Bayesian Quadrature for Multiple Related Integrals

François-Xavier Briol
University of Warwick (Department of Statistics)
Imperial College London (Department of Mathematics)

University of Sheffield
Machine Learning Seminar

arXiv:1801.04153

Collaborators

Xiaoyue Xi
(Warwick)

Mark Girolami
(Imperial & ATI)

Chris Oates
(Newcastle & ATI)

Michael Osborne
(Oxford)

Dino Sejdinovic
(Oxford & ATI)

Bayesian Numerical Methods

Bayesian Numerical Methods

- **Standard Numerical Analysis:** Study of how to best project continuous mathematical problems into discrete scales. (also thought of as the study of numerical errors).

- **Statistics:** Infer some quantity of interest (usually the parameter of a model) from data samples. Sounds familiar?

- **Bayesian Numerical Methods:** Perform Bayesian statistical inference on the solution of numerical problems.

[1] Larkin, F. M. (1972). Gaussian measure in Hilbert space and applications in numerical analysis. Rocky Mountain Journal of Mathematics, 2(3), 379-422.
[2] Diaconis, P. (1988). Bayesian Numerical Analysis. Statistical Decision Theory and Related Topics IV, 163-175.
[3] O'Hagan, A. (1992). Some Bayesian numerical analysis. Bayesian Statistics, 4, 345-363.
[4] Hennig, P., Osborne, M. A., & Girolami, M. (2015). Probabilistic Numerics and Uncertainty in Computations. J. Roy. Soc. A, 471(2179).

Bayesian Numerical Methods

- **Standard Numerical Analysis:** Study of how to best project continuous mathematical problems into discrete scales. (also thought of as the study of numerical errors).
- **Statistics:** Infer some quantity of interest (usually the parameter of a model) from data samples. Sounds familiar?
- **Bayesian Numerical Methods:** Perform Bayesian statistical inference on the solution of numerical problems.

[1] Larkin, F. M. (1972). Gaussian measure in Hilbert space and applications in numerical analysis. Rocky Mountain Journal of Mathematics, 2(3), 379-422.
[2] Diaconis, P. (1988). Bayesian Numerical Analysis. Statistical Decision Theory and Related Topics IV, 163-175.
[3] O'Hagan, A. (1992). Some Bayesian numerical analysis. Bayesian Statistics, 4, 345-363.
[4] Hennig, P., Osborne, M. A., & Girolami, M. (2015). Probabilistic Numerics and Uncertainty in Computations. J. Roy. Soc. A, 471(2179).

Bayesian Numerical Methods

- **Standard Numerical Analysis:** Study of how to best project continuous mathematical problems into discrete scales. (also thought of as the study of numerical errors).
- **Statistics:** Infer some quantity of interest (usually the parameter of a model) from data samples. Sounds familiar?
- **Bayesian Numerical Methods:** Perform Bayesian statistical inference on the solution of numerical problems.

[1] Larkin, F. M. (1972). Gaussian measure in Hilbert space and applications in numerical analysis. Rocky Mountain Journal of Mathematics, 2(3), 379-422.
[2] Diaconis, P. (1988). Bayesian Numerical Analysis. Statistical Decision Theory and Related Topics IV, 163-175.
[3] O'Hagan, A. (1992). Some Bayesian numerical analysis. Bayesian Statistics, 4, 345-363.
[4] Hennig, P., Osborne, M. A., & Girolami, M. (2015). Probabilistic Numerics and Uncertainty in Computations. J. Roy. Soc. A, 471(2179).

Bayesian Numerical Methods

- **Standard Numerical Analysis:** Study of how to best project continuous mathematical problems into discrete scales. (also thought of as the study of numerical errors).
- **Statistics:** Infer some quantity of interest (usually the parameter of a model) from data samples. Sounds familiar?
- **Bayesian Numerical Methods:** Perform Bayesian statistical inference on the solution of numerical problems.

[1] Larkin, F. M. (1972). Gaussian measure in Hilbert space and applications in numerical analysis. Rocky Mountain Journal of Mathematics, 2(3), 379-422.
[2] Diaconis, P. (1988). Bayesian Numerical Analysis. Statistical Decision Theory and Related Topics IV, 163-175.
[3] O'Hagan, A. (1992). Some Bayesian numerical analysis. Bayesian Statistics, 4, 345-363.
[4] Hennig, P., Osborne, M. A., & Girolami, M. (2015). Probabilistic Numerics and Uncertainty in Computations. J. Roy. Soc. A, 471(2179).

What is the Point?

- Quantification of the **epistemic uncertainty** associated with the numerical problem using **probability measures**, rather than worst-case bounds (not always representative of the actual error).
- Propagation of uncertainty through pipelines.
- Bayesian Numerical Methods can be framed as Bayesian Inverse Problems for the solution of the numerical problem.

[1] Cockayne, J., Oates, C., Sullivan, T., & Girolami, M. (2017). Bayesian Probabilistic Numerical Methods. arXiv:1701.04006.

What is the Point?

- Quantification of the **epistemic uncertainty** associated with the numerical problem using **probability measures**, rather than worst-case bounds (not always representative of the actual error).
- Propagation of uncertainty through pipelines.
- Bayesian Numerical Methods can be framed as Bayesian Inverse Problems for the solution of the numerical problem.

[1] Cockayne, J., Oates, C., Sullivan, T., & Girolami, M. (2017). Bayesian Probabilistic Numerical Methods. arXiv:1701.04006.

What is the Point?

- Quantification of the **epistemic uncertainty** associated with the numerical problem using **probability measures**, rather than worst-case bounds (not always representative of the actual error).
- Propagation of uncertainty through pipelines.
- Bayesian Numerical Methods can be framed as Bayesian Inverse Problems for the solution of the numerical problem.

[1] Cockayne, J., Oates, C., Sullivan, T., & Girolami, M. (2017). Bayesian Probabilistic Numerical Methods. arXiv:1701.04006.

Bayesian Numerical Integration

The Problem

- Let's come back to our problem of computing integrals! Consider a function $f : \mathcal{X} \to \mathbb{R}$ ($\mathcal{X} \subseteq \mathbb{R}^p$) assumed to be square-integrable and a probability measure Π.

$$\Pi[f] = \int_{\mathcal{X}} f(\mathbf{x}) \mathrm{d}\Pi(\boldsymbol{x}) \approx \sum_{i=1}^{n} w_i f(\boldsymbol{x}_i) = \hat{\Pi}[f]$$

where $\{\boldsymbol{x}_i\}_{i=1}^n \in \mathcal{X}$ & $\{w_i\}_{i=1}^n \in \mathbb{R}$.

- Examples include:
    1. **Monte Carlo (MC):** Sample $\{x_i\}_{i=1}^n \sim \Pi$ and let $w_i = 1/n \; \forall i$.
    2. **Markov Chain Monte Carlo (MCMC):** Sample states $\{x_i\}_{i=1}^n$ from a Markov Chain with invariant distribution Π and let $w_i = 1/n \; \forall i$.
    3. **Gaussian quadrature, importance sampling, QMC, SMC, etc...**

The Problem

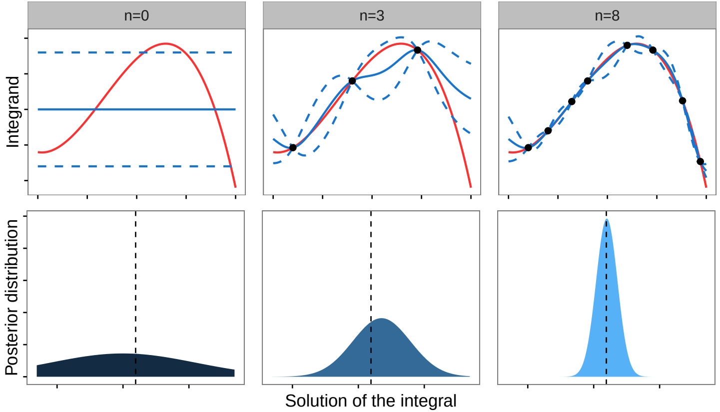
- Let's come back to our problem of computing integrals! Consider a function $f : \mathcal{X} \to \mathbb{R}$ ($\mathcal{X} \subseteq \mathbb{R}^p$) assumed to be square-integrable and a probability measure Π.

$$\Pi[f] = \int_{\mathcal{X}} f(\mathbf{x}) \mathrm{d}\Pi(\boldsymbol{x}) \approx \sum_{i=1}^{n} w_i f(\boldsymbol{x}_i) = \hat{\Pi}[f]$$

  where $\{\boldsymbol{x}_i\}_{i=1}^n \in \mathcal{X}$ & $\{w_i\}_{i=1}^n \in \mathbb{R}$.

- Examples include:
  1. **Monte Carlo (MC):** Sample $\{\boldsymbol{x}_i\}_{i=1}^n \sim \Pi$ and let $w_i = 1/n \ \forall i$.
  2. **Markov Chain Monte Carlo (MCMC):** Sample states $\{x_i\}_{i=1}^n$ from a Markov Chain with invariant distribution Π and let $w_i = 1/n \ \forall i$.
  3. **Gaussian quadrature, importance sampling, QMC, SMC, etc...**

Sketch of Bayesian Quadrature



Bayesian Quadrature

1. Place a Gaussian Process prior (assumed w.l.o.g. to have zero mean).
2. Evaluate the integrand f at several locations $\{x_i\}_{i=1}^n$ on $\mathcal{X}$. We get a Gaussian Process with mean and covariance function:

$$\begin{aligned} m_n(\boldsymbol{x}) &= k(\boldsymbol{x},\boldsymbol{X})k(\boldsymbol{X},\boldsymbol{X})^{-1}f(\boldsymbol{X}) \\ k_n(\boldsymbol{x},\boldsymbol{x}') &= k(\boldsymbol{x},\boldsymbol{x}') - k(\boldsymbol{x},\boldsymbol{X})k(\boldsymbol{X},\boldsymbol{X})^{-1}k(\boldsymbol{X},\boldsymbol{x}') \end{aligned}$$

3. Taking the pushforward through the integral operator, we get:

$$\begin{aligned} \mathbb{E}_n[\Pi[f]] &= \hat{\Pi}_{\mathrm{BQ}}[f] := \Pi[k(\cdot,\boldsymbol{X})]k(\boldsymbol{X},\boldsymbol{X})^{-1}f(\boldsymbol{X}) \\ \mathbb{V}_n[\Pi[f]] &= \Pi\bar{\Pi}[k] - \Pi[k(\cdot,\boldsymbol{X})]k(\boldsymbol{X},\boldsymbol{X})^{-1}\Pi[k(\boldsymbol{X},\cdot)]. \end{aligned}$$

[1] Larkin, F. M. (1972). Gaussian measure in Hilbert space and applications in numerical analysis. Rocky Mountain Journal of Mathematics, 2(3), 379422.
[2] Diaconis, P. (1988). Bayesian Numerical Analysis. Statistical Decision Theory and Related Topics IV, 163175.
[3] OHagan, A. (1991). Bayes-Hermite quadrature. Journal of Statistical Planning and Inference, 29, 245-260.

Bayesian Quadrature

1. Place a Gaussian Process prior (assumed w.l.o.g. to have zero mean).
2. Evaluate the integrand f at several locations $\{x_i\}_{i=1}^n$ on $\mathcal{X}$. We get a Gaussian Process with mean and covariance function:

$$\begin{aligned} m_n(\boldsymbol{x}) &= k(\boldsymbol{x}, \boldsymbol{X})k(\boldsymbol{X}, \boldsymbol{X})^{-1}f(\boldsymbol{X}) \\ k_n(\boldsymbol{x}, \boldsymbol{x}') &= k(\boldsymbol{x}, \boldsymbol{x}') - k(\boldsymbol{x}, \boldsymbol{X})k(\boldsymbol{X}, \boldsymbol{X})^{-1}k(\boldsymbol{X}, \boldsymbol{x}') \end{aligned}$$

3. Taking the pushforward through the integral operator, we get:

$$\begin{aligned} \mathbb{E}_n[\Pi[f]] &= \hat{\Pi}_{BQ}[f] := \Pi[k(\cdot, \boldsymbol{X})]k(\boldsymbol{X}, \boldsymbol{X})^{-1}f(\boldsymbol{X}) \\ \mathbb{V}_n[\Pi[f]] &= \Pi\bar{\Pi}[k] - \Pi[k(\cdot, \boldsymbol{X})]k(\boldsymbol{X}, \boldsymbol{X})^{-1}\Pi[k(\boldsymbol{X}, \cdot)]. \end{aligned}$$

[1] Larkin, F. M. (1972). Gaussian measure in Hilbert space and applications in numerical analysis. Rocky Mountain Journal of Mathematics, 2(3), 379422.
[2] Diaconis, P. (1988). Bayesian Numerical Analysis. Statistical Decision Theory and Related Topics IV, 163175.
[3] OHagan, A. (1991). Bayes-Hermite quadrature. Journal of Statistical Planning and Inference, 29, 245-260.

Bayesian Quadrature

1. Place a Gaussian Process prior (assumed w.l.o.g. to have zero mean).
2. Evaluate the integrand f at several locations $\{x_i\}_{i=1}^n$ on $\mathcal{X}$. We get a Gaussian Process with mean and covariance function:

$$\begin{aligned} m_n(\boldsymbol{x}) &= k(\boldsymbol{x}, \boldsymbol{X})k(\boldsymbol{X}, \boldsymbol{X})^{-1}f(\boldsymbol{X}) \\ k_n(\boldsymbol{x}, \boldsymbol{x}') &= k(\boldsymbol{x}, \boldsymbol{x}') - k(\boldsymbol{x}, \boldsymbol{X})k(\boldsymbol{X}, \boldsymbol{X})^{-1}k(\boldsymbol{X}, \boldsymbol{x}') \end{aligned}$$

3. Taking the pushforward through the integral operator, we get:

$$\begin{aligned} \mathbb{E}_n[\Pi[f]] &= \hat{\Pi}_{\mathrm{BQ}}[f] := \Pi[k(\cdot, \boldsymbol{X})]k(\boldsymbol{X}, \boldsymbol{X})^{-1}f(\boldsymbol{X}) \\ \mathbb{V}_n[\Pi[f]] &= \Pi\bar{\Pi}[k] - \Pi[k(\cdot, \boldsymbol{X})]k(\boldsymbol{X}, \boldsymbol{X})^{-1}\Pi[k(\boldsymbol{X}, \cdot)]. \end{aligned}$$

[1] Larkin, F. M. (1972). Gaussian measure in Hilbert space and applications in numerical analysis. Rocky Mountain Journal of Mathematics, 2(3), 379422.
[2] Diaconis, P. (1988). Bayesian Numerical Analysis. Statistical Decision Theory and Related Topics IV, 163175.
[3] OHagan, A. (1991). Bayes-Hermite quadrature. Journal of Statistical Planning and Inference, 29, 245-260.

Important Points

- The probability measure represents our uncertainty about the value of $\Pi[f]$ due to the fact that we cant evaluate the integrand f everywhere.

- We have chosen to model f (and hence $\Pi[f]$) using a Gaussian measure. This is mostly for computational tractability, but isn't necessarily the right thing to do!

- Notice that the formulae below do not specify where to evaluate f, but provide a posterior given the points at which we have evaluated.

$$
\begin{aligned}
\mathbb{E}_n[\Pi[f]] &= \hat{\Pi}_{BQ}[f] := \Pi[k(\cdot, \boldsymbol{X})]k(\boldsymbol{X}, \boldsymbol{X})^{-1} f(\boldsymbol{X}) \\
\mathbb{V}_n[\Pi[f]] &= \Pi\bar{\Pi}[k] - \Pi[k(\cdot, \boldsymbol{X})]k(\boldsymbol{X}, \boldsymbol{X})^{-1}\Pi[k(\boldsymbol{X}, \cdot)].
\end{aligned}
$$

We have the freedom to combine these with MC, MCMC, QMC, etc...

Important Points

- The probability measure represents our uncertainty about the value of $\Pi[f]$ due to the fact that we cant evaluate the integrand f everywhere.
- We have chosen to model f (and hence $\Pi[f]$) using a Gaussian measure. This is mostly for computational tractability, but isn't necessarily the right thing to do!
- Notice that the formulae below do not specify where to evaluate f, but provide a posterior given the points at which we have evaluated.

$$
\begin{aligned}
\mathbb{E}_n[\Pi[f]] &= \hat{\Pi}_{\mathrm{BQ}}[f] := \Pi[k(\cdot,\boldsymbol{X})]k(\boldsymbol{X},\boldsymbol{X})^{-1}f(\boldsymbol{X}) \\
\mathbb{V}_n[\Pi[f]] &= \Pi\bar{\Pi}[k] - \Pi[k(\cdot,\boldsymbol{X})]k(\boldsymbol{X},\boldsymbol{X})^{-1}\Pi[k(\boldsymbol{X},\cdot)].
\end{aligned}
$$

We have the freedom to combine these with MC, MCMC, QMC, etc...

Important Points

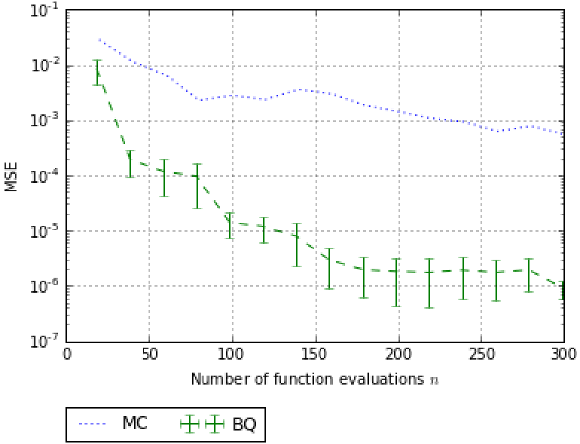
- The probability measure represents our uncertainty about the value of $\Pi[f]$ due to the fact that we cant evaluate the integrand f everywhere.
- We have chosen to model f (and hence $\Pi[f]$) using a Gaussian measure. This is mostly for computational tractability, but isn't necessarily the right thing to do!
- Notice that the formulae below do not specify where to evaluate f, but provide a posterior given the points at which we have evaluated.

$$\begin{aligned}
\mathbb{E}_n[\Pi[f]] &= \hat{\Pi}_{\mathrm{BQ}}[f] := \Pi[k(\cdot, \boldsymbol{X})]k(\boldsymbol{X}, \boldsymbol{X})^{-1} f(\boldsymbol{X}) \\
\mathbb{V}_n[\Pi[f]] &= \Pi\bar{\Pi}[k] - \Pi[k(\cdot, \boldsymbol{X})]k(\boldsymbol{X}, \boldsymbol{X})^{-1}\Pi[k(\boldsymbol{X}, \cdot)].
\end{aligned}$$

We have the freedom to combine these with MC, MCMC, QMC, etc...

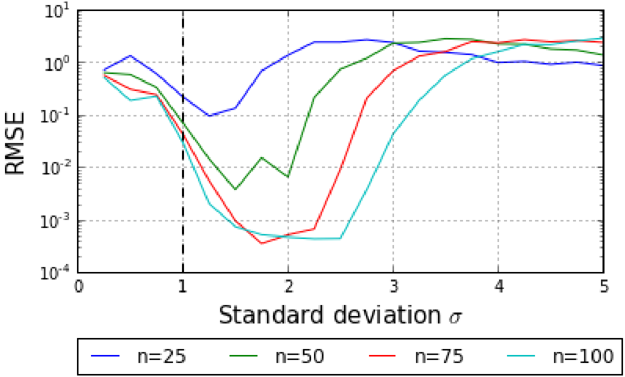
Toy example

Toy problem: $f(x) = \sin(x) + 1$ & Π is $\mathcal{N}(0, 1)$. We use the kernel $k(x, y) = \exp(-(x-y)^2/l^2)$ (with $l = 1$) and sample $\{x_i\}_{i=1}^n \sim \Pi$.



Toy example: The effect of sampling

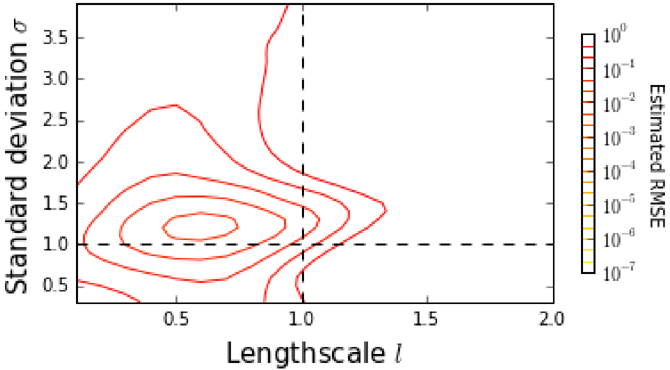
Toy problem: $f(x) = \sin(x) + 1$ & Π is $\mathcal{N}(0,1)$. We use the kernel $k(x,y) = \exp(-(x-y)^2/l^2)$ (with $l = 1$) and sample $\{x_i\}_{i=1}^n \sim \mathcal{N}(0,\sigma^2)$.



Toy example: The effect of sampling and kernel choice

Toy problem: $f(x) = \sin(x) + 1$ & Π is $\mathcal{N}(0, 1^2)$. We use the kernel $k(x,y) = \exp(-(x-y)^2/l^2)$ and sample from $\{x_i\}_{i=1}^n \sim \mathcal{N}(0, \sigma^2)$.

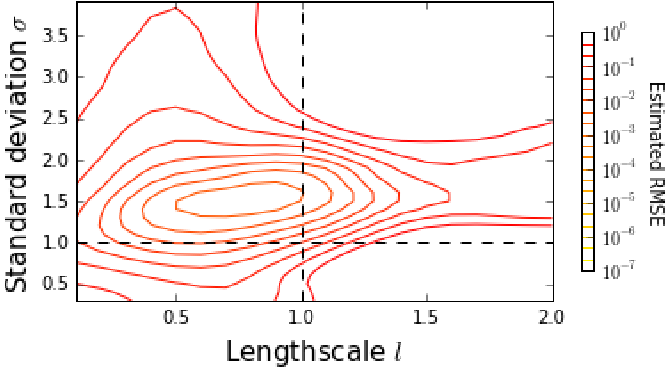
Number of samples: 25



Toy example: The effect of sampling and kernel choice

Toy problem: $f(x) = \sin(x) + 1$ & Π is $\mathcal{N}(0, 1^2)$. We use the kernel $k(x, y) = \exp(-(x-y)^2/l^2)$ and sample from $\{x_i\}_{i=1}^n \sim \mathcal{N}(0, \sigma^2)$.

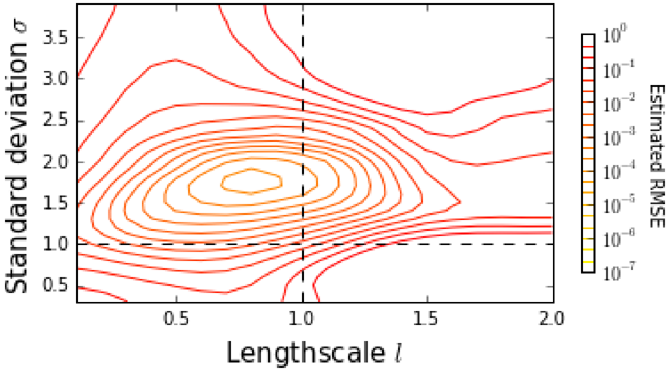
Number of samples: 50



Toy example: The effect of sampling and kernel choice

Toy problem: $f(x) = \sin(x) + 1$ & Π is $\mathcal{N}(0, 1^2)$. We use the kernel $k(x, y) = \exp(-(x-y)^2/l^2)$ and sample from $\{x_i\}_{i=1}^n \sim \mathcal{N}(0, \sigma^2)$.

Number of samples: 75



Calibration

Clearly, from the previous slide we can tell that hyperparameters will be of great importance. To do this we have several alternatives:

- Go full Bayesian and put a prior on the parameter, then look at the posterior predictive. **Problem:** We lose the conjugacy property and this becomes very expensive to do!
- Do some sort of cross validation. **Problem:** This is still expensive as requires solving many linear systems.
- Do some empirical Bayes, i.e. maximise the marginal likelihood of the data. **Problem:** Not very Bayesian.

In practice, we tend to use empirical Bayes for speed reason...

Calibration

Clearly, from the previous slide we can tell that hyperparameters will be of great importance. To do this we have several alternatives:

- Go full Bayesian and put a prior on the parameter, then look at the posterior predictive. **Problem:** We lose the conjugacy property and this becomes very expensive to do!
- Do some sort of cross validation. **Problem:** This is still expensive as requires solving many linear systems.
- Do some empirical Bayes, i.e. maximise the marginal likelihood of the data. **Problem:** Not very Bayesian.

In practice, we tend to use empirical Bayes for speed reason...

Calibration

Clearly, from the previous slide we can tell that hyperparameters will be of great importance. To do this we have several alternatives:

- Go full Bayesian and put a prior on the parameter, then look at the posterior predictive. **Problem:** We lose the conjugacy property and this becomes very expensive to do!
- Do some sort of cross validation. **Problem:** This is still expensive as requires solving many linear systems.
- Do some empirical Bayes, i.e. maximise the marginal likelihood of the data. **Problem:** Not very Bayesian.

In practice, we tend to use empirical Bayes for speed reason...

Calibration

Clearly, from the previous slide we can tell that hyperparameters will be of great importance. To do this we have several alternatives:

- Go full Bayesian and put a prior on the parameter, then look at the posterior predictive. **Problem:** We lose the conjugacy property and this becomes very expensive to do!
- Do some sort of cross validation. **Problem:** This is still expensive as requires solving many linear systems.
- Do some empirical Bayes, i.e. maximise the marginal likelihood of the data. **Problem:** Not very Bayesian.

In practice, we tend to use empirical Bayes for speed reason...

Convergence Results

So why does it converge fast?

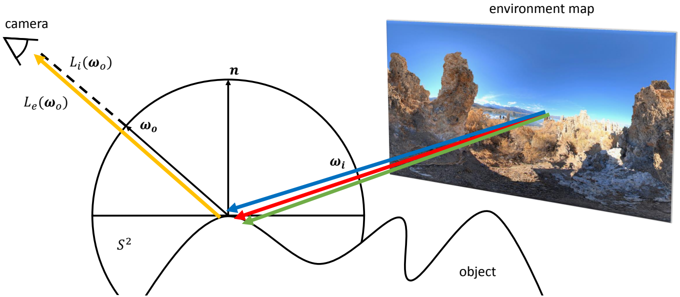
- For integration in an RKHS $\mathcal{H}$ (associated to the GP kernel k), the standard quantity to consider is the worst-case error:

$$e\left(\hat{\Pi};\Pi,\mathcal{H}\right) \;:=\; \sup_{\|f\|_{\mathcal{H}}\leq 1}\left|\Pi[f]-\hat{\Pi}[f]\right|$$

- One can show that BQ attains optimal rates of convergence for certain classes of smooth functions. In particular, with i.i.d points, one can get $\mathcal{O}(n^{-\frac{\alpha}{d}+\epsilon})$ for spaces of smoothness α and $\mathcal{O}(\exp(-Cn^{\frac{1}{d}-\epsilon}))$ for infinitely smooth functions.

Briol, F.-X., Oates, C. J., Girolami, M., Osborne, M. A., & Sejdinovic, D. (2015). Probabilistic Integration: A Role for Statisticians in Numerical Analysis?, *arXiv:1512.00933*.

Application: Global Illumination

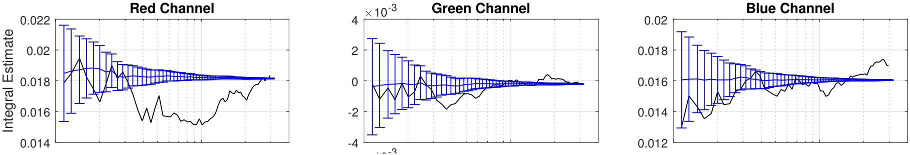


We need to compute three integrals at each pixel:

$$L_o(\omega_o) = L_e(w_o) + \int_{\mathbb{S}^2} L_i(\omega_i)\rho(\omega_i, \omega_o)[\omega_i \cdot n]_+ \mathrm{d}\sigma(\omega_i)$$

[1] Marques, R., Bouville, C., Ribardiere, M., Santos, P., & Bouatouch, K. (2013). A spherical Gaussian framework for Bayesian Monte Carlo rendering of glossy surfaces. IEEE Transactions on Visualization and Computer Graphics, 19(10), 1619-1632.

Application: Global Illumination

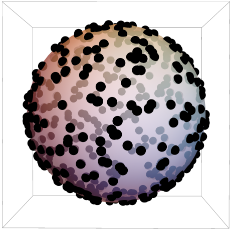


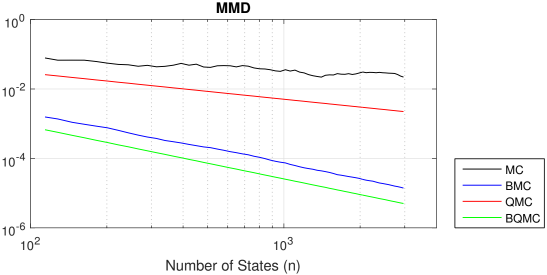
- The kernel used gives an RKHS norm-equivalent to a Sobolev space of smoothness $\frac{3}{2}$:

$$k(\boldsymbol{x}, \boldsymbol{x}') = \frac{8}{3} - \|\boldsymbol{x} - \boldsymbol{x}'\|_2 \text{ for all } \boldsymbol{x}, \boldsymbol{x}' \in \mathbb{S}^2.$$

- We can show a convergence rate of $e(\hat{\Pi}_{\mathrm{BMC}}; \Pi, \mathcal{H}) = \mathcal{O}_P(n^{-\frac{3}{4}})$ which is **optimal** for this space!

Application: Global Illumination





Spreading the points **and** re-weighting can help significantly!

[1] Briol, F.-X., Oates, C. J., Girolami, M., & Osborne, M. A. (2015). Frank-Wolfe Bayesian Quadrature: Probabilistic Integration with Theoretical Guarantees. In Advances In Neural Information Processing Systems 28 (pp. 1162–1170).
[2] Briol, F.-X., Oates, C. J., Cockayne, J., Chen, W. Y., & Girolami, M. (2017). On the Sampling Problem for Kernel Quadrature. In Proceedings of the 34th International Conference on Machine Learning (pp. 586–595).

Bayesian Quadrature for Multiple Integrals

What About Multiple Integrals?

- In the example, we actually need to approximate thousands of integrals for each frame of a virtual environment... **This is slow and expensive!**
- We can formalise the process above as that of finding the integral of a set of functions $f_1, \dots, f_D$ against some measure Π. But what if we know something about how f_1 relates to f_2, etc...?
- It might make more sense to approximate the integral with a quadrature rule of the form:

$$\hat{\Pi}[f_d] = \sum_{d'=1}^{D} \sum_{i=1}^{n} (\boldsymbol{W}_i)_{dd'} f_{d'}(\boldsymbol{x}_{d'i})$$

where the weights would encode the correlation across functions.

What About Multiple Integrals?

- In the example, we actually need to approximate thousands of integrals for each frame of a virtual environment... **This is slow and expensive!**
- We can formalise the process above as that of finding the integral of a set of functions $f_1, \dots, f_D$ against some measure Π. But what if we know something about how f_1 relates to f_2, etc...?
- It might make more sense to approximate the integral with a quadrature rule of the form:

$$\hat{\Pi}[f_d] \quad = \quad \sum_{d'=1}^{D} \sum_{i=1}^{n} (\boldsymbol{W}_i)_{dd'} f_{d'}(\boldsymbol{x}_{d'i})$$

where the weights would encode the correlation across functions.

What About Multiple Integrals?

- In the example, we actually need to approximate thousands of integrals for each frame of a virtual environment... **This is slow and expensive!**

- We can formalise the process above as that of finding the integral of a set of functions $f_1, \dots, f_D$ against some measure Π. But what if we know something about how f_1 relates to f_2, etc...?

- It might make more sense to approximate the integral with a quadrature rule of the form:

$$\hat{\Pi}[f_d] \quad = \quad \sum_{d'=1}^{D} \sum_{i=1}^{n} (\boldsymbol{W}_i)_{dd'} f_{d'}(\boldsymbol{x}_{d'i})$$

where the weights would encode the correlation across functions.

Bayesian Quadrature for Multiple Related Functions

- We can use the same type of results for Gaussian Processes on the extended space of vector-valued functions $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}^D$ (rather than $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}$) where $\boldsymbol{f}(x) = (f_1(x), \ldots, f_D(x))$.
- This approach allow us to directly encode the relation between each function f_i by specifying the kernel K.
- In this case the posterior distribution is a $\mathcal{GP}(\boldsymbol{m}_n, \boldsymbol{K}_n)$ with vector-valued mean $\boldsymbol{m}_n : \mathcal{X} \to \mathbb{R}^D$ and matrix-valued covariance $\boldsymbol{K}_n : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{D \times D}$:

$$\begin{aligned} \boldsymbol{m}_n(\boldsymbol{x}) &= \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{X})\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})^{-1}\boldsymbol{f}(\boldsymbol{X}) \\ \boldsymbol{K}_n(\boldsymbol{x}, \boldsymbol{x}') &= \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{X})\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})^{-1}\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{x}'). \end{aligned}$$

The overall cost for computing this is $\mathcal{O}(n^3 D^3)$.

Alvarez, M. A., Rosasco, L., & Lawrence, N. D. (2012). Kernels for vector-valued functions: A review. Foundations and Trends in Machine Learning, 4(3), 195-266.

Bayesian Quadrature for Multiple Related Functions

- We can use the same type of results for Gaussian Processes on the extended space of vector-valued functions $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}^D$ (rather than $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}$) where $\boldsymbol{f}(x) = (f_1(x), \ldots, f_D(x))$.
- This approach allow us to directly encode the relation between each function f_i by specifying the kernel K.
- In this case the posterior distribution is a $\mathcal{GP}(\boldsymbol{m}_n, \boldsymbol{K}_n)$ with vector-valued mean $\boldsymbol{m}_n : \mathcal{X} \to \mathbb{R}^D$ and matrix-valued covariance $\boldsymbol{K}_n : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{D \times D}$:

$$\begin{aligned}\boldsymbol{m}_n(\boldsymbol{x}) &= \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{X})\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})^{-1}\boldsymbol{f}(\boldsymbol{X}) \\ \boldsymbol{K}_n(\boldsymbol{x}, \boldsymbol{x}') &= \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{X})\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})^{-1}\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{x}').\end{aligned}$$

The overall cost for computing this is $\mathcal{O}(n^3 D^3)$.

Alvarez, M. A., Rosasco, L., & Lawrence, N. D. (2012). Kernels for vector-valued functions: A review. Foundations and Trends in Machine Learning, 4(3), 195-266.

- Consider multi-output Bayesian Quadrature with a $\mathcal{GP}(\mathbf{0}, \boldsymbol{K})$ prior on $\boldsymbol{f} = (f_1, \ldots, f_D)^\top$. The posterior distribution on $\Pi[\boldsymbol{f}]$ is a D-dimensional Gaussian with mean and covariance matrix:

$$
\begin{aligned}
\mathbb{E}_N\left[\Pi[\boldsymbol{f}]\right] &= \Pi[\boldsymbol{K}(\cdot, \boldsymbol{X})]\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})^{-1}\boldsymbol{f}(\boldsymbol{X}) \\
\mathbb{V}_N\left[\Pi[\boldsymbol{f}]\right] &= \Pi\bar{\Pi}\left[\boldsymbol{K}\right] - \Pi[\boldsymbol{K}(\cdot, \boldsymbol{X})]\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})^{-1}\bar{\Pi}[\boldsymbol{K}(\boldsymbol{X}, \cdot)]
\end{aligned}
$$

- Kernel evaluations are now matrix-valued (i.e. in $\mathbb{R}^{D \times D}$) as opposed to scalar-valued. A simple example is the following separable kernel:

$$
\boldsymbol{K}(x, x') = \boldsymbol{B}k(x, x')
$$

B encodes the covariance between function, and k the type of function in each of the components.

- In this case, we can reduce the cost to $\mathcal{O}(n^3 + D^3)$.

- Consider multi-output Bayesian Quadrature with a $\mathcal{GP}(\mathbf{0}, \boldsymbol{K})$ prior on $\boldsymbol{f} = (f_1, \ldots, f_D)^\top$. The posterior distribution on $\Pi[\boldsymbol{f}]$ is a D-dimensional Gaussian with mean and covariance matrix:

$$\begin{aligned}
\mathbb{E}_N\left[\Pi[\boldsymbol{f}]\right] &= \Pi[\boldsymbol{K}(\cdot, \boldsymbol{X})]\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})^{-1}\boldsymbol{f}(\boldsymbol{X}) \\
\mathbb{V}_N\left[\Pi[\boldsymbol{f}]\right] &= \Pi\bar{\Pi}\left[\boldsymbol{K}\right] - \Pi[\boldsymbol{K}(\cdot, \boldsymbol{X})]\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})^{-1}\bar{\Pi}[\boldsymbol{K}(\boldsymbol{X}, \cdot)]
\end{aligned}$$

- Kernel evaluations are now matrix-valued (i.e. in $\mathbb{R}^{D\times D}$) as opposed to scalar-valued. A simple example is the following separable kernel:

$$\boldsymbol{K}(x, x') = \boldsymbol{B}k(x, x')$$

  B encodes the covariance between function, and k the type of function in each of the components.

- In this case, we can reduce the cost to $\mathcal{O}(n^3 + D^3)$.

Some Theory for Multi-output BQ

Define the individual worst-case errors:

$$e(\hat{\Pi}; \Pi, \mathcal{H}_{\boldsymbol{K}}, d) \quad = \quad \sup_{\|\boldsymbol{f}\|_{\boldsymbol{K}} \leq 1} \left| \Pi[f_d] - \hat{\Pi}[f_d] \right|$$

Theorem (Convergence rate for BQ with separable kernel)

Suppose we want to approximate $\Pi[\boldsymbol{f}]$ *for some* $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}^D$ *and* $\hat{\Pi}_{BQ}[\boldsymbol{f}]$ *is the multi-output BQ rule with the kernel* $\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{B} k(\boldsymbol{x}, \boldsymbol{x}')$ *for some positive definite* $\boldsymbol{B} \in \mathbb{R}^{D \times D}$ *and scalar-valued kernel* $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ *where all functions are evaluated on a point set* $\{x_i\}_{i=1}^n$.
Then, $\forall d = 1, \ldots, D$, *we have:*

$$e(\hat{\Pi}_{BQ}; \Pi, \mathcal{H}_{\boldsymbol{K}}, d) \quad = \quad \mathcal{O}\left(e(\hat{\Pi}_{BQ}; \Pi, \mathcal{H}_k) \right)$$

Some Theory for Multi-output BQ

Assume that $\mathcal{X} \subset \mathbb{R}^p$ is a domain with Lipschitz boundary and satisfying an interior cone condition. Furthermore, assume the $\{x_i\}_{i=1}^n$ are either:
(A1) IID samples from some distribution Π' with density $\pi' > 0$ on $\mathcal{X}$, or
(A2) obtained from a quasi-uniform grid on $\mathcal{X}$.

- If $\mathcal{H}_k$ is norm-equivalent to an RKHS with Matérn kernel of smoothness $\alpha > \frac{p}{2}$, we have $\forall d = 1, \ldots, D$:

$$e(\mathcal{H}_{\boldsymbol{K}}, \hat{\Pi}_{\mathrm{BQ}}, \boldsymbol{X}, d) \quad = \quad \mathcal{O}\left(n^{-\frac{\alpha}{p}+\epsilon}\right).$$

  for $\epsilon > 0$ arbitrarily small.
- If $\mathcal{H}_k$ is norm-equivalent to the RKHS with squared-exponential, multiquadric or inverse multiquadric kernel, we have $\forall d = 1, \ldots, D$:

$$e(\mathcal{H}_{\boldsymbol{K}}, \hat{\Pi}_{\mathrm{BQ}}, \boldsymbol{X}, d) \quad = \quad \mathcal{O}\left(\exp\left(-C_1 n^{\frac{1}{p}-\epsilon}\right)\right).$$

  for some $C_1 > 0$ and $\epsilon > 0$ arbitrarily small.

Some Theory for Multi-output BQ

Assume that $\mathcal{X} \subset \mathbb{R}^p$ is a domain with Lipschitz boundary and satisfying an interior cone condition. Furthermore, assume the $\{x_i\}_{i=1}^n$ are either:
(A1) IID samples from some distribution Π' with density $\pi' > 0$ on $\mathcal{X}$, or
(A2) obtained from a quasi-uniform grid on $\mathcal{X}$.

- If $\mathcal{H}_k$ is norm-equivalent to an RKHS with Matérn kernel of smoothness $\alpha > \frac{p}{2}$, we have $\forall d = 1, \ldots, D$:

$$e(\mathcal{H}_{\boldsymbol{K}}, \hat{\Pi}_{\mathrm{BQ}}, \boldsymbol{X}, d) \;=\; \mathcal{O}\left(n^{-\frac{\alpha}{p}+\epsilon}\right).$$

  for $\epsilon > 0$ arbitrarily small.

- If $\mathcal{H}_k$ is norm-equivalent to the RKHS with squared-exponential, multiquadric or inverse multiquadric kernel, we have $\forall d = 1, \ldots, D$:

$$e(\mathcal{H}_{\boldsymbol{K}}, \hat{\Pi}_{\mathrm{BQ}}, \boldsymbol{X}, d) \;=\; \mathcal{O}\left(\exp\left(-C_1 n^{\frac{1}{p}-\epsilon}\right)\right).$$

  for some $C_1 > 0$ and $\epsilon > 0$ arbitrarily small.

Some Theory for Multi-output BQ

Assume that $\mathcal{X} \subset \mathbb{R}^p$ is a domain with Lipschitz boundary and satisfying an interior cone condition. Furthermore, assume the $\{x_i\}_{i=1}^n$ are either:
(A1) IID samples from some distribution Π' with density $\pi' > 0$ on $\mathcal{X}$, or
(A2) obtained from a quasi-uniform grid on $\mathcal{X}$.

- If $\mathcal{H}_k$ is norm-equivalent to an RKHS with Matérn kernel of smoothness $\alpha > \frac{p}{2}$, we have $\forall d = 1, \ldots, D$:

$$e(\mathcal{H}_{\boldsymbol{K}}, \hat{\Pi}_{\mathrm{BQ}}, \boldsymbol{X}, d) \quad = \quad \mathcal{O}\left(n^{-\frac{\alpha}{p}+\epsilon}\right).$$

  for $\epsilon > 0$ arbitrarily small.

- If $\mathcal{H}_k$ is norm-equivalent to the RKHS with squared-exponential, multiquadric or inverse multiquadric kernel, we have $\forall d = 1, \ldots, D$:

$$e(\mathcal{H}_{\boldsymbol{K}}, \hat{\Pi}_{\mathrm{BQ}}, \boldsymbol{X}, d) \quad = \quad \mathcal{O}\left(\exp\left(-C_1 n^{\frac{1}{p}-\epsilon}\right)\right).$$

  for some $C_1 > 0$ and $\epsilon > 0$ arbitrarily small.

Theory in the Misspecified Case

Theorem (Misspecified Convergence Result for Separable Kernel)

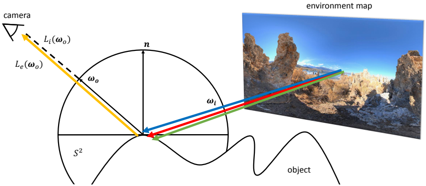
Let k_α be a kernel norm-equivalent to a Matérn kernel on some domain $\mathcal{X}$ with Lipschitz boundary and interior cone condition and consider the BQ rule $\hat{\Pi}_{BQ}[\boldsymbol{f}]$ corresponding to a separable kernel $\boldsymbol{K}_\alpha(x,x') = \boldsymbol{B} k_\alpha(x,x')$.

Suppose $\{x_i\}_{i=1}^n$ satisfies (A_2), and $\boldsymbol{f} \in \mathcal{H}_{\boldsymbol{C}_\beta}$ where $\frac{p}{2} \leq \beta \leq \alpha$.
Then, $\forall d = 1, \ldots, D$:

$$\left|\Pi[f_d] - \hat{\Pi}_{BQ}[f_d]\right| \;\; = \;\; \mathcal{O}\left(n^{-\frac{\beta}{p}+\epsilon}\right)$$

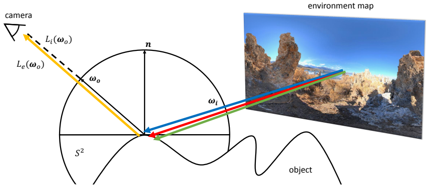
for some $\epsilon > 0$.

Multi-output BQ for the Computer Graphics Example



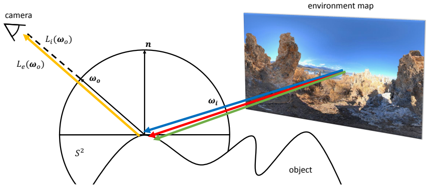
- We compute integrals for different integrands based on various angles ω_0 (akin to a camera moving).
- We pick a separable kernel $\boldsymbol{K}(x, x') = \boldsymbol{B}k(x, x')$ where $\boldsymbol{B}$ is chosen to represent the angle between integrands and $k(\boldsymbol{x}, \boldsymbol{x}') = \frac{8}{3} - \|\boldsymbol{x} - \boldsymbol{x}'\|_2$.
- We can prove that the worst-case integration error converges at a rate $\mathcal{O}(n^{-\frac{3}{4}})$ for each integrand. This is the same rate as uni-output BQ (but we usually improve on constants).

Multi-output BQ for the Computer Graphics Example



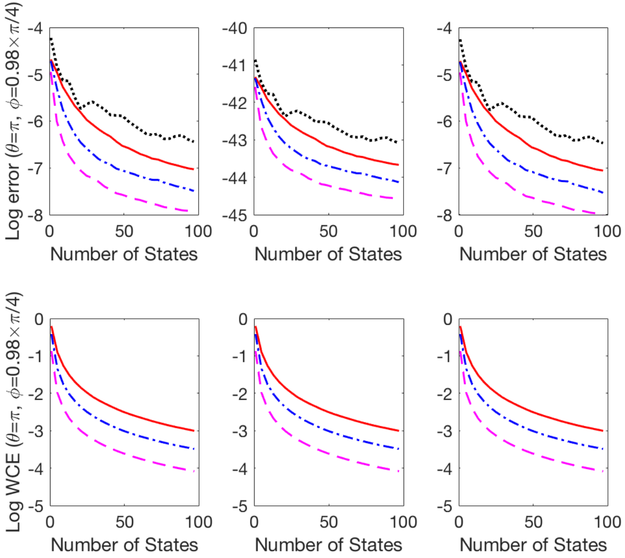
- We compute integrals for different integrands based on various angles ω_0 (akin to a camera moving).
- We pick a separable kernel $\boldsymbol{K}(x, x') = \boldsymbol{B}k(x, x')$ where $\boldsymbol{B}$ is chosen to represent the angle between integrands and $k(\boldsymbol{x}, \boldsymbol{x}') = \frac{8}{3} - \|\boldsymbol{x} - \boldsymbol{x}'\|_2$.
- We can prove that the worst-case integration error converges at a rate $\mathcal{O}(n^{-\frac{3}{4}})$ for each integrand. This is the same rate as uni-output BQ (but we usually improve on constants).

Multi-output BQ for the Computer Graphics Example



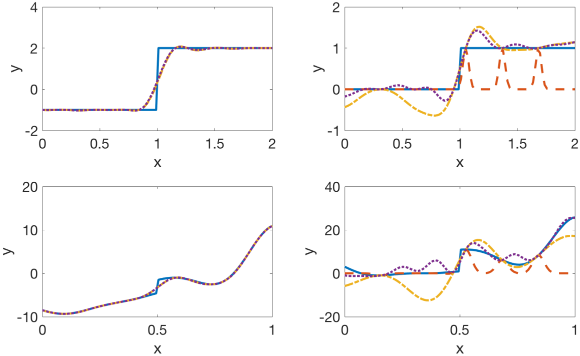
- We compute integrals for different integrands based on various angles ω_0 (akin to a camera moving).
- We pick a separable kernel $\boldsymbol{K}(x, x') = \boldsymbol{B}k(x, x')$ where $\boldsymbol{B}$ is chosen to represent the angle between integrands and $k(\boldsymbol{x}, \boldsymbol{x}') = \frac{8}{3} - \|\boldsymbol{x} - \boldsymbol{x}'\|_2$.
- We can prove that the worst-case integration error converges at a rate $\mathcal{O}(n^{-\frac{3}{4}})$ for each integrand. This is the same rate as uni-output BQ (but we usually improve on constants).

Multi-output BQ for the Computer Graphics Example



Application: Multifidelity Modelling

In each case, we have access to a cheap simulator/function (left) and an expensive simulator/function (right).



blue = truth, red = uni-output, yellow & purple = two outputs

[1] Perdikaris, P., Raissi, M., Damianou, A., Lawrence, N. D., & Karniadakis, G. E. (2016). Nonlinear information fusion algorithms for robust multi-fidelity modeling. Proceedings of the Royal Society A: Mathematical, Physical, and Engineering Sciences, 473(2198).

Application: Multifidelity Modelling

- In multifidelity modelling, multi-output Gaussian processes are already being used as efficient surrogate models.
- Furthermore, we are in general interested in some statistic of the expensive surrogate model. We therefore might as well re-use this multi-output GP approximate the integral.
- Results:

Model	1-output BQ	2-output BQ
Step function (l)	0.024 (0.223)	0.021 (0.213)
Step function (h)	0.405 (0.03)	0.09 (0.091)
Forrester function (l)	0.076 (4.913)	0.076 (4.951)
Forrester function (h)	3.962 (3.984)	2.856 (27.01)

References

[1] Larkin, F. M. (1972). Gaussian measure in Hilbert space and applications in numerical analysis. Rocky Mountain Journal of Mathematics, 2(3), 379422.

[2] Diaconis, P. (1988). Bayesian Numerical Analysis. Statistical Decision Theory and Related Topics IV, 163175.

[3] O'Hagan, A. (1991). Bayes-Hermite quadrature. Journal of Statistical Planning and Inference, 29, 245260.

[4] Rasmussen, C., & Ghahramani, Z. (2002). Bayesian Monte Carlo. In Advances in Neural Information Processing Systems (pp. 489496).

[5] Briol, F.-X., Oates, C. J., Girolami, M., Osborne, M. A., & Sejdinovic, D. (2015). Probabilistic Integration: A Role for Statisticians in Numerical Analysis?, *arXiv:1512.00933*.

[6] Xi, X., Briol, F.-X., & Girolami, M. (2018). Bayesian Quadrature for Multiple Related Integrals. arXiv:1801.04153.