Multi-Dimensional Quasi-Monte Carlo for Functions of Integrals

Aleksei G. Sorokin[1], Jagadeeswaran Rathinavel[2]

[1]Illinois Institute of Technology (IIT), Department of Applied Mathematics
[2]Wi-Tronix LLC

Motivating Examples

Problem: Given prior $P(\theta)$ and likelihood $P(Y \mid \theta)$, find posterior mean

$$\mathbb{E}[\theta \mid Y] \overset{\text{Bayes'}}{=} \frac{\int \theta P(Y \mid \theta) P(\theta) \mathrm{d}\theta}{\int P(Y \mid \theta) P(\theta) \mathrm{d}\theta} = \frac{\mu_1}{\mu_2} = C(\mu_1, \mu_2)$$

When posterior mean is unknown, approximate ratio/function of integrals
Other Problems written as functions of multiple integrals

- Vectorized expectation $\mathbb{E}[\boldsymbol{X}] = \left(\mathbb{E}[X_1], \mathbb{E}[X_2] \ldots, \mathbb{E}[X_d]\right)^T$
- $\text{Cov}(U,V) = \mathbb{E}[UV] - \mathbb{E}[U]\mathbb{E}[V]$
- Sensitivity indices for global sensitivity analysis

Question: How to extend single integral approximation to function(s) of integral(s)?

Framework

Problem: Approximate $C(\mathbb{E}[f(X)])$ within error tolerance ε from error metric $h(\cdot,\varepsilon)$, where $X \sim \mathcal{U}[0,1]^d$, f a vector function, C a scalar function

- Transformations can take a variety of integrals into $\mathbb{E}[f(X)]$, $X \sim \mathcal{U}[0,1]^d$ form [1]
- Example metrics maps
  - $h(s,\varepsilon) = \varepsilon_{\text{abs}}$, absolute error
  - $h(s,\varepsilon) = \max(\varepsilon_{\text{abs}}, |s|\varepsilon_{\text{rel}})$, absolute or relative
  - $h(s,\varepsilon) = \min(\varepsilon_{\text{abs}}, |s|\varepsilon_{\text{rel}})$, absolute and relative

Individual solutions

$$\boldsymbol{\mu} = (\mu_1,\dots,\mu_\rho)^T = (\mathbb{E}[f_1(X)],\dots\mathbb{E}[f_\rho(X)])^T$$

Combined solution

$$s = C(\boldsymbol{\mu}) = C(\mu_1,\dots,\mu_\rho)$$

Proposed Method

Ideas

- Quasi-Monte Carlo (QMC) methods can efficiently bound $\boldsymbol{\mu}$ s.t. $\boldsymbol{\mu} \in [\boldsymbol{\mu}^-, \boldsymbol{\mu}^+]$ either guaranteed or with high probability
- Often straightforward to find bound propogation functions C^-, C^+ s.t.

$$s \in [s^-, s^+] = [C^-(\boldsymbol{\mu}^-, \boldsymbol{\mu}^+), C^+(\boldsymbol{\mu}^-, \boldsymbol{\mu}^+)]$$

- QMC methods iteratively double sample size

Method

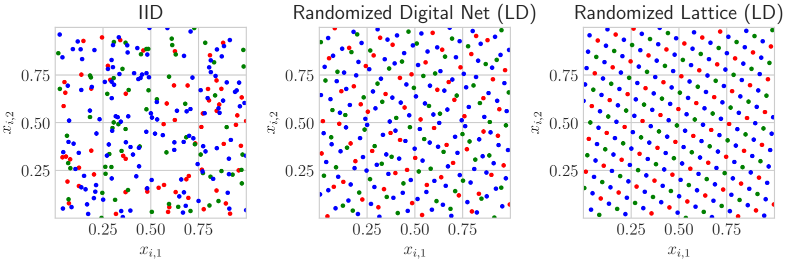
1. Sample f at n QMC samples $X_1, \dots, X_n \in [0,1]^d$
2. Compute individual bounds $\boldsymbol{\mu}^-, \boldsymbol{\mu}^+$
3. Compute combined bounds s^-, s^+
4. Compute optimal approximation $\hat{s} = \frac{1}{2}\left[s^- + s^+ + h(s^-, \varepsilon) - h(s^+, \varepsilon)\right]$
5. If $s^+ - s^- < h(s^-, \varepsilon) + h(s^+, \varepsilon)$, then the error criterion is satisfied
   Else, double samples size s.t. $n \leftarrow 2n$ and go to step 1

Quasi-Monte Carlo (QMC) Methods

$$\hat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}_i) \approx \int_{[0,1]^d} f(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = \boldsymbol{\mu}$$

$(\boldsymbol{x}_i)_{i\geq 1} \subseteq [0,1]^d$ sampling nodes chosen to be

- IID $\to$ Crude Monte Carlo $\to$ $\mathcal{O}(n^{-1/2})$ convergence of $\hat{\boldsymbol{\mu}}$ to $\boldsymbol{\mu}$
- Low Discrepancy (LD) $\to$ QMC $\to$ $\mathcal{O}(n^{-1+\delta})$ convergence, $0 < \delta \ll 1/2$
    - Prefer extensible, randomized LD sequences with $n = 2^m$



Randomized QMC with Replications Method [2]

R IID randomizations of LD points $\{\boldsymbol{x}_i\}_{i=1}^{2^m}$

$$\hat{\mu}_r = \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}_{i,r}), \qquad \hat{\mu} = \frac{1}{R}\sum_{r=1}^{R}\hat{\mu}_r, \qquad S_r = \sqrt{\frac{1}{R(R-1)}\sum_{i=1}^{R}(\hat{\mu}_r - \hat{\mu})^2}$$

$$\mu^{\pm} = \hat{\mu} \pm t_{R-1}^{1-\alpha/2} S_r$$

Vectorize to produce $\boldsymbol{\mu}^-, \boldsymbol{\mu}^+$

Covariance Example

Combined Solution: $s = \mathrm{Cov}(U,V) = \mathbb{E}[UV] - \mathbb{E}[U]\mathbb{E}[V] \in \mathbb{R}$

Individual Solutions: $\boldsymbol{\mu} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix} = \begin{pmatrix} \mathbb{E}[U] \\ \mathbb{E}[V] \\ \mathbb{E}[UV] \end{pmatrix} \in \mathbb{R}^3$

$$
\begin{aligned}
C(\boldsymbol{\mu}) &= \mu_3 - \mu_1\mu_2 \\
C^-(\boldsymbol{\mu}^-,\boldsymbol{\mu}^+) &= \mu_3^- - \max\left(\mu_1^+\mu_2^+,\, \mu_1^+\mu_2^-,\, \mu_1^-\mu_2^+,\, \mu_1^-\mu_2^-\right) \\
C^+(\boldsymbol{\mu}^-,\boldsymbol{\mu}^+) &= \mu_3^+ - \min\left(\mu_1^+\mu_2^+,\, \mu_1^+\mu_2^-,\, \mu_1^-\mu_2^+,\, \mu_1^-\mu_2^-\right)
\end{aligned}
$$

QMCPy implementation [3] does not require specifying C, only C^-,C^+

Vectorized Functions and Dependency

Up to now:	$f:[0,1]^d \to \mathbb{R}^\rho;$	$\boldsymbol{\mu} \in \mathbb{R}^\rho;$	$C^-, C^+ : \mathbb{R}^\rho \times \mathbb{R}^\rho \to \mathbb{R};$	$s \in \mathbb{R}$
Generalized:	$f:[0,1]^d \to \mathbb{R}^{\boldsymbol{\rho}};$	$\boldsymbol{\mu} \in \mathbb{R}^{\boldsymbol{\rho}};$	$C^-, C^+ : \mathbb{R}^{\boldsymbol{\rho}} \times \mathbb{R}^{\boldsymbol{\rho}} \to \mathbb{R}^{\boldsymbol{\eta}};$	$\boldsymbol{s} \in \mathbb{R}^{\boldsymbol{\eta}}$

$\boldsymbol{\rho}, \boldsymbol{\eta}$ shape vectors, e.g. $\boldsymbol{s} \in \mathbb{R}^{2\times 3\times 4} \implies \boldsymbol{\eta} = (2,3,4)^T$

Dependency function $D : \{\text{True}, \text{False}\}^{\boldsymbol{\eta}} \to \{\text{True}, \text{False}\}^{\boldsymbol{\rho}}$ answers question:

If $s_{\boldsymbol{i}}$ insufficiently approximated, which $\mu_{\boldsymbol{j}}$ require further sampling? $\equiv$

If $s_{\boldsymbol{i}}$ sufficiently approximated, which $\mu_{\boldsymbol{j}}$ can we ignore computing f for?

Multi-indices $\quad 0 \leq \boldsymbol{i} \leq \boldsymbol{\eta}, \qquad 0 \leq \boldsymbol{j} \leq \boldsymbol{\rho}$

D enables economical function evaluation

Vectorized Acquisition Functions for Bayesian Optimization (BO)

BO sequentially optimizes f via surrogate Gaussian process g [4]

Batch sequential optimization: choose next q points satisfying

$$\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_{n+q} = \underset{X\in[0,1]^{q\times d}}{\operatorname{argmax}}\ \alpha(X)$$

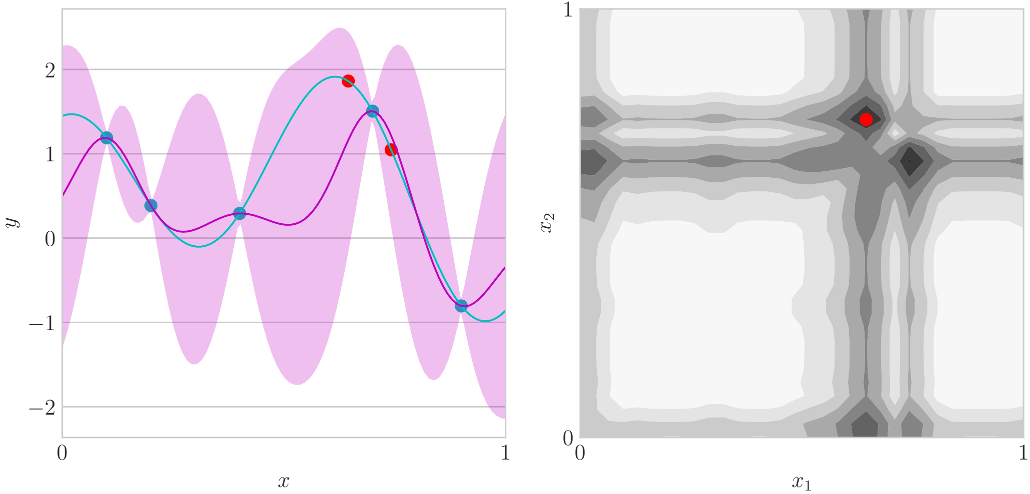
q-EI acquisition function

$$\alpha(X) = \mathbb{E}[a(Y)|\underbrace{Y \sim P(g(X)|D)}_{\text{Gaussian posterior}}], \qquad a(\boldsymbol{y}) = \max_{1\le i\le q}(y_i - y^*)_+$$

Vectorize computation at candidates $X_1,\dots,X_k \in [0,1]^{q\times d}$

$$\begin{pmatrix}\alpha(X_1)\\ \dots \\ \alpha(X_k)\end{pmatrix} = \mathbb{E}\begin{pmatrix}a(\Phi_1^{-1}(X))\\ \dots \\ a(\Phi_k^{-1}(X))\end{pmatrix}, \qquad X \sim \mathcal{U}[0,1]^q$$

BO via q-EI, $q = 2$



data true function posterior mean 95% CI next points by QEI

Bayesian Logistic Regression

Prior on coefficients

$$\boldsymbol{s} \sim \mathcal{N}(\boldsymbol{\nu}, \Sigma)$$

Likelihood based on N data points

$$P(y_1, \ldots, y_N; \boldsymbol{s}) = \prod_{i=1}^{N} \left(\frac{\exp(\boldsymbol{s}.\boldsymbol{x}_i)}{1 + \exp(\boldsymbol{s}.\boldsymbol{x}_i)} \right)^{-y_i} \left(1 - \frac{\exp(\boldsymbol{s}.\boldsymbol{x}_i)}{1 + \exp(\boldsymbol{s}.\boldsymbol{x}_i)} \right)^{1-y_i}$$

Coefficients of interest by Bayes rule, $\boldsymbol{\mu} \in \mathbb{R}^{2 \times d}$

$$\mathbb{E}[s_j; y_1, \ldots, y_N] = \frac{\overbrace{\int_{\mathbb{R}^d} s_j P(y_1, \ldots, y_N; \boldsymbol{s}) P(\boldsymbol{s}) \mathrm{d}\boldsymbol{s}}^{\mu_{1j}}}{\underbrace{\int_{\mathbb{R}^d} P(y_1, \ldots, y_N; \boldsymbol{s}) P(\boldsymbol{s}) \mathrm{d}\boldsymbol{s}}_{\mu_{2j}}}, \qquad j = 1, \ldots, d$$

Bayesian Logistic Regression Continued

Bound propagation

$$s_j^- = C_j^-(\boldsymbol{\mu}^-, \boldsymbol{\mu}^+) = \min_{\boldsymbol{\mu} \in [\boldsymbol{\mu}^-, \boldsymbol{\mu}^+]} \frac{\mu_{1j}}{\mu_{2j}} = \begin{cases} -\infty, & 0 \in [\mu_{2j}^-, \mu_{2j}^+] \\ \min\left(\frac{\mu_{1j}^-}{\mu_{2j}^-}, \frac{\mu_{1j}^+}{\mu_{2j}^-}, \frac{\mu_{1j}^-}{\mu_{2j}^+}, \frac{\mu_{1j}^+}{\mu_{2j}^+}\right), & \text{else} \end{cases},$$

$$s_j^+ = C_j^+(\boldsymbol{\mu}^-, \boldsymbol{\mu}^+) = \max_{\boldsymbol{\mu} \in [\boldsymbol{\mu}^-, \boldsymbol{\mu}^+]} \frac{\mu_{1j}}{\mu_{2j}} = \begin{cases} \infty, & 0 \in [\mu_{2j}^-, \mu_{2j}^+] \\ \max\left(\frac{\mu_{1j}^-}{\mu_{2j}^-}, \frac{\mu_{1j}^+}{\mu_{2j}^-}, \frac{\mu_{1j}^-}{\mu_{2j}^+}, \frac{\mu_{1j}^+}{\mu_{2j}^+}\right), & \text{else} \end{cases}.$$

Dependency function $D : \{\text{True}, \text{False}\}^d \to \{\text{True}, \text{False}\}^{2 \times d}$

$$D(\boldsymbol{b}) = \begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{b} \end{pmatrix}$$

Sensitivity Indices [2, 5]: quantify variance attributable to $u \subseteq \{1,\dots,d\}$

Funcational ANOVA decomposes $f \in L^2(0,1)^d$ into orthogonal $\{f_u\}_{u\subseteq\{1,\dots,d\}}$ s.t.

$$f(\boldsymbol{x}) = \sum_{u\subseteq\{1,\dots,d\}} f_u(\boldsymbol{x}_u) \qquad \text{and} \qquad \sigma^2 = \sum_{u\subseteq\{1,\dots,d\}} \sigma_u^2$$

Closed and total Sobol' indices

$$\underline{\tau}_u^2 = \sum_{v\subset u} \sigma_v^2 \qquad \text{and} \qquad \overline{\tau}_u^2 = \sum_{v\cap u\neq\emptyset} \sigma_v^2$$

Closed and total sensitivity indices (normalized Sobol' indices)

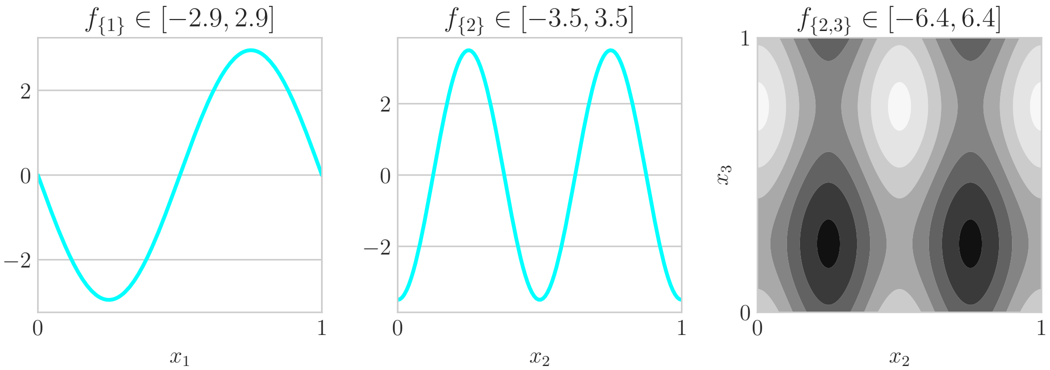
$$\underline{s}_u = \underline{\tau}_u^2/\sigma^2 \qquad \text{and} \qquad \overline{s}_u = \overline{\tau}_u^2/\sigma^2$$

$\underline{\tau}_u^2$, $\overline{\tau}_u^2$, and σ^2 can all be written in terms of expectations

Sensitivity Indices: Ishigami Function [6]

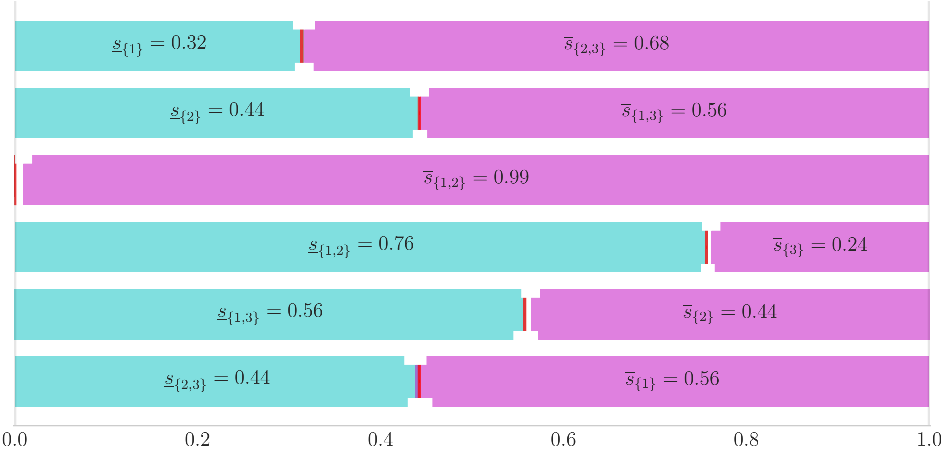
$$g(T) = \left(1 + bT_3^4\right)\sin(T_1) + a\sin^2(T_2), \qquad T \sim \mathcal{U}(-\pi,\pi)^3$$

$$f(X) = g\left(\pi\left(2X - 1\right)\right), \qquad X \sim \mathcal{U}(0,1)^3$$



Sensitivity Indices: Ishigami Function

$$\underline{s}_u + \overline{s}_{u^c} = 1, \qquad u \in \{1,\ldots,d\}$$



Sensitivity Indices: Neural Network Classifier for the Iris Dataset [7]

Trained network achieves 98% accuracy on validation set

Singleton Closed Indices

	sepal length	sepal width	petal length	petal width	sum
setosa	0.2%	5.9%	71.4%	4.6%	82.0%
versicolor	7.1%	2.2%	32.8%	2.1%	44.3%
virginica	8.2%	1.0%	50.0%	12.0%	71.2%

Petal length accounts for most variation among singletons

Non-singleton interactions, $|u| > 1$, are most important for differentiating versicolor

References I

1. Choi, S.-C. T., Hickernell, F. J., Jagadeeswaran, R., McCourt, M. J. & Sorokin, A. G. *Quasi-Monte Carlo Software.* in *Monte Carlo and Quasi-Monte Carlo Methods* (ed Keller, A.) (Springer International Publishing, Cham, 2022), 23–47. ISBN: 978-3-030-98319-2.
2. Owen, A. B. *Monte Carlo theory, methods and examples.* (2018).
3. Choi, S.-C. T., Hickernell, F. J., Jagadeeswaran, R., McCourt, M. J. & Sorokin, A. G. *QMCPy: A Quasi-Monte Carlo Python Library.* 2020+. https://github.com/QMCSoftware/QMCSoftware.
4. Frazier, P. I. *A Tutorial on Bayesian Optimization.* 2018. https://arxiv.org/abs/1807.02811.
5. Jiménez Rugama, L. A. & Gilquin, L. Reliable error estimation for Sobol' indices. *Statistics and Computing* **28,** 725–738. https://hal.inria.fr/hal-01358067 (July 2018).

References II

6. Ishigami, T. & Homma, T. *An importance quantification technique in uncertainty analysis for computer models.* in *[1990] Proceedings. First International Symposium on Uncertainty Modeling and Analysis* (1990), 398–403.

7. Hoyt, C. & Owen, A. B. Efficient Estimation of the ANOVA Mean Dimension, with an Application to Neural Net Classification. *SIAM/ASA Journal on Uncertainty Quantification* **9,** 708–730. eprint: https://doi.org/10.1137/20M1350236. https://doi.org/10.1137/20M1350236 (2021).