Intrinsic subspaces of high-dimensional inverse problems and where to find them

In collaboration with Xin Tong (NUS) and Olivier Zahm (INRIA)

Tiangang Cui
Monash Univerisity

`tiangang.cui@monash.edu`
`https://www.fastfins.org/`

DTU, 22 Nov, 2022.

Bayesian inverse problem

- Recover an **unknown parameter** x from the **noisy observation** of a forward model $G(x)$. For instance

$$y = G(x) + \xi \qquad \text{with} \qquad \xi \sim \mathcal{N}(0, \Gamma_{\text{obs}}).$$

- The distribution of $x|y$ is the **posterior**

$$\underbrace{\pi^y(x)}_{\text{posterior}} = \frac{1}{Z} \underbrace{f^y(x)}_{\text{likelihood}} \underbrace{\mu(x)}_{\text{prior}} \qquad \text{with} \qquad f^y(x) = \mathbb{P}(y|x)$$

  - Draw samples $x \sim \pi^y$
  - Find the MAP estimate $\arg\max_x \pi^y(x)$
  - Compute an expectation over posterior $\int h(x)\, \mathrm{d}\pi^y(x)$

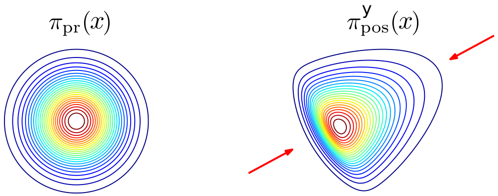
Curse of dimensionality

$$x = (x_1, \dots, x_d) \in \mathbb{R}^d$$

Standard algorithms suffer when $d \gg 1$ (slow convergence, complexity blows up...)

Low effective dimension of Bayesian inverse problems

In many situations, the data are informative only on a low-dimensional subspace



The posterior distribution is *close* to

$$\tilde{\pi}^y(x) \propto \tilde{f}^y(U_r^T x)\mu(x)$$

for some **approximate likelihood** $\tilde{f}^y : \mathbb{R}^r \to \mathbb{R}_+$ and some **matrix** $U_r \in \mathbb{R}^{d\times r}$ with rank r:

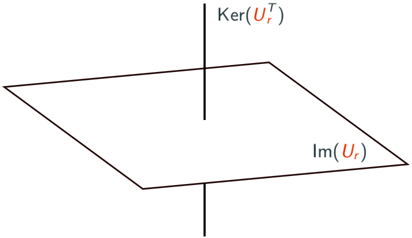
$$\boxed{\mathbb{R}^d = \underbrace{\mathrm{Im}(U_r)}_{\pi^y \neq \mu} \oplus \underbrace{\mathrm{Ker}(U_r^T)}_{\pi^y \approx \mu}}$$

How can this help us?

$$x = \underbrace{U_r x_r}_{\in \mathrm{Im}(U_r)} + \underbrace{U_\perp x_\perp}_{\in \mathrm{Ker}(U_r^T)}$$

Then

$$\tilde{\pi}^y(x) = \underbrace{\left(\frac{1}{\tilde{Z}} \tilde{f}^y(x_r)\mu(x_r)\right)}_{\tilde{\pi}_r(x_r)} \mu(x_\perp | x_r)$$

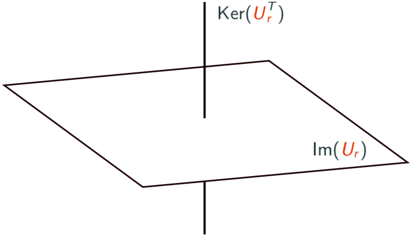


How can this help us?

$$x = \underbrace{U_r x_r}_{\in \mathrm{Im}(U_r)} + \underbrace{U_\perp x_\perp}_{\in \mathrm{Ker}(U_r^T)}$$

Then

$$\tilde{\pi}^y(x) = \underbrace{\left(\frac{1}{\tilde{Z}}\,\tilde{f}^y(x_r)\mu(x_r)\right)}_{\tilde{\pi}_r(x_r)} \mu(x_\perp | x_r)$$



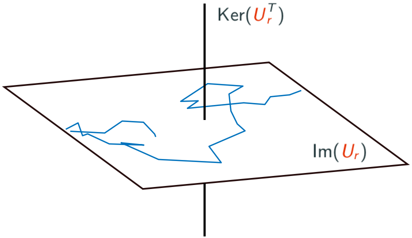
- Exploring $\tilde{\pi}^y$
  1. **Subspace MCMC / transport maps** to get samples $x_r^{(i)} \sim \tilde{\pi}_r^y(x_r)$
  2. Draw samples from the **conditional prior** $x_\perp^{(i)} \sim \mu(x_\perp | x_r^{(i)})$
  3. Assemble $x^{(i)} = U_r x_r^{(i)} + U_\perp x_\perp^{(i)} \sim \tilde{\pi}^y(x)$

How can this help us?

$$x = \underbrace{U_r x_r}_{\in \mathrm{Im}(U_r)} + \underbrace{U_\perp x_\perp}_{\in \mathrm{Ker}(U_r^T)}$$

Then

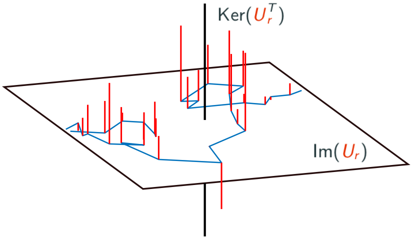
$$\tilde{\pi}^y(x) = \underbrace{\left(\frac{1}{\tilde{Z}} \tilde{f}^y(x_r)\mu(x_r)\right)}_{\tilde{\pi}_r(x_r)} \mu(x_\perp | x_r)$$



- Exploring $\tilde{\pi}^y$
  1. **Subspace MCMC / transport maps** to get samples $x_r^{(i)} \sim \tilde{\pi}_r^y(x_r)$
  2. Draw samples from the **conditional prior** $x_\perp^{(i)} \sim \mu(x_\perp | x_r^{(i)})$
  3. Assemble $x^{(i)} = U_r x_r^{(i)} + U_\perp x_\perp^{(i)} \sim \tilde{\pi}^y(x)$

How can this help us?

$$x = \underbrace{U_r x_r}_{\in \mathrm{Im}(U_r)} + \underbrace{U_\perp x_\perp}_{\in \mathrm{Ker}(U_r^T)}$$



Then

$$\tilde{\pi}^y(x) = \underbrace{\left(\frac{1}{\tilde{Z}}\,\tilde{f}^y(x_r)\mu(x_r)\right)}_{\tilde{\pi}_r(x_r)}\ \mu(x_\perp|x_r)$$

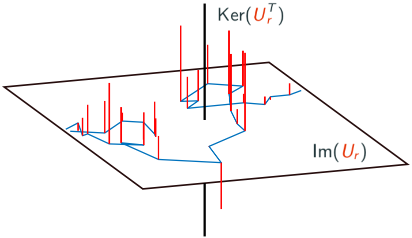
- Exploring $\tilde{\pi}^y$
  1. **Subspace MCMC / transport maps** to get samples $x_r^{(i)} \sim \tilde{\pi}_r^y(x_r)$
  2. Draw samples from the **conditional prior** $x_\perp^{(i)} \sim \mu(x_\perp|x_r^{(i)})$
  3. Assemble $x^{(i)} = U_r x_r^{(i)} + U_\perp x_\perp^{(i)} \sim \tilde{\pi}^y(x)$

How can this help us?

$$x = \underbrace{U_r x_r}_{\in \mathrm{Im}(U_r)} + \underbrace{U_\perp x_\perp}_{\in \mathrm{Ker}(U_r^T)}$$

Then

$$\tilde{\pi}^y(x) = \underbrace{\left(\frac{1}{\tilde{Z}} \tilde{f}^y(x_r)\mu(x_r)\right)}_{\tilde{\pi}_r(x_r)} \mu(x_\perp | x_r)$$



- Exploring $\tilde{\pi}^y$
  1. **Subspace MCMC / transport maps** to get samples $x_r^{(i)} \sim \tilde{\pi}_r^y(x_r)$
  2. Draw samples from the **conditional prior** $x_\perp^{(i)} \sim \mu(x_\perp | x_r^{(i)})$
  3. Assemble $x^{(i)} = U_r x_r^{(i)} + U_\perp x_\perp^{(i)} \sim \tilde{\pi}^y(x)$

- Get samples from the **exact** posterior π^y by correcting $x^{(i)}$ via importance weights or a Metropolis scheme [Cui & Zahm 2021], [Cui, Law & Marzouk 2016],...
- Build tensor-train based Rosenblatt transport [Dolgov et.al. 2016], [Cui & Dolgov 2020] to replace MCMC in $\mathrm{Im}(U_r)$

Controlled approximation problem

Given $\varepsilon > 0$, build an approximation of π^y under the form of

$$\tilde{\pi}^y(x) \propto \tilde{f}^y(U_r^T x)\mu(x) \qquad \text{with} \begin{cases} \tilde{f}^y : \mathbb{R}^r \to \mathbb{R}_{\geq 0} \\ U_r \in \mathbb{R}^{d \times r} \end{cases}$$

with $r = r(\varepsilon) \ll d$ such that

$$\boxed{\text{Variational inference} \quad D_{\mathrm{KL}}(\pi^y || \tilde{\pi}^y) \leq \varepsilon}$$

$$\boxed{\text{Function approximation} \quad D_{\mathrm{H}}(\pi^y || \tilde{\pi}^y) \leq \varepsilon}$$

Road map:

1. Constructing $U_r = U_r(y)$ using gradients of likelihood and $\tilde{f}^y$
2. Data-free dimension reduction $U_r = \cancel{U_r(y)}$
3. A sampling strategy
4. Conclusion

Constructing $U_r = U_r(y)$ using gradients of likelihood and $\tilde{f}^y$

Error analysis (in KL divergence)

$$\tilde{\pi}^y(x) \propto \tilde{f}^y(U_r^T x)\mu(x)$$

Optimal approximation given U_r [Banerjee, Guo & Wang 2005]

Given U_r, the function

$$\tilde{f}^y(x_r) \equiv f_r^y(x_r) = \mathbb{E}_{X\sim\mu}\left(f^y(X)\middle| U_r^T X = x_r\right)$$

minimizes $D_{\mathrm{KL}}\left(\pi^y\middle|\middle|\tilde{\pi}^y\right)$. Then, $\tilde{\pi}^y(x)$ writes

$$\tilde{\pi}_f^y(x) = \pi_f^y(x_r)\mu(x_\perp|x_r), \qquad \pi_f^y(x_r) = \frac{1}{Z} f_r^y(x_r)\mu(x_r)$$

Error analysis (in KL divergence)

$$\tilde{\pi}^y(x) \propto \tilde{f}^y(U_r^T x)\mu(x)$$

Optimal approximation given U_r [Banerjee, Guo & Wang 2005]

Given U_r, the function

$$\tilde{f}^y(x_r) \equiv f_r^y(x_r) = \mathbb{E}_{X\sim\mu}\left(f^y(X)\middle| U_r^T X = x_r\right)$$

minimizes $D_{\mathrm{KL}}\left(\pi^y \middle|\middle| \tilde{\pi}^y\right)$. Then, $\tilde{\pi}^y(x)$ writes

$$\tilde{\pi}_f^y(x) = \pi_f^y(x_r)\mu(x_\perp|x_r), \qquad \pi_f^y(x_r) = \frac{1}{Z} f_r^y(x_r)\mu(x_r)$$

Build U_r by minimizing a certified error bound [Zahm et al. 2022]

Assume $\mu = \mathcal{N}(m_{\mathrm{pr}}, \Sigma_{\mathrm{pr}})$ and let $\tilde{f}^y$ be as above. By logarithmic Sobolev inequalities [Gross 1975] we have

$$\begin{aligned} D_{\mathrm{KL}}\left(\pi^y \middle|\middle| \tilde{\pi}_f^y\right) &\leq \frac{\kappa}{2}\int \|(I_d - U_r U_r^T)\nabla \log f^y\|_{\Sigma_{\mathrm{pr}}}^2 \,\mathrm{d}\pi^y \\ &= \frac{\kappa}{2}\mathrm{tr}\left[\Sigma_r(I_d - U_r U_r^T)\underbrace{\left(\int (\nabla \log f^y)(\nabla \log f^y)^T \,\mathrm{d}\pi^y\right)}_{\mathbf{H}(y)}(I_d - U_r U_r^T)\right] \end{aligned}$$

Building $U_r \in \mathbb{R}^{d \times r}$ by minimizing the bound

Principal Component Analysis of $\nabla \log f^y$

Bound on D_{KL} relies on the Gram matrix

$$\mathbf{H}(y) = \int (\nabla \log f^y)(\nabla \log f^y)^T \, \mathrm{d}\pi^y$$

Find U_r that minimizes the **truncation residual**

$$\mathcal{R}(H(y), U_r) = \operatorname{tr}\left[\Sigma_r (I_d - U_r U_r^T)\mathbf{H}(y)(I_d - U_r U_r^T)\right]$$

1. Solve the generalized eigenvalue problem $\mathbf{H}(y) u_i^y = \lambda_i^y \Sigma_{\text{pr}}^{-1} u_i^y$
2. Assemble $U_r = [u_1^y, \ldots, u_r^y] \in \mathbb{R}^{d \times r}$

In the end we get

$$D_{\text{KL}}\left(\pi^y \middle\| \tilde{\pi}_f^y\right) \leq \frac{\kappa}{2}(\lambda_{r+1}^y + \cdots + \lambda_d^y)$$

Building $U_r \in \mathbb{R}^{d\times r}$ by minimizing the bound

Principal Component Analysis of $\nabla \log f^y$

Bound on D_{KL} relies on the Gram matrix

$$\mathbf{H}(y) = \int (\nabla \log f^y)\,(\nabla \log f^y)^T \;\, \mathrm{d}\pi^y$$

Find U_r that minimizes the **truncation residual**

$$\mathcal{R}(H(y), U_r) = \mathrm{tr}\left[\Sigma_r(I_d - U_r U_r^T)\mathbf{H}(y)(I_d - U_r U_r^T)\right]$$

1. Solve the generalized eigenvalue problem $\mathbf{H}(y)u_i^y = \lambda_i^y \Sigma_{\mathrm{pr}}^{-1} u_i^y$
2. Assemble $U_r = [u_1^y, \dots, u_r^y] \in \mathbb{R}^{d\times r}$

In the end we get

$$D_{\mathrm{KL}}\left(\pi^y \middle|\middle| \tilde{\pi}_f^y\right) \le \frac{\kappa}{2}(\lambda_{r+1}^y + \cdots + \lambda_d^y)$$

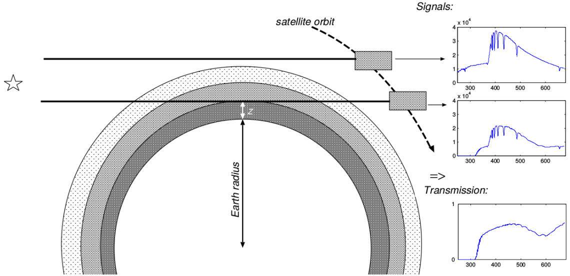
Alternative: coordinate selection $U_r x = x_\tau$ for some $\tau \subset \{1, \dots, d\}$

$$D_{\mathrm{KL}}\left(\pi^y \middle|\middle| \tilde{\pi}_f^y\right) \le \frac{\kappa}{2} \sum_{i \notin \tau} \mathbf{H}(y)_{ii} (\Sigma_{\mathrm{pr}})_{ii}$$

Benchmark GOMOS: atmospheric remote sensing (eg [Haario et al. 2004])

- Estimate gas densities $x = \rho^{\text{gas}}(z)$ from transmission spectra $y_\omega(z)$
- Beer's law:

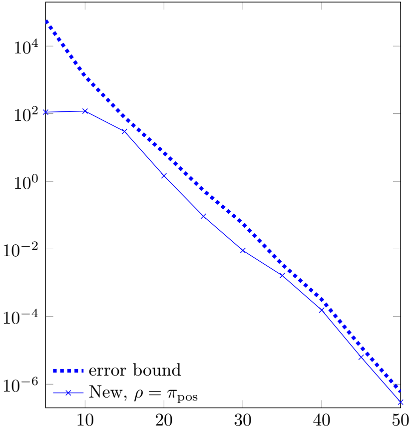
$$y_\omega(z) = \exp\left(-\int_{\text{light path}} \sum_{\text{gas}} \alpha_\omega^{\text{gas}}(z(\zeta))\, \rho^{\text{gas}}(z(\zeta))\, \mathrm{d}\zeta\right) + \xi$$



- Log-normal prior $\mathcal{N}(\mu_{\text{pr}}, \Sigma_{\text{pr}})$ with squared exponential kernel covariance
- After discretization of the atmosphere, $\dim(x) = 200$.

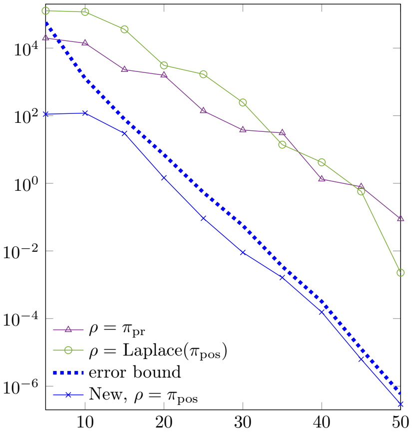
Results [Zahm, Cui, Law, Spantini & Marzouk 2022]

$$D_{\text{KL}}(\pi^y || \tilde{\pi}_f^y) = \mathit{function}(r)$$



$$\mathbf{H}(y) = \int (\nabla \log f^y)(\nabla \log f^y)^T \, \mathrm{d}\pi^y$$

$$D_{\mathrm{KL}}(\pi^y || \tilde{\pi}_f^y) = function(r)$$



$$\mathbf{H}^{(\rho)}(y) = \int (\nabla \log \mathcal{L}_y)(\nabla \log \mathcal{L}_y)^T \, \mathrm{d}\rho$$

Further error analysis (in Hellinger divergence)

$$\tilde{\pi}^y(x) \propto \tilde{f}^y(U_r^T x)\mu(x)$$

Optimal approximation given U_r [Cui, Dolgov & Zahm, 2022]

Given U_r, the function

$$\tilde{f}^y(x_r) \equiv g_r^y(x_r)^2, \quad g_r^y(x_r) = \mathbb{E}_{X\sim\mu}\left(\sqrt{f^y}(X)\middle| U_r^T X = x_r\right)$$

minimizes $D_{\mathrm{H}}\left(\pi^y \middle|\middle| \tilde{\pi}^y\right)$. Then, $\tilde{\pi}^y(x)$ writes

$$\tilde{\pi}_g^y(x) = \pi_g^y(x_r)\mu(x_\perp|x_r), \qquad \pi_g^y(x_r) = \frac{1}{Z_g} g_r^y(x_r)^2 \mu(x_r)$$

Some useful bounds [Cui & Tong 2022]

Assume $\mu = \mathcal{N}(m_{\mathrm{pr}}, \Sigma_{\mathrm{pr}})$. We have

$$D_{\mathrm{H}}^2\left(\pi^y \middle|\middle| \tilde{\pi}_g^y\right) \le \frac{1}{Z} \int \mathrm{var}_\mu\left[\sqrt{f^y}(X)\middle| U_r^T X = x_r\right] \,\mathrm{d}\mu(x_r)$$

$$D_{\mathrm{H}}^2\left(\pi^y \middle|\middle| \tilde{\pi}_f^y\right) \le \frac{1}{Z} \int \mathrm{var}_\mu\left[\sqrt{f^y}(X)\middle| U_r^T X = x_r\right] \,\mathrm{d}\mu(x_r)$$

$$D_{\mathrm{H}}\left(\pi^y \middle|\middle| \tilde{\pi}_g^y\right) \le D_{\mathrm{H}}\left(\pi^y \middle|\middle| \tilde{\pi}_f^y\right)$$

Further error analysis (in Hellinger divergence)

Bound on conditional variance [Cui & Tong 2022]

By Poincaré inequality, we have

$$\frac{1}{Z}\int \mathrm{var}_{\mu}\left[\sqrt{f^{y}}(X)\middle| U_r^T X = x_r\right] \mathrm{d}x_r \leq \frac{C}{4}\int \|(I_d - U_r U_r^T)\nabla \log f^{y}\|_{\Sigma_{\mathrm{pr}}}^2 \, \mathrm{d}\pi^{y}$$

Leads to a similar upper bound on Hellinger as that for the KL divergence.

$$D_{\mathrm{H}}^2\left(\pi^{y}\middle|\middle|\tilde{\pi}^{y}_{\{f,g\}}\right) \leq \frac{C}{4}\mathrm{tr}\left[\Sigma_r(I_d - U_r U_r^T)\mathbf{H}(y)(I_d - U_r U_r^T)\right]$$

Building $U_r \in \mathbb{R}^{d\times r}$ by minimizing the bound

Principal Component Analysis of $\nabla \log f^y$

Similar to the KL case, given the same Gram matrix

$$\mathbf{H}(y) = \int (\nabla \log f^y)(\nabla \log f^y)^T \, \mathrm{d}\pi^y$$

Leading eigenvectors of $\mathbf{H}(y) u_i^y = \lambda_i^y \Sigma_{\mathrm{pr}}^{-1} u_i^y$ defines $U_r = [u_1^y, \ldots, u_r^y]$ that minimizes the **truncation residual**

$$\mathcal{R}(H(y), U_r) = \mathrm{tr}\left[\Sigma_r (I_d - U_r U_r^T)\mathbf{H}(y)(I_d - U_r U_r^T)\right]$$

In the end we get

$$D_{\mathrm{H}}\left(\pi^y \middle|\middle| \tilde{\pi}^y_{\{f,g\}}\right) \leq \frac{C}{2}\sqrt{\lambda_{r+1}^y + \cdots + \lambda_d^y}$$

Alternative: coordinate selection $U_r x = x_\tau$ for some $\tau \subset \{1, \ldots, d\}$

$$D_{\mathrm{H}}\left(\pi^y \middle|\middle| \tilde{\pi}^y_{\{f,g\}}\right) \leq \frac{C}{2}\left(\sum_{i \notin \tau} \mathbf{H}(y)_{ii} (\Sigma_{\mathrm{pr}})_{ii}\right)^{\frac{1}{2}}$$

Monte Carlo approximations

Given U_r built from the Gram matrix $\mathbf{H}(y)$:

Conditional expectations

$$f_r^y(x_r) = \mathbb{E}_{X\sim\mu}\left(f^y(X) \middle| U_r^T X = x_r \right)$$
$$g_r^y(x_r) = \mathbb{E}_{X\sim\mu}\left(\sqrt{f^y}(X) \middle| U_r^T X = x_r \right)$$

Optimal approximations

$$\text{KL}: \quad \tilde{\pi}_f^y(x) = \frac{1}{Z} f_r^y(x_r)\mu(x_r)\mu(x_\perp|x_r)$$
$$\text{Hellinger}: \quad \tilde{\pi}_g^y(x) = \frac{1}{Z_g} g_r^y(x_r)^2\mu(x_r)\mu(x_\perp|x_r)$$

- How to approximate the conditional expectation in $f_r^y(x_r)$ and $g_r^y(x_r)$?
- How to approximate the Gram matrix $\mathbf{H}(y)$ and the basis U_r?
- What are the approximation errors?

Monte Carlo approximation of conditional expectations

Given conditional prior samples $x_\perp^{(j)} \sim \mu(x_\perp | x_r), j = 1, \ldots, N$, we have

$$f_r^y(x_r) = \mathbb{E}_{X\sim\mu}\left(f^y(X) \middle| U_r^T X = x_r \right) \approx \frac{1}{N}\sum_{j=1}^N f^y(U_r x_r + U_\perp x_\perp^{(j)}) \equiv f_N^y(x_r)$$

$$g_r^y(x_r) = \mathbb{E}_{X\sim\mu}\left(\sqrt{f^y}(X) \middle| U_r^T X = x_r \right) \approx \frac{1}{N}\sum_{j=1}^N \sqrt{f^y}(U_r x_r + U_\perp x_\perp^{(j)}) \equiv g_N^y(x_r),$$

Monte Carlo estimates of the optimal approximations

$$\tilde{\pi}_f^y(x) \approx \pi_{f,N}^y(x) \propto f_N^y(x_r)\mu(x_r)\mu(x_\perp|x_r)$$

$$\tilde{\pi}_g^y(x) \approx \pi_{g,N}^y(x) \propto g_N^y(x_r)^2\mu(x_r)\mu(x_\perp|x_r)$$

Bound the sampling error [Cui & Tong 2022]

$$\mathbb{E}\left[D_{\mathrm{H}}(\pi_f^y \| \pi_{f,N}^y)\right] = \mathcal{O}\left(\frac{1}{\sqrt{N}}\sqrt{\mathcal{R}(H(y), U_r)}\right)$$

$$\mathbb{E}\left[D_{\mathrm{H}}(\pi_g^y \| \pi_{g,N}^y)\right] = \mathcal{O}\left(\frac{1}{\sqrt{N}}\sqrt{\mathcal{R}(H(y), U_r)}\right)$$

- N can be small for a small truncation residual $\mathcal{R}(H(y), U_r)$
- $\pi_{g,N}^y$ has almost the same accuracy (in D_{H}) as $\pi_{f,N}^y$ in practice
- Sharp estimate on D_{KL} is still not available

Monte Carlo approximation of the Gram matrix

Sample-based estimation:

1. Monte Carlo approximation

$$\hat{\mathbf{H}}(y) = \frac{1}{M} \sum_{i=1}^{M} \left(\nabla \log f^y(X^{(i)}) \right) \left(\nabla \log f^y(X^{(i)}) \right)^T \frac{\pi^y((X^{(i)}))}{\tilde{\pi}^y((X^{(i)}))}, \quad X^{(i)} \sim \tilde{\pi}^y$$

   Iterative adaptation: $\pi_{\text{pr}} \to H^{(\pi_{\text{pr}})} \to \tilde{\pi}^y \to H^{(\tilde{\pi}^y)} \to \dots$

2. Solve the generalized eigenvalue problem $\hat{\mathbf{H}}(y) u_i^y = \hat{\lambda}_i^y \Sigma_{\text{pr}}^{-1} \hat{u}_i^y$
3. Assemble $\hat{U}_r = [\hat{u}_1^y, \dots, \hat{u}_r^y] \in \mathbb{R}^{d \times r}$

Given a (random) $\hat{U}_r$, what is $\mathcal{R}(H(y), \hat{U}_r)$? [Cui & Tong 2022]

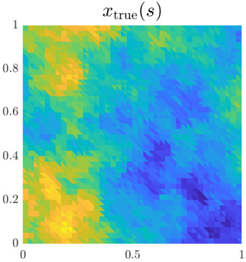
$$\mathbb{E}\left[\mathcal{R}(H(y), \hat{U}_r) \right] \leq \sum_{i=r+1}^{d} \hat{\lambda}_i^y + \frac{\sqrt{r \operatorname{var}(\mathbf{H}(y))}}{\sqrt{M}}$$

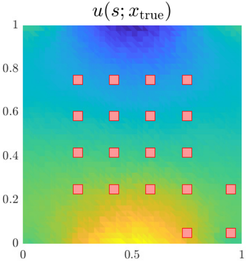
- Eigenvalues $\hat{\lambda}_i^y$, rank r, and sample size M are known
- $\operatorname{var}(\mathbf{H}(y))$ is a constant (unknown)
- For linear inverse problems, the bound is independent of the dimension d
- **Does not relies on the spectral gap assumption** of $\mathbf{H}(y)$, which is a typical assumption but often does not hold in practice

A numerical example: elliptic PDE

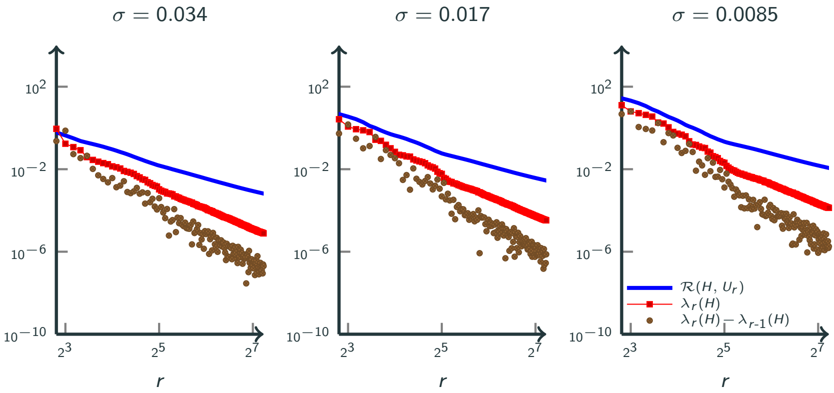
$$-\nabla \cdot \Big(\kappa(s)\nabla u(s)\Big) = f(s), \quad s \in [0,1]^2$$

- Boundary conditions: $u|_{s_1=0} = 1$ and $u|_{s_1=1} = 0$, no flux on others
- Parameter: $x(s) = \log \kappa(s)$
- Data: $y = (u(s_1), \ldots, u(s_m)) + \mathcal{N}(0, \sigma^2 \mathbf{I})$ (Gaussian likelihood)
- Gaussian process prior: $K(s, s') = \exp(-\frac{1}{\ell}\|s - s'\|)$



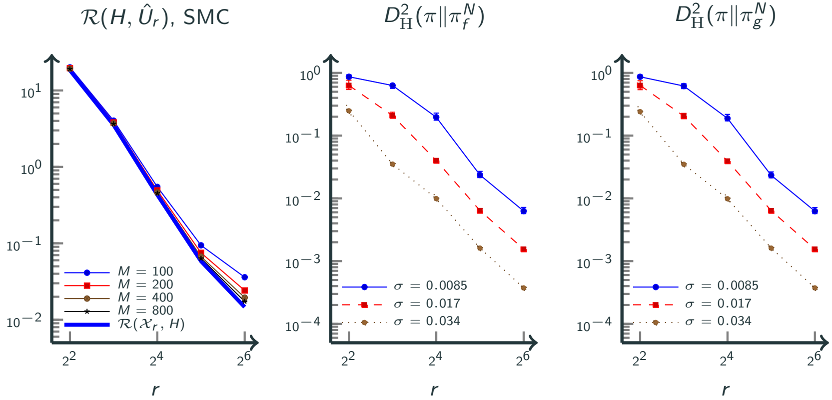


Spectral gap of $\mathbf{H}(y)$ decays with r



Results [Cui & Tong 2022]

- SMC adaptive estimation of $\mathbf{H}(y)$ and $\hat{U}_r$, with different sample size
- Using $N = 4$ for conditional expectations
- Negligible variances



Data-free dimension reduction $U_r = \cancel{U_r(y)}$

Recall that, in the Bayesian perspective, the observed data y is a **realization** of a random variable

$$Y \sim \pi_{\text{data}}$$

Objective

Find a $U_r = \cancel{U_r(Y)}$ such that

$$D_{(\cdot)}(\pi^Y || \tilde{\pi}_f^Y) \leq tol \tag{1}$$

in high probability (w.r.t. Y). Here $(\cdot) = \{\mathrm{KL}, \mathrm{H}\}$.

By Markov inequality,

$$\mathbb{E}\Big(D_{(\cdot)}(\pi^Y || \tilde{\pi}_f^Y)\Big) \leq \varepsilon$$

is sufficient to ensure (1) with probability greater than $1 - \varepsilon / tol$.

Offline

1. Compute
$$\mathbf{H} = \mathbb{E}(\mathbf{H}(Y))$$
2. Solve the generalized eigenvalue problem $\mathbf{H}u_i = \lambda_i \Sigma_{\mathrm{pr}} u_i$ and let
$$U_r = [u_1, \dots, u_r] \in \mathbb{R}^{d \times r}$$

Online

3. Receive a realization y of Y,
4. Compute the optimal reduced likelihood $f_r^y = \mathbb{E}_{X\sim\mu}\left(f^y(X) \middle| U_r^T X = x_r\right)$
5. Assemble the posterior approximation $\tilde{\pi}_f^y \propto f_r^y(U_r^T x)\mu(x)$

Proposition

Assume $\mu = \mathcal{N}(m_{\mathrm{pr}}, \Sigma_{\mathrm{pr}})$. The above procedure yields
$$\mathbb{E}\Big(D_{\mathrm{KL}}(\pi^Y || \tilde{\pi}_f^Y)\Big) \leq \frac{1}{2}(\lambda_{r+1} + \cdots + \lambda_d)$$

Similar results can be obtained for D_{H} by convexity.

Proposition [Cui & Zahm 2021]

$$\mathbf{H} = \int \mathcal{I}(x)\,\mathrm{d}\mu(x)$$

where $\mathcal{I}(x)$ is the **Fisher information matrix** of the likelihood $f^y(x) \propto \pi(y|x)$ defined by

$$\mathcal{I}(x) = \int \nabla \log f^y(x) \nabla \log f^y(x)^T \,\pi(y|x)\mathrm{d}y$$

Explicit expression on the Fisher information matrix when:

- **Gaussian likelihood:** $f^y(x) = \exp(-\frac{1}{2}\|G(x) - y\|^2_{\Gamma_{\mathrm{obs}}^{-1}})$

$$\boxed{H = \int \nabla G(x)^T \Gamma_{\mathrm{obs}}^{-1} \nabla G(x)^T \,\mathrm{d}\mu(x)}$$

- **Poisson likelihood:** $f^y(x) = \prod_{i=1}^m \frac{G_i(x)^{y_i} \exp(-G_i(x))}{y_i!}$

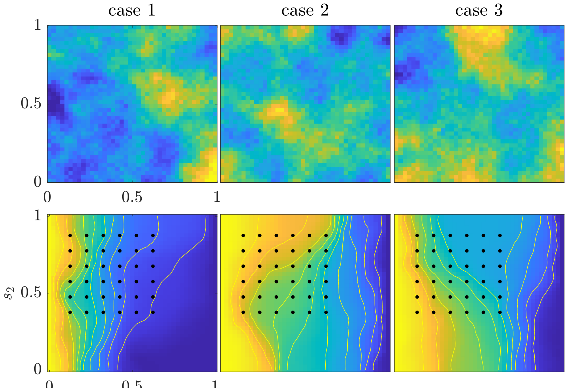
$$\boxed{H = \int \nabla G(x)^T \mathrm{diag}\Big(G_1(x), \dots, G_m(x)\Big)^{-1} \nabla G(x)^T \,\mathrm{d}\mu(x)}$$

- ...

A numerical example: elliptic PDE

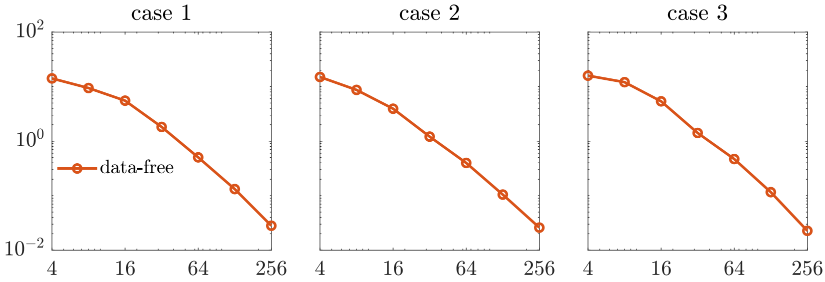
$$-\nabla \cdot \Big(\kappa(s)\nabla u(s)\Big) = f(s), \quad s \in [0,1]^2$$

- Boundary conditions: $u|_{s_1=0} = 1$ and $u|_{s_1=1} = 0$, no flux on others
- Parameter: $x(s) = \log \kappa(s)$
- Data: $y = (u(s_1), \ldots, u(s_m)) + \mathcal{N}(0, \sigma^2 \mathbf{I})$ (Gaussian likelihood)
- Gaussian prior: $-\Delta x + \gamma x = \mathcal{W}$ with $\mathcal{W}$ = white noise and $\gamma = 10$



Elliptic PDE: results for three different observations $Y^{(1)}, Y^{(2)}, Y^{(3)}$

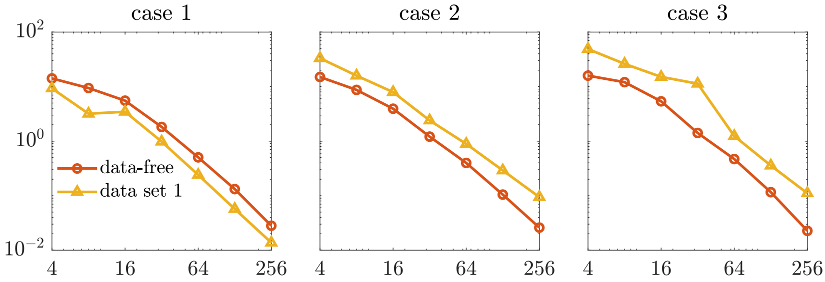
$$D_{\mathrm{KL}}(\pi^{Y^{(i)}} || \tilde{\pi}_f^{Y^{(i)}}) = function(r)$$



- data-free: U_r computed via $\mathbf{H} = \mathbb{E}(\mathbf{H}(Y))$

Elliptic PDE: results for three different observations $Y^{(1)}, Y^{(2)}, Y^{(3)}$

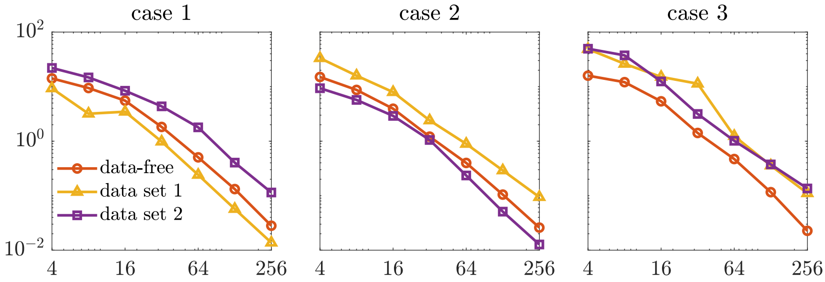
$$D_{\mathrm{KL}}(\pi^{Y^{(i)}} || \tilde{\pi}_f^{Y^{(i)}}) = function(r)$$



- data-free: U_r computed via $\mathbf{H} = \mathbb{E}(\mathbf{H}(Y))$
- data set 1: $U_r(Y^{(1)})$ computed via $\mathbf{H}(Y^{(1)})$

Elliptic PDE: results for three different observations $Y^{(1)}, Y^{(2)}, Y^{(3)}$

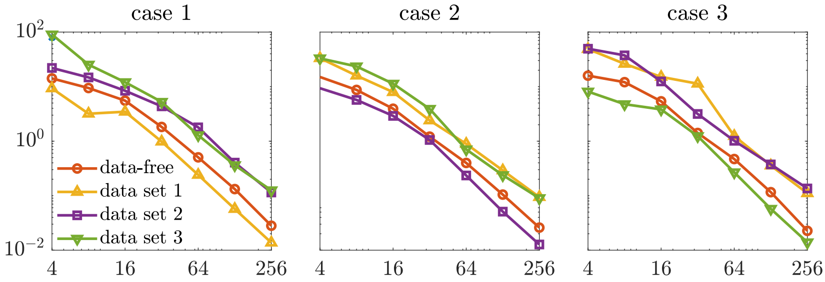
$$D_{\mathrm{KL}}(\pi^{Y^{(i)}} || \tilde{\pi}_f^{Y^{(i)}}) = function(r)$$



- data-free: U_r computed via $\mathbf{H} = \mathbb{E}(\mathbf{H}(Y))$
- data set 1: $U_r(Y^{(1)})$ computed via $\mathbf{H}(Y^{(1)})$
- data set 2: $U_r(Y^{(2)})$ computed via $\mathbf{H}(Y^{(2)})$

Elliptic PDE: results for three different observations $Y^{(1)}, Y^{(2)}, Y^{(3)}$

$$D_{\mathrm{KL}}(\pi^{Y^{(i)}} || \tilde{\pi}_f^{Y^{(i)}}) = \mathit{function}(r)$$



- data-free: U_r computed via $\mathbf{H} = \mathbb{E}(\mathbf{H}(Y))$
- data set 1: $U_r(Y^{(1)})$ computed via $\mathbf{H}(Y^{(1)})$
- data set 2: $U_r(Y^{(2)})$ computed via $\mathbf{H}(Y^{(2)})$
- data set 3: $U_r(Y^{(3)})$ computed via $\mathbf{H}(Y^{(3)})$

A sampling strategy

Sample from the approximate posterior

For a given U_r, consider the marginal posterior

$$\pi_r^y(x_r) = \frac{1}{Z} \underbrace{\left(\int f^y(U_r x_r + U_\perp x_\perp) \mu(x_\perp | x_r) \mathrm{d}x_\perp \right)}_{f_r^y(x_r) = \mathbb{E}_{X\sim\mu}\left(f^y(X) \middle| U_r^T X = x_r\right)} \mu(x_r)$$

where f_r^y is the optimal likelihood approximation in KL

Apply Monte Carlo approximation

$$f_r^y(x_r) \approx f_N^y(x_r) = \frac{1}{N} \sum_{j=1}^{N} f^y(U_r x_r + U_\perp x_\perp^{(j)}), \quad x_\perp^{(j)} \sim \mu(x_\perp | x_r)$$

Leads to the approximate posterior

$$\tilde{\pi}^y(x) \propto \underbrace{f_N^y(x_r) \mu(x_r)}_{\pi_N^y(x_r)} \; \mu(x_\perp | x_r)$$

Sample from the approximate posterior

For a given U_r, consider the marginal posterior

$$\pi_r^y(x_r) = \frac{1}{Z} \underbrace{\left(\int f^y(U_r x_r + U_\perp x_\perp) \mu(x_\perp | x_r) \mathrm{d}x_\perp \right)}_{f_r^y(x_r) = \mathbb{E}_{X\sim\mu}\left(f^y(X) \middle| U_r^T X = x_r\right)} \mu(x_r)$$

where f_r^y is the optimal likelihood approximation in KL

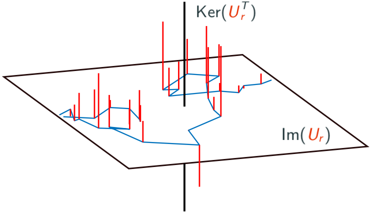
Apply Monte Carlo approximation

$$f_r^y(x_r) \approx f_N^y(x_r) = \frac{1}{N} \sum_{j=1}^{N} f^y(U_r x_r + U_\perp x_\perp^{(j)}), \quad x_\perp^{(j)} \sim \mu(x_\perp | x_r)$$

Leads to the approximate posterior

$$\tilde{\pi}^y(x) \propto \underbrace{f_N^y(x_r)\mu(x_r)}_{\pi_N^y(x_r)} \; \mu(x_\perp | x_r)$$

1. Approximate marginal $x_r^{(i)} \sim \pi_N^y(x_r)$
2. Conditional prior $x_\perp^{(i)} \sim \mu(x_\perp | x_r^{(i)})$
3. Assemble $U_r x_r^{(i)} + U_\perp x_\perp^{(i)} \sim \tilde{\pi}^y(x)$



Sample from the ~~approximate~~ exact posterior

For a given U_r, consider the marginal posterior

$$\pi_r^y(x_r) = \frac{1}{Z} \underbrace{\left(\int f^y(U_r x_r + U_\perp x_\perp) \mu(x_\perp | x_r) \mathrm{d}x_\perp \right)}_{f_r^y(x_r) = \mathbb{E}_{X\sim\mu}\left(f^y(X) \middle| U_r^T X = x_r\right)} \mu(x_r)$$

where f_r^y is the optimal likelihood approximation in KL

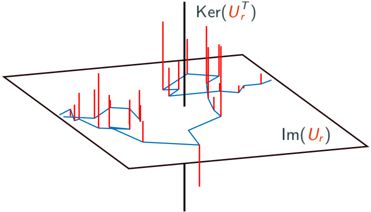
Apply Monte Carlo approximation

$$f_r^y(x_r) \approx f_N^y(x_r) = \frac{1}{N} \sum_{j=1}^{N} f^y(U_r x_r + U_\perp x_\perp^{(j)}), \quad x_\perp^{(j)} \sim \mu(x_\perp | x_r)$$

Leads to the approximate posterior

$$\tilde{\pi}^y(x) \propto \underbrace{f_N^y(x_r)\mu(x_r)}_{\pi_N^y(x_r)} \; \mu(x_\perp | x_r)$$

1. ~~Approximate~~ marginal $x_r^{(i)} \sim \pi_r^y(x_r)$
2. ~~Conditional prior~~ $\cancel{x_\perp^{(i)} \sim \mu(x_\perp | x_r^{(i)})}$
3. Assemble $U_r x_r^{(i)} + U_\perp x_\perp^{(i)} \sim \pi^y(x)$



Sample from the ~~approximate~~ exact posterior

Pseudo-Marginal MCMC [Andrieu & Roberts 2009] to sample from $\pi_r^y(x_r)$

$$\pi_r^y(x_r) \approx \pi_N^y(x_r) = \frac{\mu(x_r)}{N} \sum_{i=1}^{N} f^y(U_r x_r + U_\perp x_\perp^{(j)}), \quad x_\perp^{(j)} \sim \mu(x_\perp | x_r)$$

- **Low-variance** estimator by construction of U_r (N can be small)
- **Unbiased** estimator for the marginal $\pi_r^y(x_r)$
- Pseudo-Marginal trick: redraw $x_\perp^{(j)} \sim \mu(x_\perp | x_r)$ at each MCMC iteration. Then, Markov chain converges to the exact marginal: $x_r^{(i)} \sim \pi_r^y(x_r)$.

Sample from the ~~approximate~~ exact posterior

Pseudo-Marginal MCMC [Andrieu & Roberts 2009] to sample from $\pi_r^y(x_r)$

$$\pi_r^y(x_r) \approx \pi_N^y(x_r) = \frac{\mu(x_r)}{N} \sum_{i=1}^{N} f^y(U_r x_r + U_\perp x_\perp^{(j)}), \quad x_\perp^{(j)} \sim \mu(x_\perp | x_r)$$

- **Low-variance** estimator by construction of U_r (N can be small)
- **Unbiased** estimator for the marginal $\pi_r^y(x_r)$
- Pseudo-Marginal trick: redraw $x_\perp^{(j)} \sim \mu(x_\perp | x_r)$ at each MCMC iteration. Then, Markov chain converges to the exact marginal: $x_r^{(i)} \sim \pi_r^y(x_r)$.

Recycle $x_\perp^{(j)}$ to sample from the exact full posterior $\pi^y(x)$ [Cui & Zahm 2021]

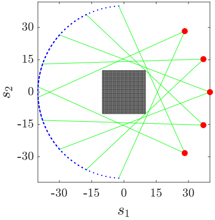
Instead of drawing $x_\perp^{(i)} \sim \mu(x_\perp | x_r^{(i)})$, pick $\tilde{x}_\perp^{(i)} \in \{x_\perp^{(1)}, \dots, x_\perp^{(N)}\}$ at random according to the likelihood weights

$$\{f^y(U_r x_r^{(i)} + U_\perp x_\perp^{(1)}), \dots, f^y(U_r x_r^{(i)} + U_\perp x_\perp^{(N)})\}$$

Then $U_r x_r^{(i)} + U_\perp \tilde{x}_\perp^{(i)} \sim \pi^y(x)$

A numerical example: X-ray tomography with Poisson data

Identify the density of a material in a domain of interest (blue square) using five X-ray sources (red points) and $m = 100$ sensors (blue points)



- Data: $Y \in \mathbb{N}^m$ integer-valued vector (number of incident photons)
- **Poisson likelihood** of the form

$$f^y(x) = \prod_{i=1}^{m} \frac{G_i(x)^{y_i} \exp(-G_i(x))}{y_i!}$$

where the forward model $G(x)$ stems from Beer's law.

- **Besov-1 (Laplace) prior**

$$\mu(x) \propto \prod_{i=1}^{d=64^2} \exp(-\lambda |x_i|)$$

- We use **coordinate selection** to reduce the dimension.

MCMC using H-MALA[1] as a baseline to sample the exact posterior

We use **Integrated Auto Correlation Time (IACT)** to measure the mixing performances of the MCMC.

		IACT	$\sqrt{\mathrm{var}}[\log f_N^y]$
$N=2$	$r=16$	85.1±2.7	1.54±0.02
	$r=32$	54.1±3.1	0.61±.007
	$r=48$	49.4±2.6	0.45±.002
$N=5$	$r=16$	60.0±6.2	0.93±.006
	$r=32$	47.6±2.5	0.39±.004
	$r=48$	46.5±1.4	0.29±.001

IACT of the full-dimensional H-MALA: 95.9 ± 3.3

[1] Hessian-preconditioned Metropolis-Adjusted Langevin Algorithm

Conclusion

Conclusion

- Detect the low effective dimensionality of Bayesian inverse problems by:
    - deriving an upper bound on the error (KL-divergence and Hellinger)
    - minimizing the bound ($\equiv$ PCA on $\nabla \log f^y$)
- Upper bounds on sampling errors in building the subspace and likelihood approximation
- Extension to data-free:
    - find directions that *will be* informed by data with high probability
    - provides bound on KL-divergence in expectation
- Exact subspace MCMC computations

References

[Cui & Zahm 2021] *Data-free likelihood-informed dimension reduction for bayesian inverse problems*, Inverse Problems, 37 (4), 045009.

[Cui & Tong 2022] *A unified performance analysis of likelihood-informed subspace methods*, Bernoulli 28 (4), 2788–2815.

[Zahm, Cui, Law, Spantini & Marzouk 2022] *Certified dimension reduction in nonlinear Bayesian inverse problems*, Mathematics of Computation, 91 (336), 1789–1835.

[Cui, Law & Marzouk 2016] *Dimension-independent likelihood-informed MCMC*, Journal of Computational Physics, 304 (1), 109–137.

[Cui, Martin, Marzouk, Solonen & Spantini 2014] *Likelihood-informed dimension reduction for nonlinear inverse problems*, Inverse Problems, 30 (11), 114015.