $1-\delta$ and for N, T large enough:

$$\left\| \hat{I}_{\text{CBQ}} - I \right\|_{L^2(\Theta)} \leq C_0(\delta) N^{-\frac{s_{\mathcal{X}}}{d}+\varepsilon} + C_1(\delta) T^{-\frac{1}{4}}$$

Fast BQ rate!

Slow rate.
Can probably be improved…

Illustration: Bayesian sensitivity analysis

Setting: Bayesian linear regression with $\mathcal{N}(0,\text{diag}(\theta))$ prior with $\theta \in (1,3)^d$ on the coefficients.

QoI: Sum of second moments of the posterior; i.e. $f(x;\theta) = x^\top x$.

Illustration: Bayesian sensitivity analysis

Setting: Bayesian linear regression with $\mathcal{N}(0,\text{diag}(\theta))$ prior with $\theta \in (1,3)^d$ on the coefficients.

QoI: Sum of second moments of the posterior; i.e. $f(x;\theta) = x^\top x$.

Available in closed form!

Illustration: Bayesian sensitivity analysis

Setting: Bayesian linear regression with $\mathcal{N}(0,\text{diag}(\theta))$ prior with $\theta \in (1,3)^d$ on the coefficients.

QoI: Sum of second moments of the posterior; i.e. $f(x;\theta) = x^\top x$.

Available in closed form!

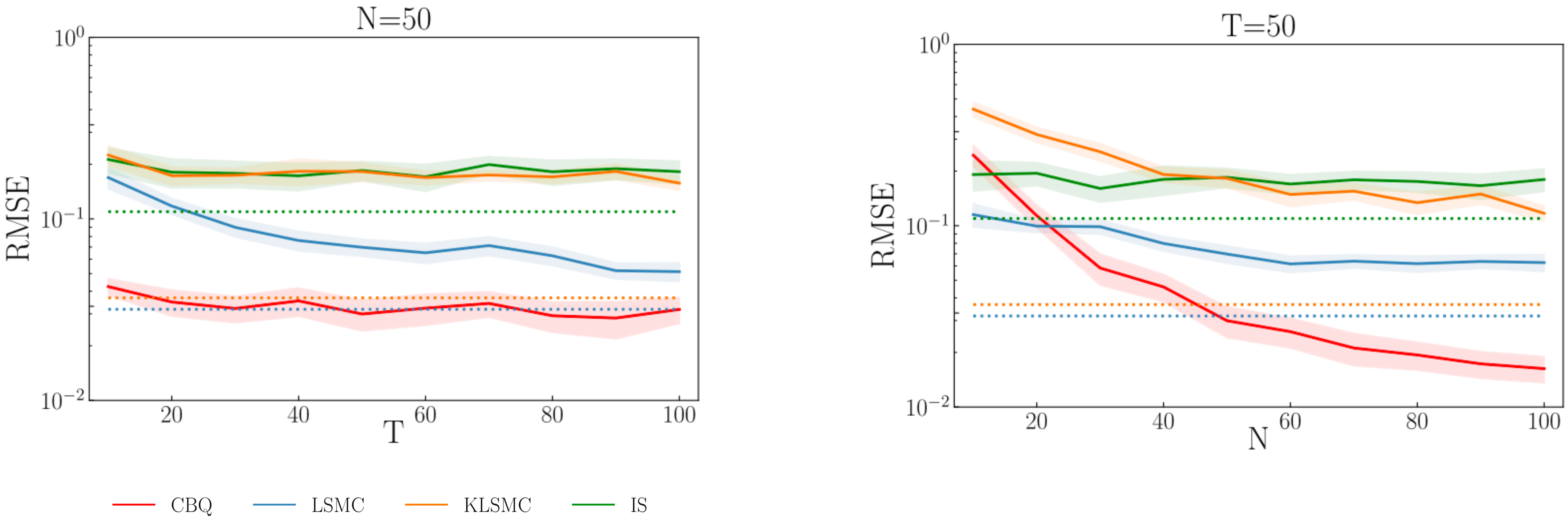
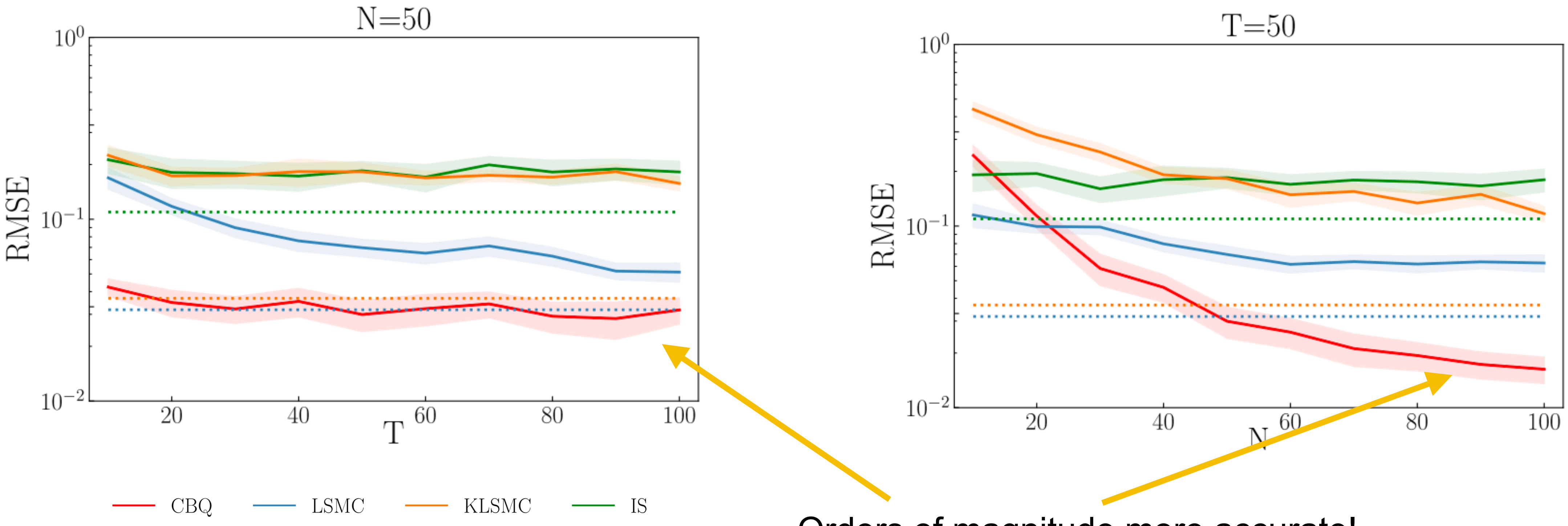


Illustration: Bayesian sensitivity analysis

Setting: Bayesian linear regression with $\mathcal{N}(0,\text{diag}(\theta))$ prior with $\theta \in (1,3)^d$ on the coefficients.

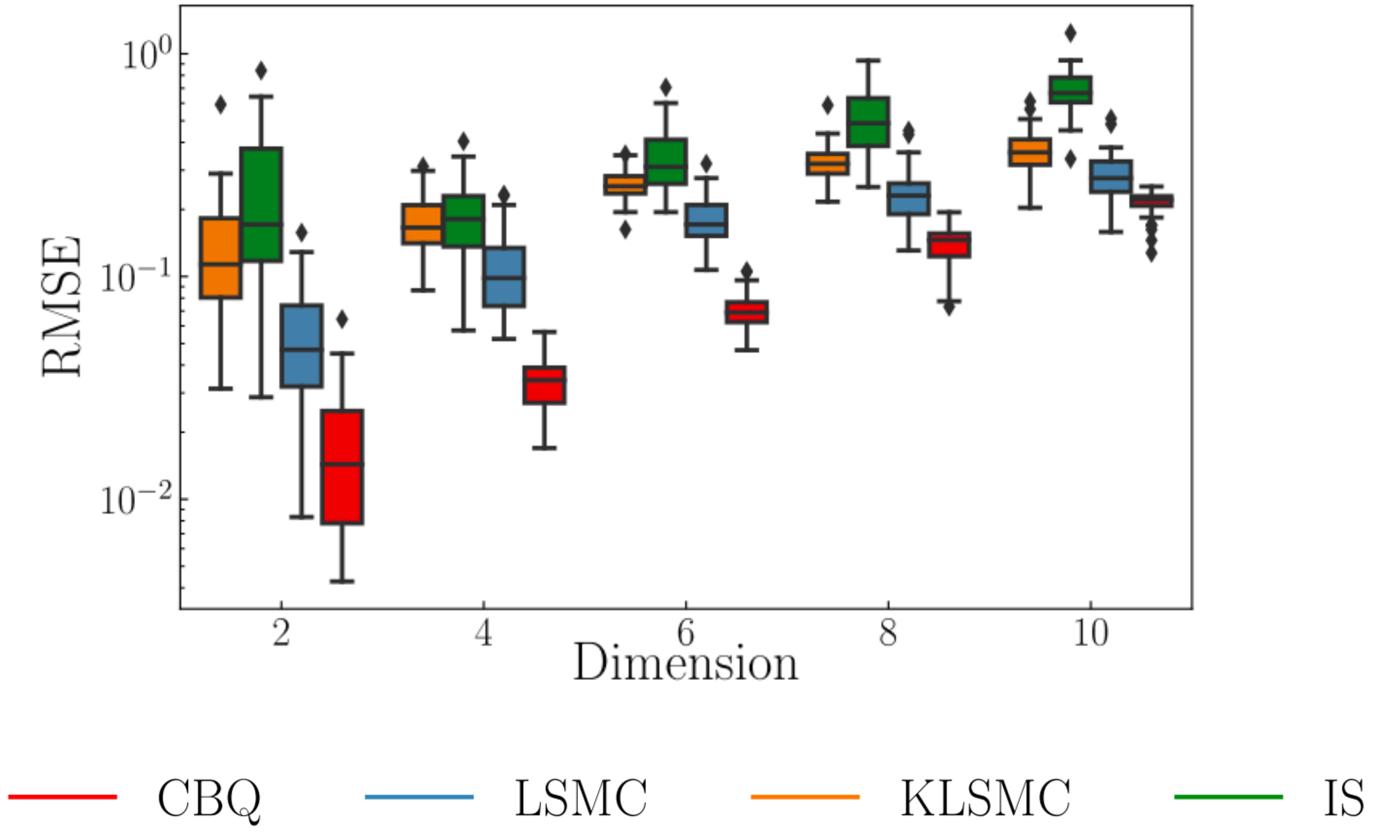
QoI: Sum of second moments of the posterior; i.e. $f(x;\theta) = x^\top x$. **Available in closed form!**



Orders of magnitude more accurate!

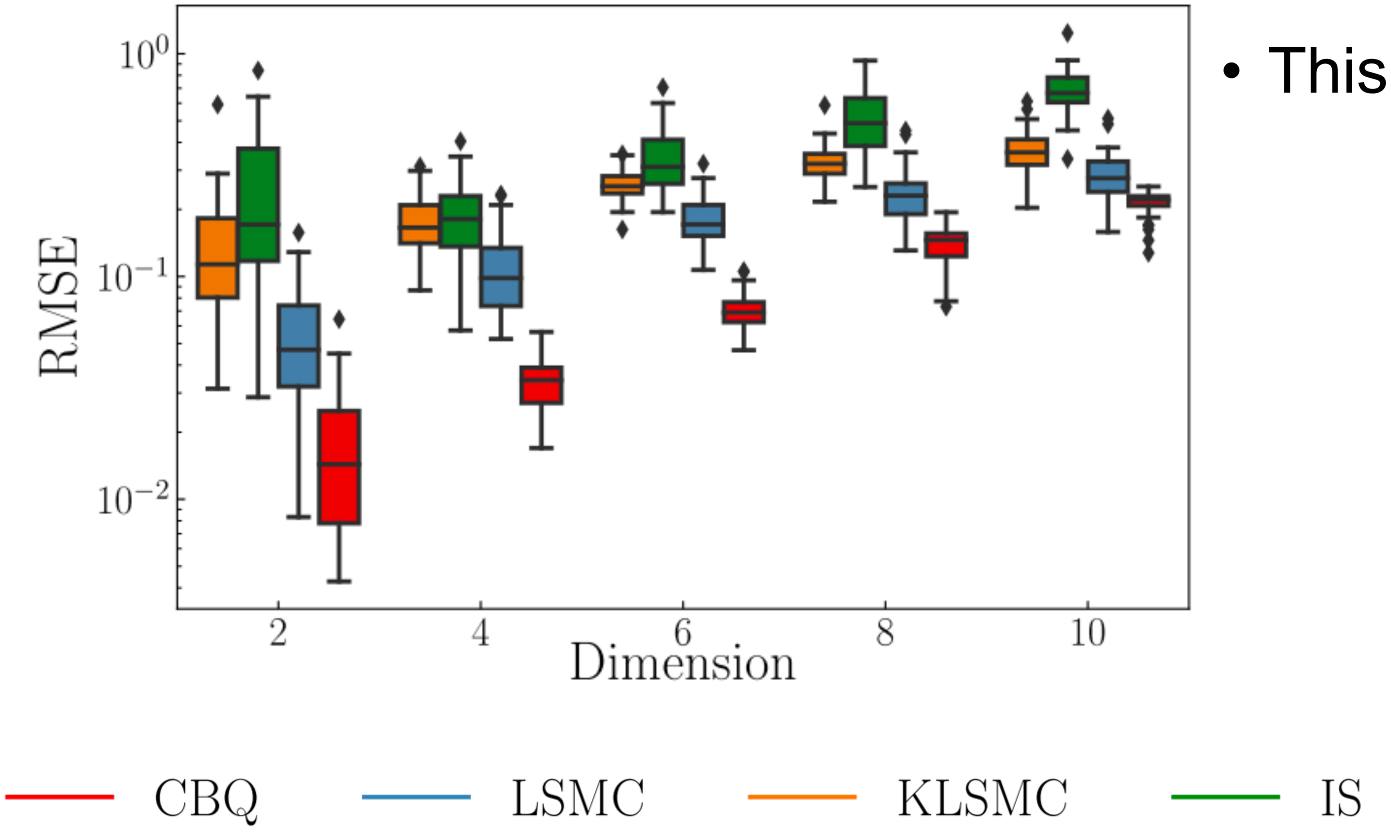
Bayesian sensitivity in varying dims

- A well-known drawback of BQ is that it performs less well in high-dimensions.



Bayesian sensitivity in varying dims

- A well-known drawback of BQ is that it performs less well in high-dimensions.

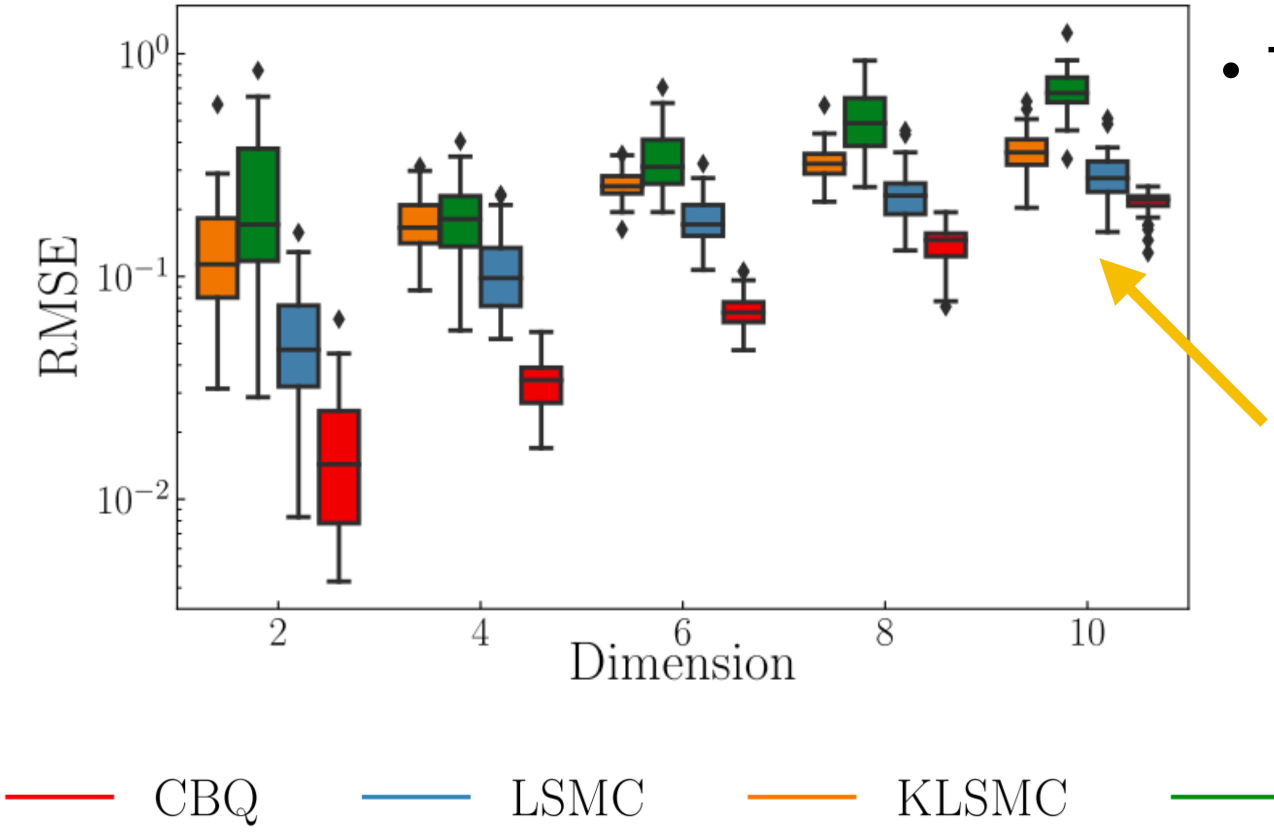


- This shows in our convergence rate…

$$\left\| \ldots \right\|_{L^2(\Theta)} \leq [\ldots] N^{-\frac{s_{\mathcal{X}}}{d}+\varepsilon} + [\ldots]$$

Bayesian sensitivity in varying dims

- A well-known drawback of BQ is that it performs less well in high-dimensions.

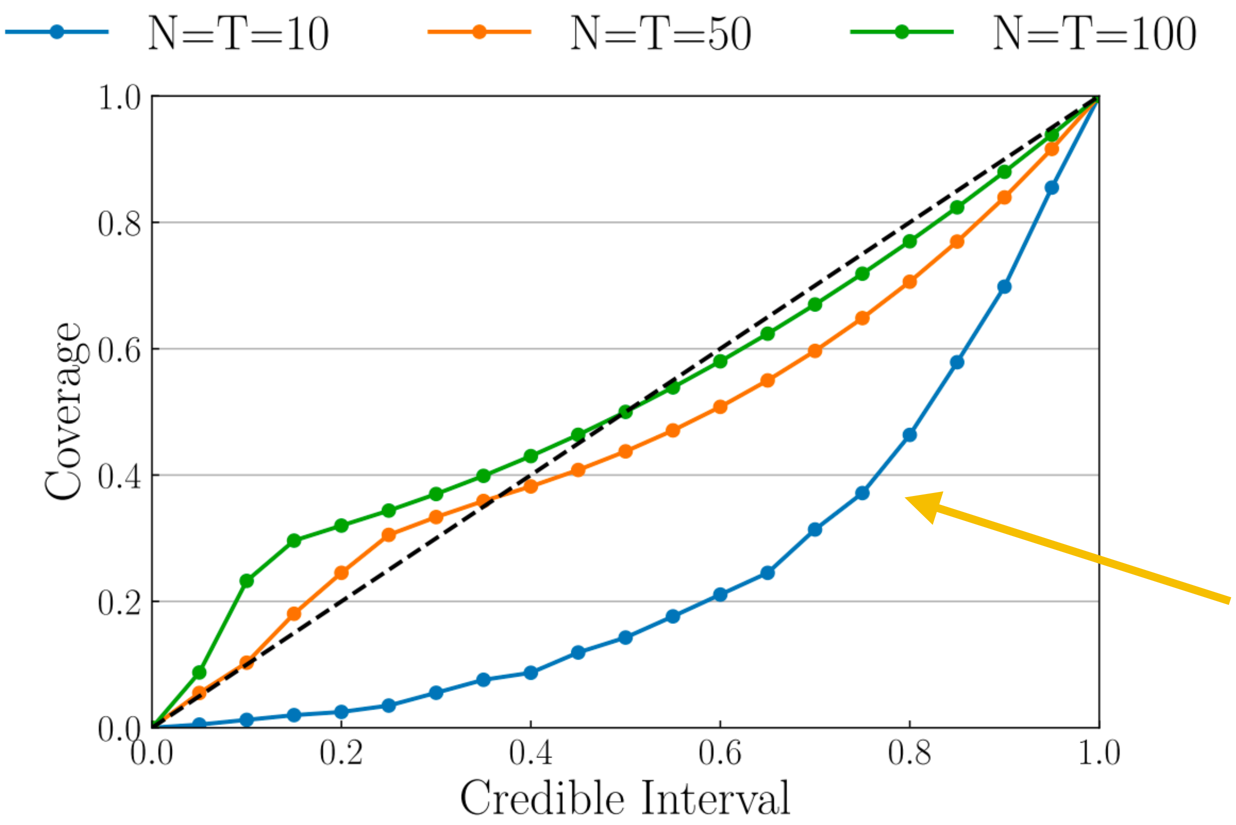


- This shows in our convergence rate…

$$\left\| \ldots \right\|_{L^2(\Theta)} \leq [\ldots] N^{-\frac{s_{\mathcal{X}}}{d}+\varepsilon} + [\ldots]$$

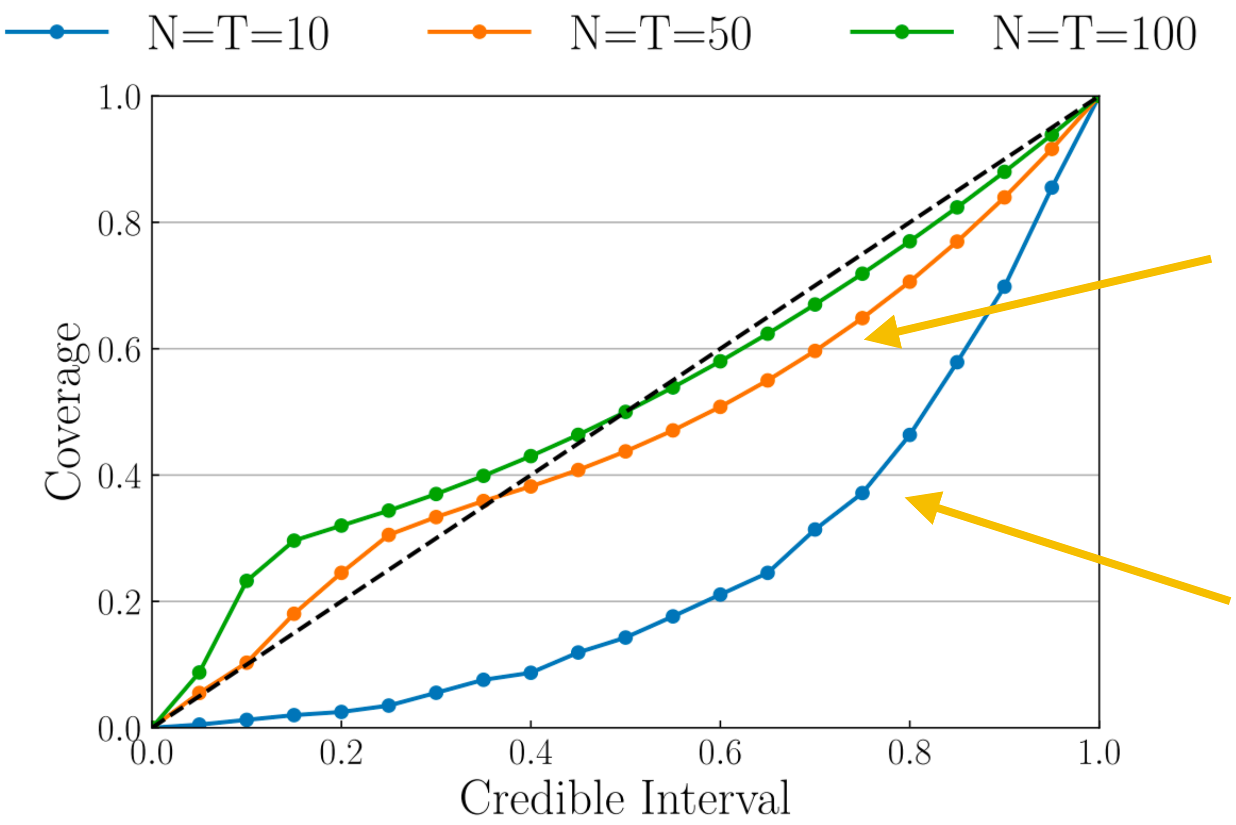
- It also rate bears out in practice

Calibration of the CBQ posterior (d=2)



- The CBQ posterior tends to be poorly calibrated when the number of data points is extremely small

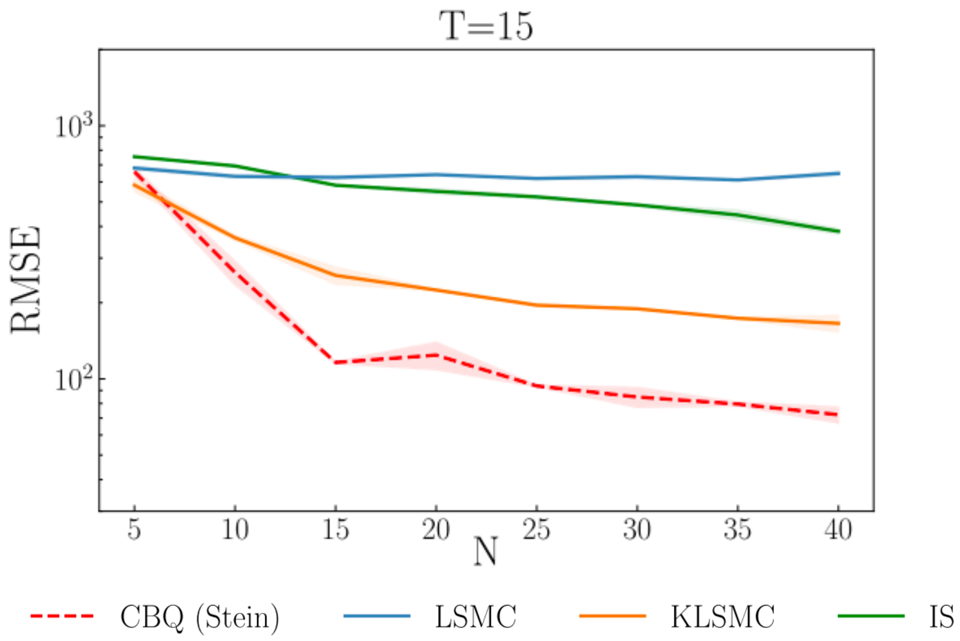
Calibration of the CBQ posterior (d=2)

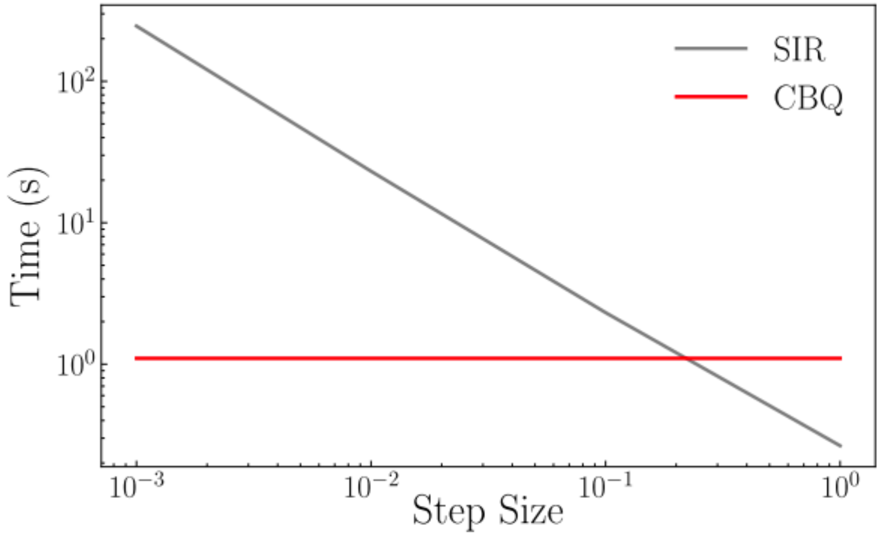


- But things get better for large N, T (although we didn't study this theoretically…)
- The CBQ posterior tends to be poorly calibrated when the number of data points is extremely small

Bayesian sensitivity analysis for SIR

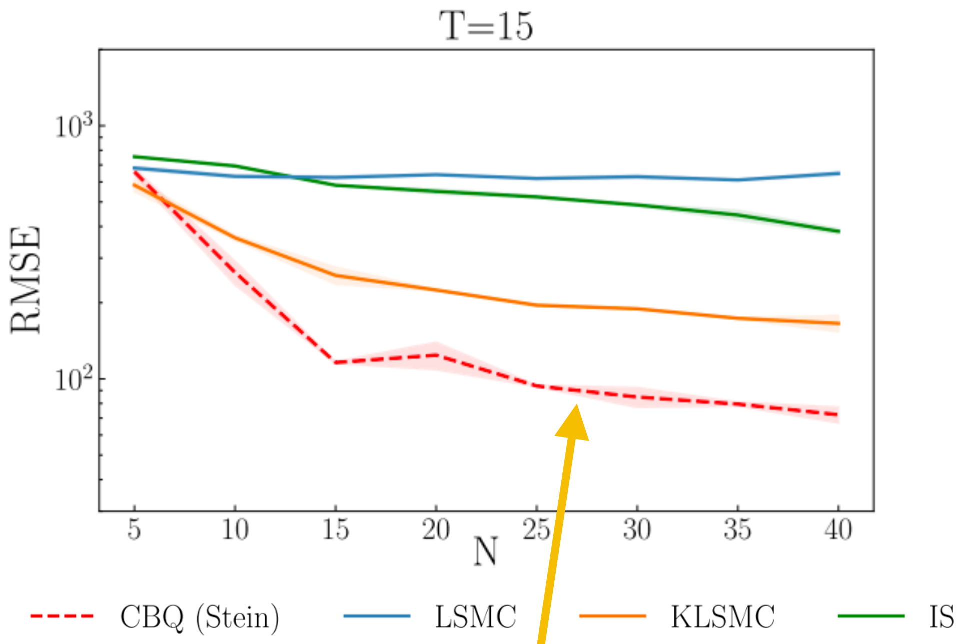
Setting: Bayesian sensitivity with Gamma$(\theta, 10)$ prior on infection rate.
QoI: Expected peak number of infected individuals over time period.

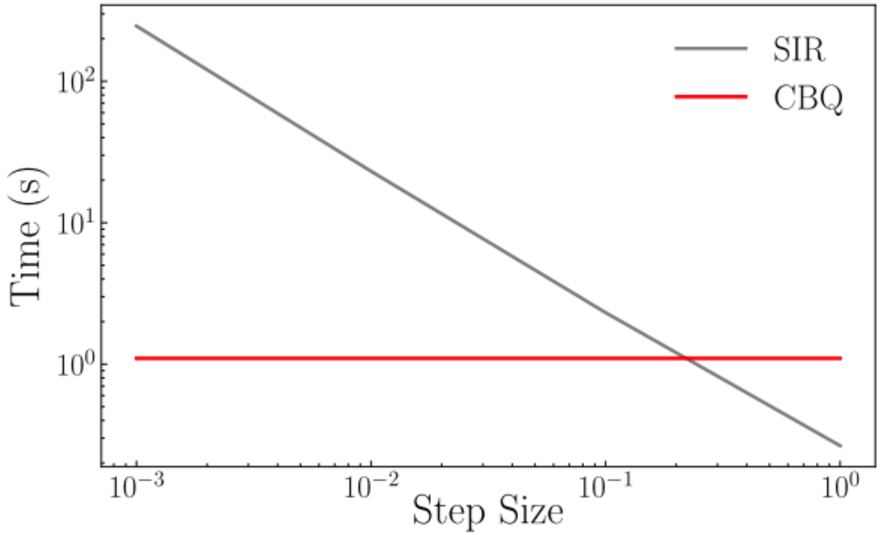




Bayesian sensitivity analysis for SIR

Setting: Bayesian sensitivity with Gamma$(\theta, 10)$ prior on infection rate.
QoI: Expected peak number of infected individuals over time period.

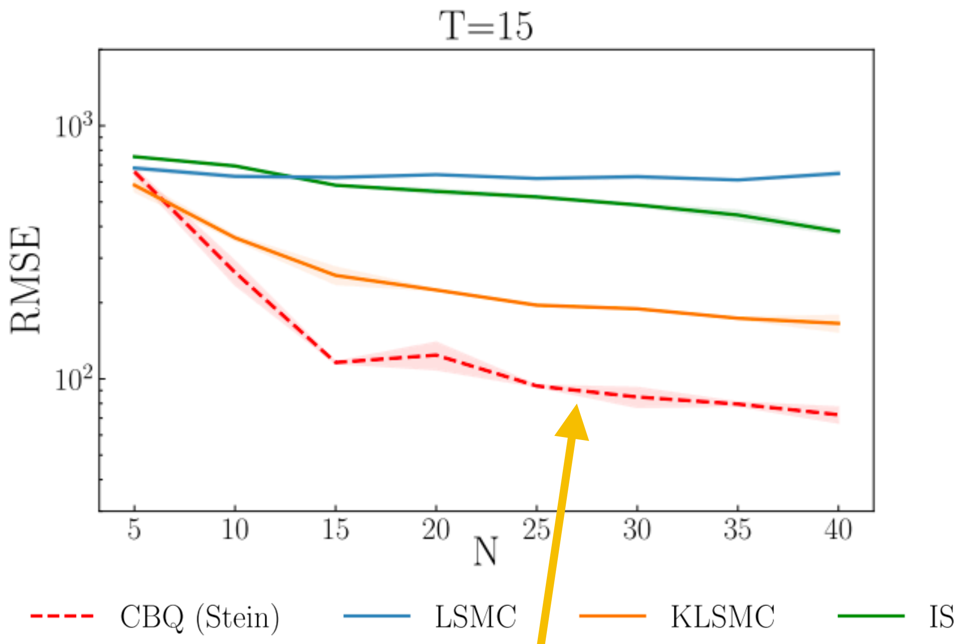


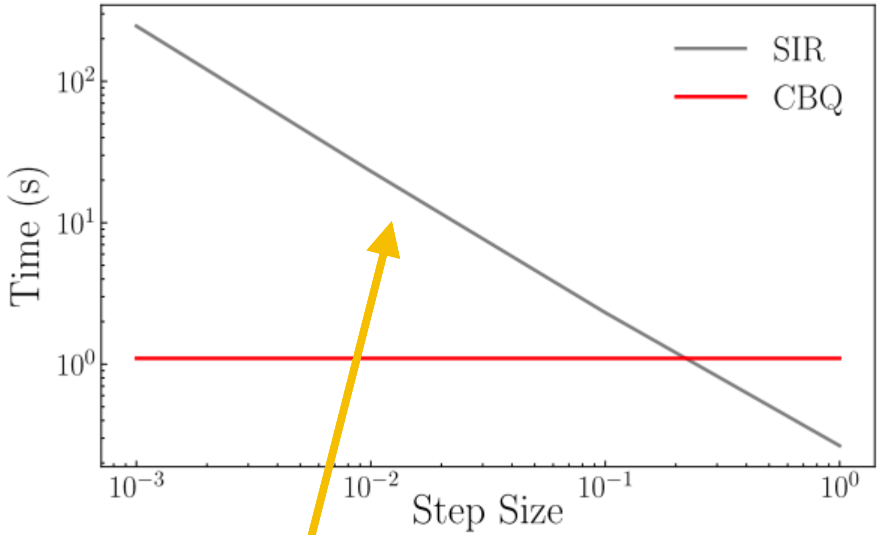


We get much faster convergence than alternatives!

Bayesian sensitivity analysis for SIR

Setting: Bayesian sensitivity with Gamma$(\theta, 10)$ prior on infection rate.
QoI: Expected peak number of infected individuals over time period.





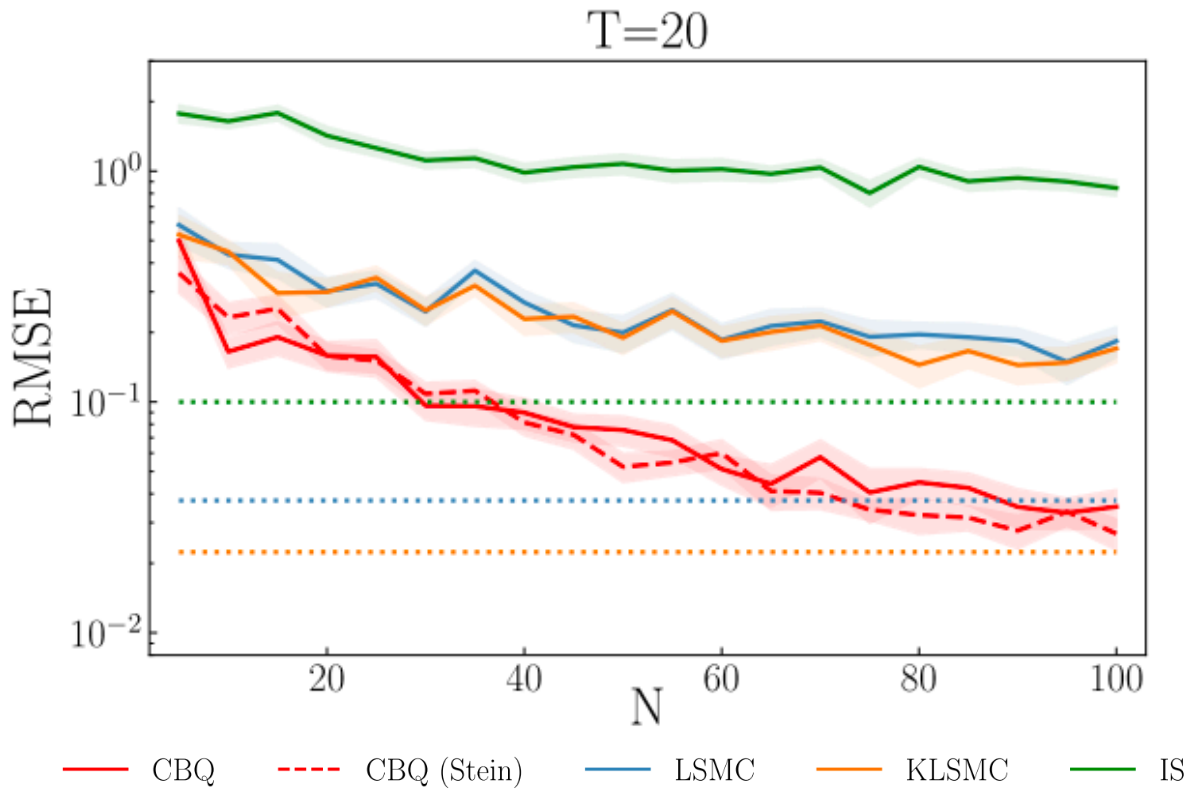
We get much faster convergence than alternatives!

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

Option pricing in finance

Setting: Pricing of butterfly call option using Black-Scholes formula.

QoI: Nested expectation representing expected loss.

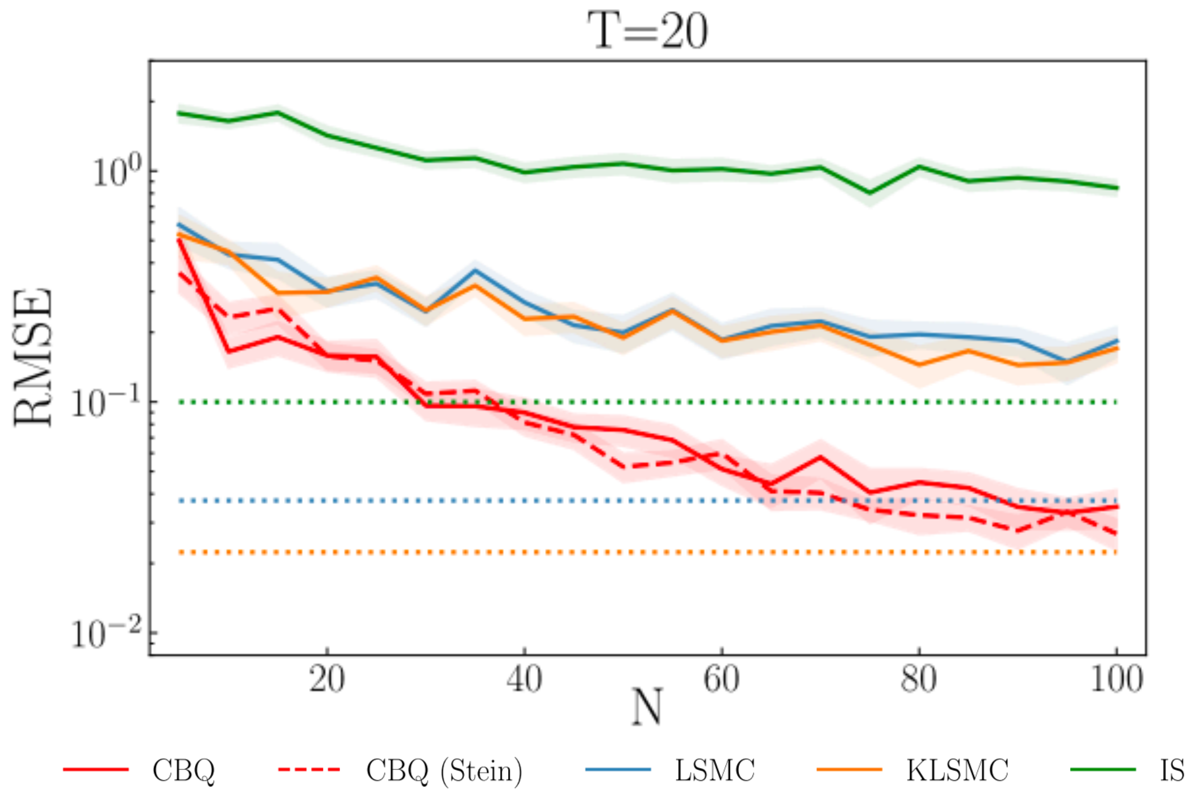


Option pricing in finance

Setting: Pricing of butterfly call option using Black-Scholes formula.

QoI: Nested expectation representing expected loss.

This specific problem can be solved in closed-form, but is representative of option pricing which usually requires **expensive simulations of SDEs**….



Option pricing in finance

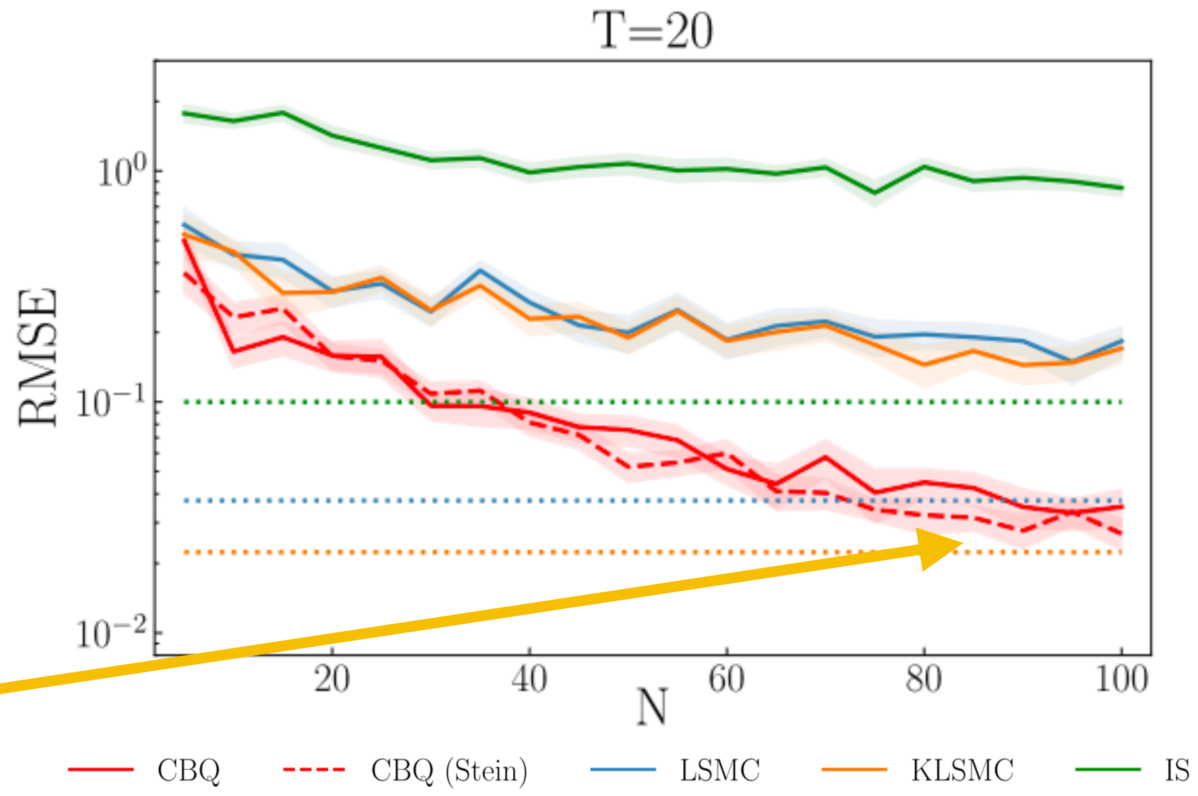
Setting: Pricing of butterfly call option using Black-Scholes formula.

QoI: Nested expectation representing expected loss.

This specific problem can be solved in closed-form, but is representative of option pricing which usually requires **expensive simulations of SDEs**….

CBQ significantly outperforms competitors!

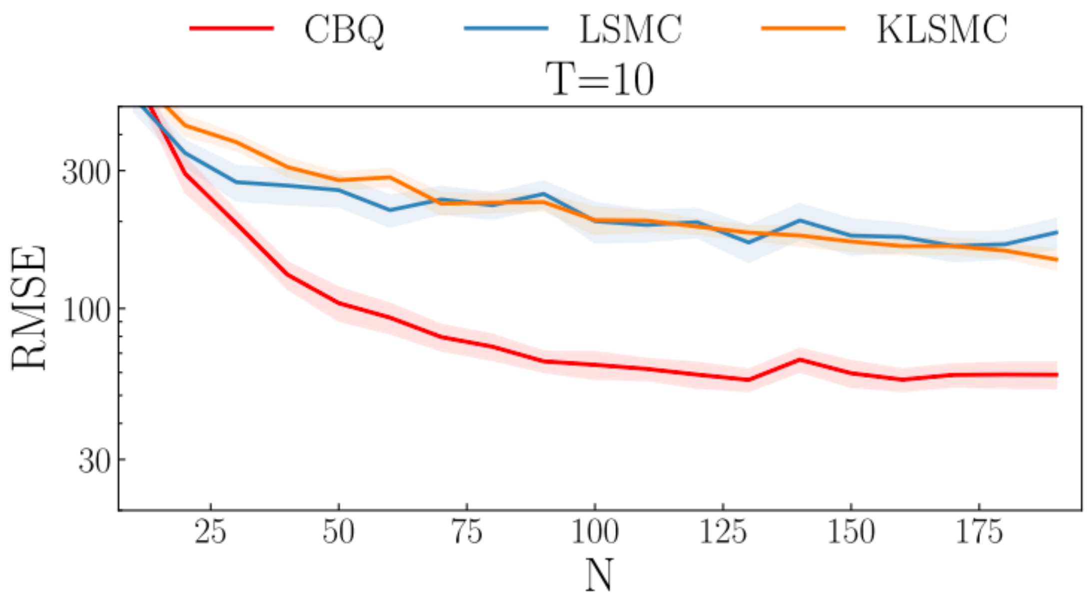
(Dotted lines is performance when $N = T = 1000$)



Health economics

Setting: Expected value of perfect information in Health economics.

QoI: Nested expectation representing expected value of collecting additional measurements from patients.

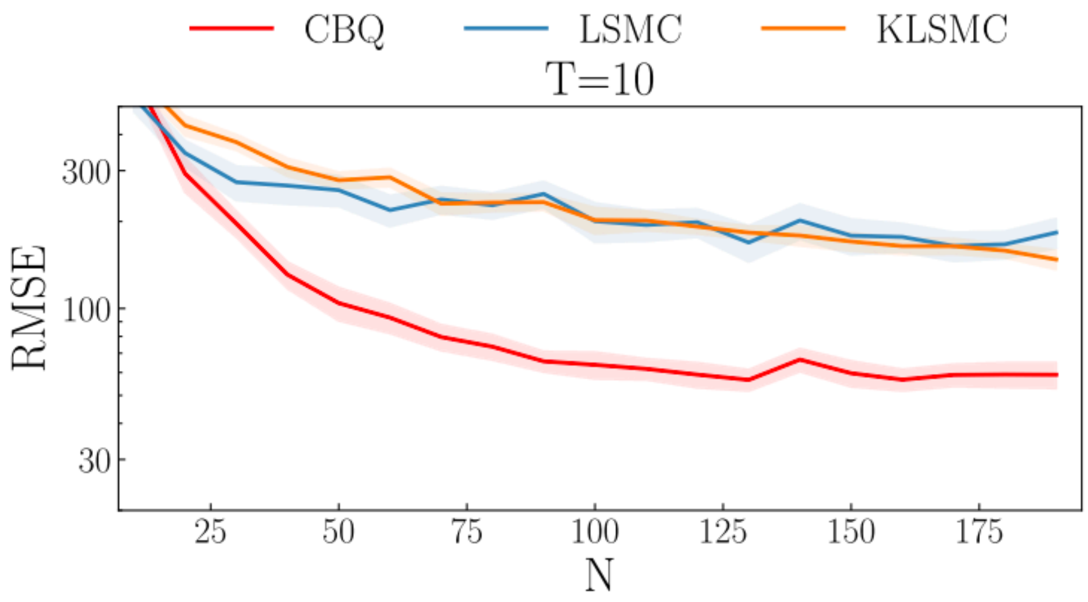


Health economics

Setting: Expected value of perfect information in Health economics.

QoI: Nested expectation representing expected value of collecting additional measurements from patients.

This experiment is toy, but is representative of a challenging computational problem where **each data point requires examining/testing a patient** (expensive!!)

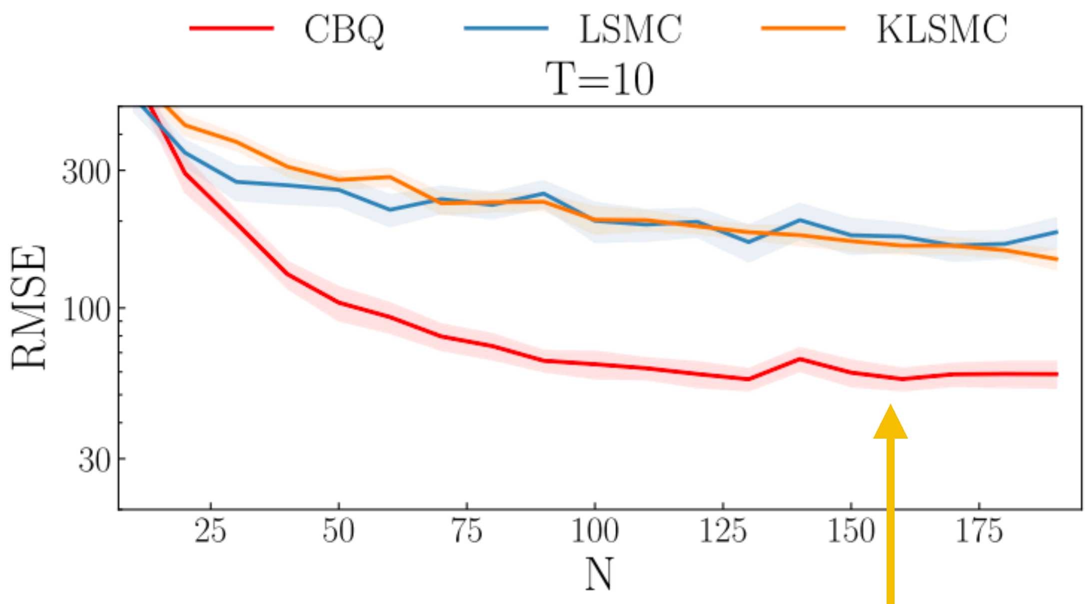


Health economics

Setting: Expected value of perfect information in Health economics.

QoI: Nested expectation representing expected value of collecting additional measurements from patients.

This experiment is toy, but is representative of a challenging computational problem where **each data point requires examining/testing a patient** (expensive!!)



Again much faster convergence! i.e. we need a lot less patients!

Conclusion and future work

Conclusion and future work

- We considered the problem of approximating parametric expectations and proposed a Bayesian algorithm to tackled this task, providing Bayesian UQ and a fast convergence rate.

Conclusion and future work

- We considered the problem of approximating parametric expectations and proposed a Bayesian algorithm to tackled this task, providing Bayesian UQ and a fast convergence rate.

- Plenty of work remaining including:
    - Lower bounds on the error.

Conclusion and future work

- We considered the problem of approximating parametric expectations and proposed a Bayesian algorithm to tackled this task, providing Bayesian UQ and a fast convergence rate.

- Plenty of work remaining including:
    - Lower bounds on the error.
    - Faster convergence in T, the number of tasks.

Conclusion and future work

- We considered the problem of approximating parametric expectations and proposed a Bayesian algorithm to tackled this task, providing Bayesian UQ and a fast convergence rate.

- Plenty of work remaining including:
    - Lower bounds on the error.
    - Faster convergence in T, the number of tasks.
    - Active learning for a task and across tasks.

Any Questions?

Conditional Bayesian Quadrature

Zonghao Chen[1,*] **Masha Naslidnyk**[1,*] **Arthur Gretton**[2] **François-Xavier Briol**[3]

[1]Department of Computer Science, University College London, London, UK
[2]Gatsby Computational Neuroscience Unit, University College London, London, UK
[3]Department of Statistical Science, University College London, London, UK

Recently appeared at **UAI 2024**!