Adaptive Probabilistic Numerical Methods Using Fast Transforms

Fred J. Hickernell & Jagadees Rathinavel
Department of Applied Mathematics
Center for Interdisciplinary Scientific Computation
Illinois Institute of Technology
hickernell@iit.edu mypages.iit.edu/~hickernell

Thanks to the organizers, the GAIL team, NSF-DMS-1522687 and NSF-DMS-1638521 (SAMSI)

SAMSI-Lloyds-Turing Workshop on Probabilistic Numerical Methods, April 11, 2018

When Do We Stop?

Compute the solution to a linear problem:

$$\text{sol}(f) = \begin{cases} \displaystyle\int_{\mathbb{R}^d} f(\mathbf{x})\, \varrho(\mathbf{x})\, \mathrm{d}\mathbf{x} & \text{Bayesian inference, financial risk, statistical physics, } \ldots \\ f(\mathbf{x}) & \text{surrogate modeling for computer experiments, } \ldots \\ \vdots & \end{cases}$$

Desired solution: An adaptive algorithm, $\text{app}(\cdot, \cdot)$ of the form

$$\text{app}(f, \varepsilon) = w_{0,n} + \sum_{i=1}^{n} w_{i,n} f(\mathbf{x}_i),$$

where n, $\{\mathbf{x}_i\}_{i=1}^{\infty}$, $w_{0,n}$, and $\boldsymbol{w} = (w_{i,n})_{i=1}^{n}$ are chosen to guarantee

$$\left|\text{sol}(f) - \text{app}(f, \varepsilon)\right| \leqslant \varepsilon \text{ with high probability} \qquad \forall \varepsilon > 0, \text{ reasonable } f$$

The Probabilistic Numerics Approach

Assume $f \sim \mathcal{GP}(m, s^2 C_\theta)$, a sample from a Gaussian process. Defining

$$c = \mathrm{sol}^{\cdot}(\mathrm{sol}^{\cdot\cdot}(C_\theta(\cdot,\cdot\cdot))), \qquad \mathbf{c} = \left(\mathrm{sol}^{\cdot}(C_\theta(\cdot,\mathbf{x}_1)), \ldots, \mathrm{sol}^{\cdot}(C_\theta(\cdot,\mathbf{x}_n))\right)^T, \qquad \mathsf{C} = \left(C_\theta(\mathbf{x}_i,\mathbf{x}_j)\right)_{i,j=1}^n$$

and choosing the weights as

$$w_0 = m[\mathrm{sol}(1) - \mathbf{c}^T\mathsf{C}^{-1}\mathbf{1}], \quad \boldsymbol{w} = \mathsf{C}^{-1}\mathbf{c}, \qquad \mathrm{app}(f,\varepsilon) = w_0 + \boldsymbol{w}^T\mathbf{f}, \quad \mathbf{f} = \left(f(\mathbf{x}_i)\right)_{i=1}^n.$$

yields an unbiased approximation:

$$\mathrm{sol}(f) - \mathrm{app}(f,\varepsilon) \,\big|\, \mathbf{f} = \mathbf{y} \sim \mathcal{N}\Big(0, s^2(c - \mathbf{c}^T\mathsf{C}^{-1}\mathbf{c})\Big)$$

If n is chosen large enough to make

$$2.58 s\sqrt{c - \mathbf{c}^T\mathsf{C}^{-1}\mathbf{c}} \leqslant \varepsilon,$$

then we are assured that

$$\mathbb{P}_f\left[|\mathrm{sol}(f) - \mathrm{app}(f,\varepsilon)| \leqslant \varepsilon\right] \geqslant 99\%.$$

The Probabilistic Numerics Approach

Assume $f \sim \mathcal{GP}(m, s^2 C_\theta)$, a sample from a Gaussian process. Defining

$$c = \mathrm{sol}^{\cdot}(\mathrm{sol}^{\cdot\cdot}(C_\theta(\cdot,\cdot\cdot))), \qquad \mathbf{c} = \left(\mathrm{sol}^{\cdot}(C_\theta(\cdot,\mathbf{x}_1)),\ldots,\mathrm{sol}^{\cdot}(C_\theta(\cdot,\mathbf{x}_n))\right)^T, \qquad \mathsf{C} = \left(C_\theta(\mathbf{x}_i,\mathbf{x}_j)\right)_{i,j=1}^n$$

and choosing the weights as

$$w_0 = m[\mathrm{sol}(1) - \mathbf{c}^T\mathsf{C}^{-1}\mathbf{1}], \quad \boldsymbol{w} = \mathsf{C}^{-1}\mathbf{c}, \qquad \mathrm{app}(f,\varepsilon) = w_0 + \boldsymbol{w}^T\mathbf{f}, \quad \mathbf{f} = \left(f(\mathbf{x}_i)\right)_{i=1}^n.$$

yields an unbiased approximation:

$$\mathrm{sol}(f) - \mathrm{app}(f,\varepsilon) \,\big|\, \mathbf{f} = \mathbf{y} \sim \mathcal{N}\Big(0, s^2(c - \mathbf{c}^T\mathsf{C}^{-1}\mathbf{c})\Big)$$

If n is chosen large enough to make

$$2.58 s\sqrt{c - \mathbf{c}^T\mathsf{C}^{-1}\mathbf{c}} \leqslant \varepsilon,$$

then we are assured that

$$\mathbb{P}_f\left[|\mathrm{sol}(f) - \mathrm{app}(f,\varepsilon)| \leqslant \varepsilon\right] \geqslant 99\%.$$

There are issues requiring attention.

Maximum Likelihood Estimation

Minimize minus twice the log likelihood observed with $\mathbf{f} = \mathbf{y}$:

$$2n\log(s) + \log(\det(\mathsf{C})) + s^{-2}(\mathbf{y} - m\mathbf{1})^T\mathsf{C}^{-1}(\mathbf{y} - m\mathbf{1})$$

first with respect to m, then s, then $\boldsymbol{\theta}$:

$$m_{\mathsf{MLE}} = \frac{\mathbf{1}^T\mathsf{C}^{-1}\mathbf{y}}{\mathbf{1}^T\mathsf{C}^{-1}\mathbf{1}}, \qquad s^2_{\mathsf{MLE}} = \frac{1}{n}\mathbf{y}^T\left[\mathsf{C}^{-1} - \frac{\mathsf{C}^{-1}\mathbf{1}\mathbf{1}^T\mathsf{C}^{-1}}{\mathbf{1}^T\mathsf{C}^{-1}\mathbf{1}}\right]\mathbf{y}$$

$$\boldsymbol{\theta}_{\mathsf{MLE}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \left\{ n\log\left(\mathbf{y}^T\left[\mathsf{C}^{-1} - \frac{\mathsf{C}^{-1}\mathbf{1}\mathbf{1}^T\mathsf{C}^{-1}}{\mathbf{1}^T\mathsf{C}^{-1}\mathbf{1}}\right]\mathbf{y}\right) + \log(\det(\mathsf{C})) \right\}$$

Maximum Likelihood Estimation

Minimize minus twice the log likelihood observed with $\mathbf{f} = \mathbf{y}$:

$$2n\log(s) + \log(\det(\mathsf{C})) + s^{-2}(\mathbf{y} - m\mathbf{1})^T\mathsf{C}^{-1}(\mathbf{y} - m\mathbf{1})$$

first with respect to m, then s, then θ:

$$m_{\mathsf{MLE}} = \frac{\mathbf{1}^T\mathsf{C}^{-1}\mathbf{y}}{\mathbf{1}^T\mathsf{C}^{-1}\mathbf{1}}, \qquad s^2_{\mathsf{MLE}} = \frac{1}{n}\mathbf{y}^T\left[\mathsf{C}^{-1} - \frac{\mathsf{C}^{-1}\mathbf{1}\mathbf{1}^T\mathsf{C}^{-1}}{\mathbf{1}^T\mathsf{C}^{-1}\mathbf{1}}\right]\mathbf{y}$$

$$\theta_{\mathsf{MLE}} = \underset{\theta}{\operatorname{argmin}} \left\{ n\log\left(\mathbf{y}^T\left[\mathsf{C}^{-1} - \frac{\mathsf{C}^{-1}\mathbf{1}\mathbf{1}^T\mathsf{C}^{-1}}{\mathbf{1}^T\mathsf{C}^{-1}\mathbf{1}}\right]\mathbf{y}\right) + \log(\det(\mathsf{C})) \right\}$$

Stopping criterion becomes

$$2.58\sqrt{\underbrace{\frac{(c - \mathbf{c}^T\mathsf{C}^{-1}\mathbf{c})}{n}}_{\text{depends on design}} \underbrace{\mathbf{y}^T\left[\mathsf{C}^{-1} - \frac{\mathsf{C}^{-1}\mathbf{1}\mathbf{1}^T\mathsf{C}^{-1}}{\mathbf{1}^T\mathsf{C}^{-1}\mathbf{1}}\right]\mathbf{y}}_{\text{depends on data}}} \leqslant \varepsilon,$$

Maximum Likelihood Estimation

$$m_{\text{MLE}} = \frac{\mathbf{1}^T \mathbf{C}^{-1} \mathbf{y}}{\mathbf{1}^T \mathbf{C}^{-1} \mathbf{1}}, \qquad s^2_{\text{MLE}} = \frac{1}{n} \mathbf{y}^T \left[\mathbf{C}^{-1} - \frac{\mathbf{C}^{-1} \mathbf{1} \mathbf{1}^T \mathbf{C}^{-1}}{\mathbf{1}^T \mathbf{C}^{-1} \mathbf{1}} \right] \mathbf{y}$$

$$\theta_{\text{MLE}} = \underset{\theta}{\text{argmin}} \left\{ n \log \left(\mathbf{y}^T \left[\mathbf{C}^{-1} - \frac{\mathbf{C}^{-1} \mathbf{1} \mathbf{1}^T \mathbf{C}^{-1}}{\mathbf{1}^T \mathbf{C}^{-1} \mathbf{1}} \right] \mathbf{y} \right) + \log(\det(\mathbf{C})) \right\}$$

Stopping criterion becomes

$$2.58 \sqrt{\underbrace{\frac{(c - \mathbf{c}^T \mathbf{C}^{-1} \mathbf{c})}{n}}_{\text{depends on design}} \underbrace{\mathbf{y}^T \left[\mathbf{C}^{-1} - \frac{\mathbf{C}^{-1} \mathbf{1} \mathbf{1}^T \mathbf{C}^{-1}}{\mathbf{1}^T \mathbf{C}^{-1} \mathbf{1}} \right] \mathbf{y}}_{\text{depends on data}}} \leqslant \varepsilon,$$

Q1: How large a family of kernels, C_θ is needed in practice to be confident in the error bound?

Q2: Are the better ways of finding the right θ, say cross-validation?

Q3: Can we check out assumption that f really comes from a Gaussian process? If not, are there alternatives

Low Discrepancy Sampling

Suppose that the domain is $[0,1]^d$. Low discrepancy sampling places the $\mathbf{x}_i$ more evenly than IID sampling[1]

IID points

Sobol' points

Integration lattice points

[1] Dick, J. *et al.* High dimensional integration — the Quasi-Monte Carlo way. *Acta Numer.* **22,** 133–288 (2013), H., F. J. *et al. SAMSI Program on Quasi-Monte Carlo and High-Dimensional Sampling Methods for Applied Mathematics.* https://www.samsi.info/programs-and-activities/year-long-research-programs/2017-18-program-quasi-monte-carlo-high-dimensional-sampling-methods-applied-mathematics-qmc/.

Covariance Kernels that Match the Design

Suppose that the covariance kernel, C_θ, and the design, $\{x_i\}_{i=1}^n$, have special properties:

$$\mathsf{C} = \Big(C_\theta(x_i, x_j)\Big)_{i,j=1}^n = \big(\boldsymbol{C}_1, \ldots, \boldsymbol{C}_n\big) = \frac{1}{n}\mathsf{V}\Lambda\mathsf{V}^H, \quad \mathsf{V}^H = n\mathsf{V}^{-1}, \quad \Lambda = \mathsf{diag}(\lambda_1, \ldots, \lambda_n) = \mathsf{diag}(\boldsymbol{\lambda})$$

$$\mathsf{V} = \begin{pmatrix} \boldsymbol{V}_1 & \cdots & \boldsymbol{V}_n \end{pmatrix} = \begin{pmatrix} \boldsymbol{v}_1 & \cdots & \boldsymbol{v}_n \end{pmatrix}^T, \quad \boldsymbol{V}_1 = \boldsymbol{v}_1 = \boldsymbol{1}$$

Suppose that $\mathsf{V}^T z$ is a fast transform ($\mathcal{O}(n \log n)$ cost) applied to z. Then it follows that

$$\boldsymbol{\lambda} = \mathsf{V}^T \boldsymbol{C}_1 \text{ is fast}, \qquad \mathsf{C}^{-1}\boldsymbol{1} = \frac{\boldsymbol{1}}{\lambda_1}$$

Let $\boldsymbol{y}$ be the observed function values. Recall $c = \mathsf{sol}^{\cdot}(\mathsf{sol}^{\cdot\cdot}(C_\theta(\cdot,\cdot\cdot)))$, and let

$$\hat{\boldsymbol{y}} = \mathsf{V}^T\boldsymbol{y}, \qquad \hat{\boldsymbol{c}} = \mathsf{V}^T\boldsymbol{c}, \quad \boldsymbol{c} = \big(\mathsf{sol}^{\cdot}(C_\theta(\cdot, x_1)), \ldots, \mathsf{sol}^{\cdot}(C_\theta(\cdot, x_n))\big)^T$$

Then using the MLE estimates, the approximate solution and the stopping criterion become:

$$\mathsf{app}(f, \varepsilon) = \frac{\hat{y}_1\, \mathsf{sol}(1)}{n} + \sum_{i=2}^{n} \frac{\hat{y}_i^* \hat{c}_i}{\lambda_i}, \qquad 2.58\sqrt{\left(c - \frac{1}{n}\sum_{i=1}^{n} \frac{|\hat{c}_i|^2}{\lambda_i}\right) \frac{1}{n^2} \sum_{i=2}^{n} \frac{|\hat{y}_i|^2}{\lambda_i}} \leqslant \varepsilon$$

Covariance Kernels that Match the Design

$$c = \mathrm{sol}^{\cdot}(\mathrm{sol}^{\cdot\cdot}(C_\theta(\cdot,\cdot\cdot))), \qquad \mathsf{C} = \left(C_\theta(\boldsymbol{x}_i,\boldsymbol{x}_j)\right)_{i,j=1}^{n} = \frac{1}{n}\mathsf{V}\Lambda\mathsf{V}^H, \quad \Lambda = \mathrm{diag}(\boldsymbol{\lambda}), \quad \boldsymbol{V}_1 = \boldsymbol{v}_1 = \mathbf{1}$$

$$\mathsf{V}^T\boldsymbol{z} \text{ is } \mathcal{O}(n\log n), \qquad \boldsymbol{\lambda} = \mathsf{V}^T\boldsymbol{C}_1, \qquad \mathsf{C}^{-1}\mathbf{1} = \frac{\mathbf{1}}{\lambda_1}$$

$$\widehat{\boldsymbol{y}} = \mathsf{W}^T\boldsymbol{y}, \qquad \widehat{\boldsymbol{c}} = \mathsf{W}^T\boldsymbol{c}, \quad \boldsymbol{c} = \left(\mathrm{sol}^{\cdot}(C_\theta(\cdot,\boldsymbol{x}_1)),\ldots,\mathrm{sol}^{\cdot}(C_\theta(\cdot,\boldsymbol{x}_n))\right)^T$$

$$\mathrm{app}(f,\varepsilon) = \frac{\widehat{y}_1\,\mathrm{sol}(1)}{n} + \sum_{i=2}^{n}\frac{\widehat{y}_i^*\widehat{c}_i}{\lambda_i}, \qquad 2.58\sqrt{\left(c - \frac{1}{n}\sum_{i=1}^{n}\frac{|\widehat{c}_i|^2}{\lambda_i}\right)\frac{1}{n^2}\sum_{i=2}^{n}\frac{|\widehat{y}_i|^2}{\lambda_i}} \leqslant \varepsilon$$

$$\theta_{\mathsf{MLE}} = \operatorname*{argmin}_{\theta}\left\{n\log\left(\sum_{i=2}^{n}\frac{|\widehat{y}_i|^2}{\lambda_i}\right) + \sum_{i=1}^{n}\log(\lambda_i)\right\}$$

For integration with respect to a density and our special kernels, $\mathrm{sol}(1) = 1$, $\mathbf{c} = \mathbf{1}$, and $c = 1$:

$$\mathrm{app}(f,\varepsilon) = \frac{\widehat{y}_1}{n}, \qquad 2.58\sqrt{\left(1 - \frac{n}{\lambda_1}\right)\frac{1}{n^2}\sum_{i=2}^{n}\frac{|\widehat{y}_i|^2}{\lambda_i}} \leqslant \varepsilon$$

Covariance Kernels that Match the Design

$$c = \mathrm{sol}^{\cdot}(\mathrm{sol}^{\cdot\cdot}(C_\theta(\cdot,\cdot\cdot))), \qquad \mathsf{C} = \left(C_\theta(\mathbf{x}_i,\mathbf{x}_j)\right)_{i,j=1}^{n} = \frac{1}{n}\mathsf{V}\Lambda\mathsf{V}^H, \quad \Lambda = \mathrm{diag}(\boldsymbol{\lambda}), \quad \mathbf{V}_1 = \boldsymbol{\nu}_1 = \mathbf{1}$$

$$\mathsf{V}^T\mathbf{z} \text{ is } \mathcal{O}(n\log n), \qquad \boldsymbol{\lambda} = \mathsf{V}^T\boldsymbol{C}_1, \qquad \mathsf{C}^{-1}\mathbf{1} = \frac{\mathbf{1}}{\lambda_1}$$

$$\widehat{\mathbf{y}} = \mathsf{W}^T\mathbf{y}, \qquad \widehat{\mathbf{c}} = \mathsf{W}^T\mathbf{c}, \quad \mathbf{c} = \left(\mathrm{sol}^{\cdot}(C_\theta(\cdot,\mathbf{x}_1)),\ldots,\mathrm{sol}^{\cdot}(C_\theta(\cdot,\mathbf{x}_n))\right)^T$$

$$\mathrm{app}(f,\varepsilon) = \frac{\widehat{y}_1\,\mathrm{sol}(1)}{n} + \sum_{i=2}^{n}\frac{\widehat{y}_i^*\widehat{c}_i}{\lambda_i}, \qquad 2.58\sqrt{\left(c - \frac{1}{n}\sum_{i=1}^{n}\frac{|\widehat{c}_i|^2}{\lambda_i}\right)\frac{1}{n^2}\sum_{i=2}^{n}\frac{|\widehat{y}_i|^2}{\lambda_i}} \leqslant \varepsilon$$

$$\theta_{\mathsf{MLE}} = \operatorname*{argmin}_{\theta}\left\{n\log\left(\sum_{i=2}^{n}\frac{|\widehat{y}_i|^2}{\lambda_i}\right) + \sum_{i=1}^{n}\log(\lambda_i)\right\}$$

For integration with respect to a density and our special kernels, $\mathrm{sol}(1) = 1$, $\mathbf{c} = \mathbf{1}$, and $c = 1$:

$$\mathrm{app}(f,\varepsilon) = \frac{\widehat{y}_1}{n}, \qquad 2.58\sqrt{\left(1 - \frac{n}{\lambda_1}\right)\frac{1}{n^2}\sum_{i=2}^{n}\frac{|\widehat{y}_i|^2}{\lambda_i}} \leqslant \varepsilon$$

Q4: How do we avoid round-off in evaluating $1 - n/\lambda_1$?

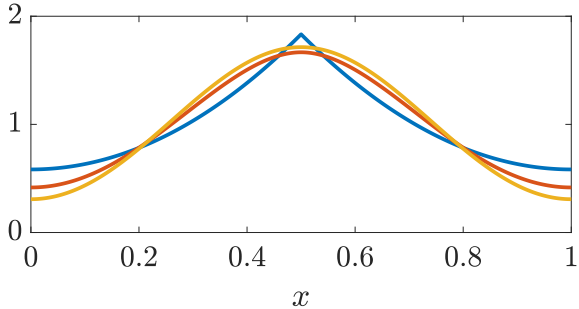
Form of Matching Covariance Kernels

Typically the domain of f is $[0,1)^d$, and

$$C(\mathbf{x},\mathbf{t}) = \begin{cases} \widetilde{C}(\mathbf{x}-\mathbf{t} \bmod 1) & \text{integration lattices} \\ \widetilde{C}(\mathbf{x}\oplus\mathbf{t}) & \text{Sobol' sequences, } \oplus \text{ means digitwise addition modulo 2} \end{cases}$$

E.g., for integration lattices

$$C(\mathbf{x},\mathbf{t}) = \prod_{k=1}^{d}[1-\theta_1(-1)^{\theta_2}B_{2\theta_2}(x_k - t_k \bmod 1)], \qquad \theta_1 > 0, \theta_2 \in \mathbb{N}$$



Form of Matching Covariance Kernels

Typically the domain of f is $[0,1)^d$, and

$$C(\mathbf{x},\mathbf{t}) = \begin{cases} \widetilde{C}(\mathbf{x}-\mathbf{t} \bmod 1) & \text{integration lattices} \\ \widetilde{C}(\mathbf{x} \oplus \mathbf{t}) & \text{Sobol' sequences, } \oplus \text{ means digitwise addition modulo 2} \end{cases}$$

E.g., for integration lattices

$$C(\mathbf{x},\mathbf{t}) = \prod_{k=1}^{d} [1-\theta_1(-1)^{\theta_2} B_{2\theta_2}(x_k - t_k \bmod 1)], \qquad \theta_1 > 0, \theta_2 \in \mathbb{N}$$

Q5: How do we periodize f to take advantage of integration lattices and smoother covariance kernels? Does this even work for function approximation?

Q6: May we get higher order convergence with higher order nets and smoother kernels? What do those kernels look like?

Q7: Are low discrepancy designs also good for function approximation?

Q8: Are there other kernel/design combinations that expedite vector-matrix operations?

Option Pricing

$$\mu = \text{fair price} = \int_{\mathbb{R}^d} e^{-rT} \max\left(\frac{1}{d}\sum_{j=1}^{d} S_j - K, 0\right) \frac{e^{-z^T z/2}}{(2\pi)^{d/2}}\, dz \approx \$13.12$$

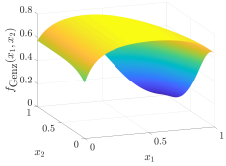
$$S_j = S_0 e^{(r-\sigma^2/2)jT/d + \sigma x_j} = \text{stock price at time } jT/d,$$

$$x = Az, \quad AA^T = \Sigma = \left(\min(i,j)T/d\right)_{i,j=1}^{d}, \quad T = 1/4,\ d = 13 \text{ here}$$

Abs. Error Tolerance	Method	Median Error	Accuracy	Worst 10% n	Worst 10% Time (s)	
1E−2	IID	diff	2E−3	100%	6.1E7	33.000
1E−2	Scr. Sobol'	PCA	1E−3	100%	1.6E4	0.040
1E−2	Scr. Sob. cont. var.	PCA	2E−3	100%	4.1E3	0.019
1E−2	Bayes. Latt.	PCA	2E−3	99%	1.6E4	0.051

Gaussian Probability

$$\mu = \int_{[\mathbf{a},\mathbf{b}]} \frac{\exp\left(-\frac{1}{2}\mathbf{t}^T\Sigma^{-1}\mathbf{t}\right)}{\sqrt{(2\pi)^d \det(\Sigma)}}\,\mathrm{d}\mathbf{t} = \int_{[0,1]^{d-1}} f(\mathbf{x})\,\mathrm{d}\mathbf{x}$$



For some typical choice of $\mathbf{a}$, $\mathbf{b}$, Σ, $d = 3$; $\mu \approx 0.6763$

Rel. Error Tolerance	Method	Median Error	Accuracy	Worst 10% n	Worst 10% Time (s)
1E−2	IID	5E−4	100%	8.1E4	0.020
1E−2	Scr. Sobol'	4E−5	100%	1.0E3	0.005
1E−2	Bayes. Latt.	5E−5	100%	4.1E3	0.023
1E−3	IID	9E−5	100%	2.0E6	0.400
1E−3	Scr. Sobol'	2E−5	100%	2.0E3	0.006
1E−3	Bayes. Latt.	3E−7	100%	6.6E4	0.076
1E−4	Scr. Sobol'	4E−7	100%	1.6E4	0.018
1E−4	Bayes. Latt.	6E−9	100%	5.2E5	0.580
1E−4	Bayes. Latt. Smth.	1E−7	100%	3.3E4	0.047

Recap of Questions

- **Q1:** How large a family of kernels, C_θ is needed in practice to be confident in the error bound?
- **Q2:** Are the better ways of finding the right θ, say cross-validation?
- **Q3:** Can we check out assumption that f really comes from a Gaussian process? If not, are there alternatives
- **Q4:** How do we avoid round-off in evaluating $1 - n/\lambda_1$?
- **Q5:** How do we periodize f to take advantage of integration lattices and smoother covariance kernels? Does this even work for function approximation?
- **Q6:** May we get higher order convergence with higher order nets and smoother kernels? What do those kernels look like?
- **Q7:** Are low discrepancy designs also good for function approximation?
- **Q8:** Are there other kernel/design combinations that expedite vector-matrix operations?
- **Q9:** Is this adaptive Bayesian algorithm competitive with others?
- **Q10:** What other problems are amenable to matched kernels and designs beyond cubature and function approximation?

Thank you

Slides available at www.slideshare.net/fjhickernell/probabilistic-numerics-2018-April-11-london

Dick, J., Kuo, F. & Sloan, I. H. High dimensional integration — the Quasi-Monte Carlo way. *Acta Numer.* **22,** 133–288 (2013).

H., F. J., Kuo, F. Y., L'Ecuyer, P. & Owen, A. B. *SAMSI Program on Quasi-Monte Carlo and High-Dimensional Sampling Methods for Applied Mathematics.* https://www.samsi.info/programs-and-activities/year-long-research-programs/2017-18-program-quasi-monte-carlo-high-dimensional-sampling-methods-applied-mathematics-qmc/.