BayesCG: A Probabilistic Numeric Linear Solver

Ilse C.F. Ipsen

NC STATE UNIVERSITY

Raleigh, NC, USA

Research supported in part by NSF DMS RTG and FRG

Jon Cockayne, University of Southampton, UK
Joey Hart, Sandia National Laboratories, USA
Chris Oates, University of Newcastle, UK
Tim Reid, North Carolina State University, USA

After this talk: Tim Reid and Johnathan Wenger
Implementing Probabilistic Linear Solvers

Probabilistic Numerics

- Statistical treatment of approximation errors in deterministic numerical methods:

  Assign probability distributions to quantities of interest

  Express methods as probabilistic inference

- Model uncertainty due to limited computational resources

  (Truncation errors in discretizations, early termination of iterative methods)

But why??? Want error estimates that

- are more informative than traditional, pessimistic bounds
- can be propagated through computational pipelines

From the UQ perspective:

Our approach is an instance of model discrepancy
with epistemic uncertainties [Ralph Smith 2014]

Modern perspectives on Probabilistic Numerics

- Inverse Problems: A Bayesian Perspective
  A.M. Stuart (2010)
- A Call to Arms for Probabilistic Numerical Methods
  P. Hennig, M.A. Osborne, M. Girolami (2015)
- Probabilistic Interpretation of Linear Solvers
  P. Hennig (2015)
- Bayesian Probabilistic Numerical Methods
  J. Cockayne, C.J. Oates, T.J. Sullivan, M. Girolami (2019)
- A Modern Retrospective on Probabilistic Numerics
  C.J. Oates, T.J. Sullivan (2019)

Probabilistic Numeric Linear Solvers

- P. Hennig (2015): Probabilistic Interpretation of Linear Solvers, *SIAM J. Optim.*
- S. Bartels, P. Hennig (2015): Probabilistic Approximate Least-Squares, *Proc. Machine Learning Research*
- F. Schäfer, T.J. Sullivan, H. Owhadi (2021): Compression, Inversion, and Approximate PCA of Dense Kernel Matrices at Near-Linear Computational Complexity, *Multiscale Model. Simul.*
- J. Cockayne, C.J. Oates, I.C.F. Ipsen, M. Girolami (2019): A Bayesian Conjugate Gradient Method, *Bayesian Anal.*
- S. Bartels, J. Cockayne, I.C.F. Ipsen, P. Hennig (2019): Probabilistic Linear Solvers: A Unifying View, *Stat. Comput.*
- J. Cockayne, I. C. F. Ipsen, C.J. Oates, T.W. Reid (2021): Probabilistic Iterative Methods for Linear Systems, *J. Mach. Learn. Res.*
- T.W. Reid, I. C. F. Ipsen, J. Cockayne, C. J. Oates, C. J. (2020): BayesCG as an Uncertainty Aware Version of CG,*arXiv:2008.03225*

Overview

1. Systems of linear equations
2. Probabilistic linear system solution with BayesCG
3. Prior distributions for BayesCG
4. Low-rank approximate Krylov posteriors

   We are dispensing with explicit priors

   Implicit priors customized to linear system

5. Application: UQ in PDE-constrained optimization

   Piping posteriors from 1. linear solve into 2. solve

   Probe uncertainty in downstream computation

6. Take-home message

1. Systems of linear equations

Systems of linear equations

Solve for d unknowns in d equations

$$\boldsymbol{Ax} = \boldsymbol{b}$$

Matrix $\boldsymbol{A} \in \mathbb{R}^{d \times d}$ is symmetric positive definite (spd)

$$\boldsymbol{A}^T = \boldsymbol{A} \qquad \text{and} \qquad \boldsymbol{x}^T\boldsymbol{Ax} > 0 \quad \text{for } \boldsymbol{x} \neq 0$$

Unique solution $\boldsymbol{x}_* = \boldsymbol{A}^{-1}\boldsymbol{b}$

- If $\boldsymbol{A}$ is dense or has small dimension[1], use direct method
  Cholesky factorization
- If $\boldsymbol{A}$ is sparse or has large dimension, use iterative method
  Residual $\boldsymbol{Ax} - \boldsymbol{b}$ is gradient of optimization problem

[1] Laptop: $d \leq 10^5$

Solution of spd system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ as optimization problem[2]

$$\min_{\boldsymbol{x}\in\mathbb{R}^d} \phi(\boldsymbol{x}) \qquad \text{where} \qquad \phi(\boldsymbol{x}) \equiv \tfrac{1}{2}\,\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x} - \boldsymbol{x}^T\boldsymbol{b}$$

Function $\phi(\boldsymbol{x})$ is convex, has gradient $\nabla\phi(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}$

- Solution: If $\boldsymbol{A}\boldsymbol{x}_* = \boldsymbol{b}$ then $\phi(\boldsymbol{x}_*) = -\tfrac{1}{2}\,\boldsymbol{x}_*^T\boldsymbol{b}$
- Approximation: If $\boldsymbol{A}\boldsymbol{x}_c \approx \boldsymbol{b}$ then

$$\begin{aligned}\phi(\boldsymbol{x}_c) &= (\phi(\boldsymbol{x}_c) - \phi(\boldsymbol{x}_*)) + \phi(\boldsymbol{x}_*)\\ &= \tfrac{1}{2}\underbrace{(\boldsymbol{x}_c - \boldsymbol{x}_*^T)\boldsymbol{A}(\boldsymbol{x}_c - \boldsymbol{x}_*)}_{\|\boldsymbol{x}_* - \boldsymbol{x}_c\|_{\boldsymbol{A}}^2} + \phi(\boldsymbol{x}_*)\end{aligned}$$

- Measure error in $\boldsymbol{A}$-norm: $\|\boldsymbol{x}_* - \boldsymbol{x}_c\|_{\boldsymbol{A}}$ where $\|\boldsymbol{x}\|_{\boldsymbol{A}}^2 \equiv \boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x}$

[2] Golub and Van Loan: Matrix Computations, 4th edition (2013)

Iterative methods for spd systems: Steepest descent

Follow direction of negative gradient

- User-specified initial guess $\boldsymbol{x}_0$
- Next iterate $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \nabla\phi(\boldsymbol{x}_k)$ $\quad \alpha_k \equiv \frac{\nabla\phi(\boldsymbol{x}_k)^T \nabla\phi(\boldsymbol{x}_k)}{\nabla\phi(\boldsymbol{x}_k)^T \boldsymbol{A} \nabla\phi(\boldsymbol{x}_k)}$
- Global convergence

$$\|\boldsymbol{x}_* - \boldsymbol{x}_k\|_{\boldsymbol{A}} \leq \left(1 - \frac{1}{\kappa(\boldsymbol{A})}\right)^{k/2} \|\boldsymbol{x}_* - \boldsymbol{x}_0\|_{\boldsymbol{A}}$$

Condition number $\kappa(\boldsymbol{A}) \equiv \|\boldsymbol{A}\|_2 \, \|\boldsymbol{A}^{-1}\|_2$

- But: Convergence is too slow
- Why? Steepest descent minimizes $\|\boldsymbol{x}_* - \boldsymbol{x}\|_{\boldsymbol{A}}^2$ over one-dimensional subspace $\boldsymbol{x}_k - \operatorname{span}\{\nabla\phi(\boldsymbol{x}_k)\}$

Conjugate Gradient (CG) method [Hestenes, Stiefel 1952]

- User-specified initial guess $\boldsymbol{x}_0$

$$\text{Solve} \quad \boldsymbol{A}\,(\boldsymbol{x}_* - \boldsymbol{x}_0) = \underbrace{\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0}_{\boldsymbol{r}_0}$$

- Iteration k
  CG minimizes $\|\boldsymbol{x}_* - \boldsymbol{x}\|_{\boldsymbol{A}}^2$ over k-dimensional Krylov space

$$\boldsymbol{x}_0 + \underbrace{\text{span}\{\boldsymbol{r}_0, \boldsymbol{A}\boldsymbol{r}_0, \ldots, \boldsymbol{A}^{k-1}\boldsymbol{r}_0\}}_{\mathcal{K}_k(\boldsymbol{A}, \boldsymbol{r}_0)}$$

- Orthogonal projection: $\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_k \perp \mathcal{K}_k(\boldsymbol{A},\, \boldsymbol{r}_0)$
- Global convergence

$$\|\boldsymbol{x}_* - \boldsymbol{x}_k\|_{\boldsymbol{A}} \le 2 \left(\frac{\sqrt{\kappa(\boldsymbol{A})} - 1}{\sqrt{\kappa(\boldsymbol{A})} + 1}\right)^k \|\boldsymbol{x}_* - \boldsymbol{x}_0\|_{\boldsymbol{A}}$$

- [Hennig 2015] CG in the context of quasi-Newton methods

Iterative solvers in practice

Relation between error and residual

$$\underbrace{\frac{\|\boldsymbol{x}_* - \boldsymbol{x}_k\|}{\|\boldsymbol{x}_*\|}}_{\text{what we want}} \leq \underbrace{\kappa(\boldsymbol{A})}_{\text{amplifier}} \underbrace{\frac{\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_k\|}{\|\boldsymbol{b}\|}}_{\text{we can control}}$$

- Floating point arithmetic defined by relative errors
- Condition number $\kappa(\boldsymbol{A}) = \|\boldsymbol{A}\|_2 \|\boldsymbol{A}^{-1}\|_2$ amplifies noise in input, and roundoff errors (does not depend on solver)
- Residual $\boldsymbol{r}_k = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_k$ indicates accuracy of solver
- User-specified stopping tolerance η

$$\text{Solver stops once} \quad \frac{\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_k\|}{\|\boldsymbol{b}\|} \leq \eta$$

Numerically stable direct methods in IEEE float64 deliver $\eta \approx 10^{-16}$

Our approach:
Replace error bounds by probability distributions

Where to place the distribution?

- Matrix-based inference

  Hennig (2015), Bartels, Hennig (2015), Bartels, Cockayne, I., Hennig (2019)

- Solution-based inference

  Cockayne, Oates, I., Girolami (2019), Bartels, Cockayne, I., Hennig (2019), Cockayne, I., Oates, Reid (2021), Reid, I., Cockayne, Oates (2021)

  - Prior Distribution
    Reflects initial knowledge about $\boldsymbol{x}_*$
  - Posterior Distribution in iteration k
    Reflects knowledge about $\boldsymbol{x}_*$ after k iterations

Goals

1. Accurate error estimation for single solve
2. Error propagation through computational pipelines

2. Probabilistic Linear System Solution with BayesCG

Probabilistic linear system solution

Given: Symmetric positive definite $\boldsymbol{A} \in \mathbb{R}^{d \times d}$, vector $\boldsymbol{b} \in \mathbb{R}^{d}$
Want: Solution of $\boldsymbol{A}\boldsymbol{x}_* = \boldsymbol{b}$
Initial guess $\boldsymbol{x}_0$: $\boldsymbol{A}(\boldsymbol{x}_* - \boldsymbol{x}_0) = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0$
Solver: Computes approximations $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_k \to \boldsymbol{x}_*$

Solution-based inference

Probability distribution over solution space $\boldsymbol{x} \in \mathbb{R}^{d}$
Approximations = means of probability distribution

- User specifies Gaussian prior $\mathcal{N}(\boldsymbol{x}_0, \Sigma_0)$
  Prior reflects initial knowledge about $\boldsymbol{x}_*$

- Solver computes Gaussian posteriors $\mathcal{N}(\boldsymbol{x}_k, \Sigma_k)$
  Posteriors reflect knowledge about $\boldsymbol{x}_*$ after k iterations

Computing the posteriors in iteration $k \geq 1$

[Cockayne, Oates, Sullivan, Girolami 2019]

- User-specified prior: $X \sim \mathcal{N}(\boldsymbol{x}_0, \Sigma_0)$
- Iteration k: Search directions $\boldsymbol{S}_k = \begin{bmatrix} \boldsymbol{s}_1 & \cdots & \boldsymbol{s}_k \end{bmatrix}$
- Condition on $Y_k = \boldsymbol{S}_k^T \boldsymbol{A} X$
- Exists unique Bayesian method that outputs posterior

$$(X \mid Y_k = \boldsymbol{S}_k^T \underbrace{\boldsymbol{A}\,\boldsymbol{x}_*}_{\boldsymbol{b}}) \sim \mathcal{N}(\boldsymbol{x}_k,\, \Sigma_k)$$

with

$$\begin{aligned} \boldsymbol{x}_k &= \boldsymbol{x}_0 + \Sigma_0\, \boldsymbol{A}\, \boldsymbol{S}_k \left(\boldsymbol{S}_k^T \boldsymbol{A}\, \Sigma_0 \boldsymbol{A}\, \boldsymbol{S}_k \right)^{-1} \boldsymbol{S}_k^T \,(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0) \\ \Sigma_k &= \Sigma_0 - \Sigma_0\, \boldsymbol{A}\, \boldsymbol{S}_k \left(\boldsymbol{S}_k^T \boldsymbol{A}\, \Sigma_0 \boldsymbol{A}\, \boldsymbol{S}_k \right)^{-1} \boldsymbol{S}_k^T \boldsymbol{A}\, \Sigma_0 \end{aligned}$$

- Choose $\boldsymbol{A}\,\Sigma_0\,\boldsymbol{A}$-orthonormal search directions

$$\boldsymbol{S}_k^T \boldsymbol{A}\, \Sigma_0 \boldsymbol{A}\, \boldsymbol{S}_k = \boldsymbol{I}_k$$

Bayesian Conjugate Gradient Method (BayesCG)

[Cockayne, Oates, I., Girolami 2019], [Reid, I., Cockayne, Oates, 2021]

```
$\boldsymbol{r}_0 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0$    {initial residual}
$\boldsymbol{v}_1 = \boldsymbol{r}_0$    {initial search direction}
$k = 0$
while not converged do
    $k = k + 1$
    $\alpha_k = \left(\boldsymbol{r}_{k-1}^T \boldsymbol{r}_{k-1}\right) / \left(\boldsymbol{s}_k^T \boldsymbol{A}\,\Sigma_0\,\boldsymbol{A}\,\boldsymbol{s}_k\right)$
    $\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + \alpha_k\,\Sigma_0\,\boldsymbol{A}\,\boldsymbol{s}_k$    {next iterate}
    $\Sigma_k = \Sigma_{k-1} - \Sigma_0\,\boldsymbol{A}\,\boldsymbol{s}_k\,(\Sigma_0 \boldsymbol{A}\,\boldsymbol{s}_k)^T\,/(\boldsymbol{s}_k^T \boldsymbol{A}\,\Sigma_0\,\boldsymbol{A}\boldsymbol{s}_k)$
                                                    {next posterior}
    $\boldsymbol{r}_k = \boldsymbol{r}_{k-1} - \alpha_k\,\boldsymbol{A}\,\Sigma_0\,\boldsymbol{A}\,\boldsymbol{s}_k$    {next residual}
    $\beta_k = \left(\boldsymbol{r}_k^T \boldsymbol{r}_k\right) / \left(\boldsymbol{r}_{k-1}^T \boldsymbol{r}_{k-1}\right)$
    $\boldsymbol{v}_{k+1} = \boldsymbol{r}_k + \beta_k \boldsymbol{v}_k$    {next search direction}
end while
```

Red expressions do not appear in CG

Properties of BayesCG

[Cockayne, Oates, I., Girolami 2019], [Reid, I., Cockayne, Oates, 2021]

- Krylov space for iterates: $\boldsymbol{x}_k \in \mathcal{K}_k^*$

$$\mathcal{K}_k^* = \boldsymbol{x}_0 + \mathcal{K}_k \left(\Sigma_0\, \boldsymbol{A}^2,\, \Sigma_0\, \boldsymbol{A}\,(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0)\right)$$

- Error minimization
  - If Σ_0 nonsingular, then

$$\|\boldsymbol{x}_* - \boldsymbol{x}_k\|_{\Sigma_0^{-1}} = \min_{\boldsymbol{x}\in\mathcal{K}_k^*} \|\boldsymbol{x}_* - \boldsymbol{x}\|_{\Sigma_0^{-1}}$$

  - If Σ_0 singular and $\boldsymbol{x}_* - \boldsymbol{x}_0 \in \mathrm{range}(\Sigma_0)$, then

$$\boldsymbol{x}_k \in \mathrm{argmin}_{\boldsymbol{x}\in\mathcal{K}_k^*}\, (\boldsymbol{x}_* - \boldsymbol{x})\, \Sigma_0^{\dagger}\, (\boldsymbol{x}^* - \boldsymbol{x})$$

- Convergence bound for nonsingular Σ_0

$$\|\boldsymbol{x}_* - \boldsymbol{x}_k\|_{\Sigma_0^{-1}} \leq 2 \left(\frac{\sqrt{\kappa(\Sigma_0\, \boldsymbol{A}^2)}-1}{\sqrt{\kappa(\Sigma_0\, \boldsymbol{A}^2)}+1}\right)^k \|\boldsymbol{x}_* - \boldsymbol{x}_0\|_{\Sigma_0^{-1}}$$

Summary: BayesCG

Solve symmetric positive definite system $\boldsymbol{A}\boldsymbol{x}_* = \boldsymbol{b}$

- BayesCG = probabilistic extension of Conjugate Gradient
  Models uncertainty in solution $\boldsymbol{x}_*$ due to early termination
- Input: Gaussian prior $\mathcal{N}(\boldsymbol{x}_0, \Sigma_0)$
  Models initial uncertainty about solution $\boldsymbol{x}_*$
  Mean identical to initial approximation $\boldsymbol{x}_0 \approx \boldsymbol{x}_*$
- Iteration k: Gaussian posteriors $\mathcal{N}(\boldsymbol{x}_k, \Sigma_k)$
  Models uncertainty about solution $\boldsymbol{x}_*$ after k iterations
  Mean identical to kth approximation $\boldsymbol{x}_k \approx \boldsymbol{x}_*$

How to choose efficient prior distribution Σ_0?

3. Prior Distributions for BayesCG

Choices for prior distributions

- Noninformative: $\Sigma_0 = \boldsymbol{I}_d$
- Inverse: $\Sigma_0 = \boldsymbol{A}^{-1}$, BayesCG = CG
  CG = Bayesian inference with prior $\mathcal{N}(\boldsymbol{x}_0, \boldsymbol{A}^{-1})$
- Natural: $\Sigma_0 = \boldsymbol{A}^{-2}$, convergence in 1 iteration
- Preconditioner: $\Sigma_0 = (\boldsymbol{P}^T\boldsymbol{P})^{-1} \approx \boldsymbol{A}^{-2}$
- New Krylov:[3] $\Sigma_0 = \boldsymbol{V}\Phi\boldsymbol{V}^T$
  where columns of $\boldsymbol{V}$ are basis for Krylov space
- Hierarchical: $\Sigma_0 = \nu\,\hat{\Sigma}_0$ with Jeffrey's improper $p(\nu) \sim \nu^{-1}$

Priors that reproduce CG: Inverse and Krylov

Impractical academic priors
We will develop practical low-rank approximations

[3] Different from Krylov prior in [Cockayne, Oates, I., Girolami 2019]

Pointwise error estimates from distributions

Relation between approximation error and covariance

If $X \sim \mathcal{N}(\boldsymbol{x}, \Sigma)$ and $\boldsymbol{A} \in \mathbb{R}^{d \times d}$ spd, then

$$\mathbb{E}\left[\|X - \boldsymbol{x}\|_{\boldsymbol{A}}^2\right] = \mathrm{trace}(\boldsymbol{A}\,\Sigma)$$

Why? [Mathai, Provost 1992]

If $Z \sim \mathcal{N}(\boldsymbol{z}, \Sigma)$ then $\mathbb{E}\left[Z^T \boldsymbol{A} Z\right] = \mathrm{trace}(\boldsymbol{A}\,\Sigma) + \boldsymbol{z}^T \boldsymbol{A}\, \boldsymbol{z}$

Iteration k of BayesCG:

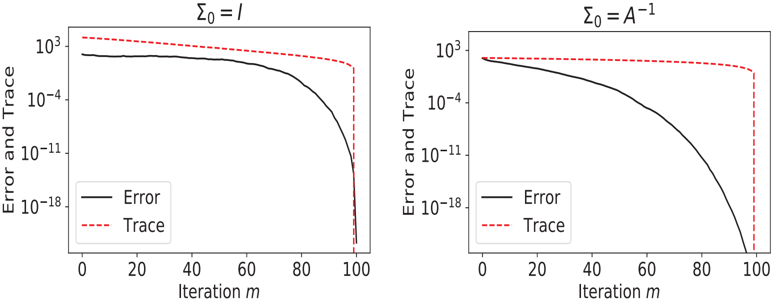
- Convergence of posterior means: $\|\boldsymbol{x}_* - \boldsymbol{x}_k\|_{\boldsymbol{A}}^2$
  posterior mean = approximation $\boldsymbol{x}_k$, traditional error
- Convergence of posterior covariances: $\mathrm{trace}(\boldsymbol{A}\,\Sigma_k)$

  If $\boldsymbol{x}_* \sim \mathcal{N}(\boldsymbol{x}_k, \Sigma_k)$, then $\mathbb{E}\left[\|\boldsymbol{x}_* - \boldsymbol{x}_k\|_{\boldsymbol{A}}^2\right] = \mathrm{trace}(\boldsymbol{A}\,\Sigma_k)$

Bayes CG under identity and inverse priors

$\boldsymbol{A} \in \mathbb{R}^{100\times 100}$ with $\kappa(\boldsymbol{A}) = 10^3$ and eigenvalues $10^{3(k-1)/99}$, $1 \le k \le 100$

Error: $\|\boldsymbol{x}_* - \boldsymbol{x}_m\|_{\boldsymbol{A}}^2$ Trace: $\operatorname{trace}(\boldsymbol{A}\Sigma_m) = \mathbb{E}\left[\|X - \boldsymbol{x}_k\|_{\boldsymbol{A}}^2\right]$



Means converge faster than posterior covariances

Error estimates from posterior covariances too pessimistic

(Impractical) Krylov prior

$$\Gamma_0 = \boldsymbol{V} \Phi \boldsymbol{V}^T$$

- n is maximal dimension (invariance index) of Krylov space

$$\mathcal{K}_n(\boldsymbol{A}, \boldsymbol{r}_0) = \text{span}\{\boldsymbol{r}_0, \boldsymbol{A}\boldsymbol{r}_0, \ldots, \boldsymbol{A}^{n-1}\boldsymbol{r}_0\}$$

- Columns of $\boldsymbol{V}$ are n normalized CG search directions

$$\boldsymbol{v}_k / \sqrt{\boldsymbol{v}_k^T \boldsymbol{A} \boldsymbol{v}_k} \qquad 1 \leq k \leq n$$

- Diagonal matrix Φ has n diagonal elements

$$\Phi_{kk} = \|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|_{\boldsymbol{A}}^2 = \gamma_k \|\boldsymbol{r}_{k-1}\|_2^2 \qquad 1 \leq k \leq n$$

where $\gamma_k \equiv \|\boldsymbol{r}_k\|_2^2 / (\boldsymbol{v}_k^T \boldsymbol{A} \boldsymbol{v}_k)$ step sizes computed in CG

(Impractical) Krylov posteriors

- Maintain Krylov prior $\Gamma_0 = \boldsymbol{V}\Phi\boldsymbol{V}^T$ in factored form $(\boldsymbol{V}, \Phi)$
- Partition the factors

$$\boldsymbol{V} = \begin{bmatrix}\boldsymbol{V}_{1:k} & \boldsymbol{V}_{k+1:n}\end{bmatrix} \qquad \Phi = \begin{bmatrix}\Phi_{1:k} & 0 \\ 0 & \Phi_{k+1:n}\end{bmatrix} \qquad 1 \leq k \leq n$$

Trailing end of Krylov prior is Krylov posterior

$$\Gamma_0 = \boldsymbol{V}_{1:k}\,\Phi_{1:k}\,\boldsymbol{V}_{1:k}^T + \underbrace{\boldsymbol{V}_{k+1:n}\,\Phi_{k+1:n}\,\boldsymbol{V}_{k+1:n}^T}_{\text{posterior } \Gamma_k}$$

- Posterior covariances Γ_k are trailing submatrices of prior
  No computation required for $(\boldsymbol{V}_{k+1:n}, \Phi_{k+1:n})$
- Krylov posterior covariances capture CG error

$$\mathrm{trace}(\boldsymbol{A}\Gamma_k) = \mathrm{trace}(\Phi_{k+1:n}) = \|\boldsymbol{x}_* - \boldsymbol{x}_k\|_{\boldsymbol{A}}^2 \qquad 1 \leq k \leq n$$

[Hestenes, Stiefel 1952]

Summary: BayesCG under (impractical) Krylov prior for solving $\boldsymbol{Ax} = \boldsymbol{b}$

Krylov prior $\mathcal{N}(\boldsymbol{x}_0, \Gamma_0)$ with $\Gamma_0 = \boldsymbol{V}\Phi\boldsymbol{V}^T$
Krylov posteriors $\mathcal{N}(\boldsymbol{x}_k, \Gamma_k)$, $k \geq 1$

- Posterior means $\boldsymbol{x}_k$ identical to CG iterates
- Posterior covariances Γ_k give pointwise error: $\mathrm{trace}(\boldsymbol{A}\Gamma_k) = \|\boldsymbol{x}_* - \boldsymbol{x}_k\|_{\boldsymbol{A}}^2$ equal to CG error
- Posterior covariances Γ_k require no computation
- Krylov prior is customized to linear system

But:
Computation of Krylov prior more expensive than solving $\boldsymbol{Ax} = \boldsymbol{b}$

4. Low-rank approximate Krylov posteriors

(Practical) approximate Krylov posteriors

At iteration k of BayesCG under Krylov prior

- Krylov posterior covariance

$$\Gamma_k = \boldsymbol{V}_{k+1:n}\,\Phi_{k+1:n}\,\boldsymbol{V}_{k+1:n}^T$$

  requires CG quantities from $n-k$ future iterations (search directions $\boldsymbol{v}_j$ and residuals $\boldsymbol{r}_j$, $k+1 \leq j \leq n$)

- Approximate posterior covariance of rank ℓ

$$\widehat{\Gamma}_k = \boldsymbol{V}_{k+1:k+\ell}\,\Phi_{k+1:k+\ell}\,\boldsymbol{V}_{k+1:k+\ell}^T$$

  requires CG quantities from only ℓ future iterations

- Dispensing with explicit priors:
  Don't compute corresponding approximate prior of rank $k+\ell$

$$\widehat{\Gamma}_0 = \boldsymbol{V}_{1:k+\ell}\,\Phi_{1:k+\ell}\,\boldsymbol{V}_{1:k+\ell}^T$$

BayesCG under approximate Krylov posteriors

```
Input: spd matrix $\boldsymbol{A} \in \mathbb{R}^{d\times d}$, $\boldsymbol{b} \in \mathbb{R}^d$, posterior rank $\ell \geq 1$
    Run CG until convergence
            Compute approximate solution $\boldsymbol{x}_m$
    Run $\ell$ additional CG iterations (delay, look ahead)
            Compute search directions $\boldsymbol{v}_{m+1}, \ldots, \boldsymbol{v}_{m+\ell}$
            and residuals $\boldsymbol{r}_{m+1}, \ldots, \boldsymbol{r}_{m+\ell}$
    Construct approximate Krylov posterior covariance $\widehat{\Gamma}_m$
            in factored form $(\boldsymbol{V}_{m+1:m+\ell}, \Phi_{m+1:m+\ell})$
Output: Approximate Krylov posterior $\mathcal{N}(\boldsymbol{x}_m, \widehat{\Gamma}_m)$
```

Same computations as in CG error estimation with delay

[Golub, Meurant 1994,1995], [Golub, Strakoš 1994], [Meurant 1998]
[Meurant, Tichý, 2013, 2019], [Strakoš, Tichý 2002, 2005]

However, we use them to compute probability distributions

Accuracy of approximate Krylov posteriors

$\boldsymbol{A}$-norm Wasserstein distance between Gaussians μ and ν

$$W_{\boldsymbol{A}}(\mu,\nu) = \left(\inf_{\pi \in \Pi(\mu,\nu)} \int_{\mathbb{R}^{d\times d}\times\mathbb{R}^{d\times d}} \|M-N\|_{\boldsymbol{A}}^2 \, d\pi(M,N)\right)^{1/2}$$

Exact Krylov posterior covariance $\mu_k \equiv \mathcal{N}(\boldsymbol{x}_k, \Gamma_k)$

$$\Gamma_k = \boldsymbol{V}_{k+1:n}\, \Phi_{k+1:n}\, \boldsymbol{V}_{k+1:n}^T$$

Approximate rank-ℓ posterior covariance $\widehat{\mu}_k \equiv \mathcal{N}(\boldsymbol{x}_k, \widehat{\Gamma}_k)$

$$\widehat{\Gamma}_k = \boldsymbol{V}_{k+1:k+\ell}\, \Phi_{k+1:k+\ell}\, \boldsymbol{V}_{k+1:k+\ell}^T$$

Distance between rank-ℓ and exact distributions

$$W_{\boldsymbol{A}}(\mu_k, \widehat{\mu}_k) = (\mathrm{trace}(\Phi_{k+1:n}) - \mathrm{trace}(\Phi_{k+1:k+\ell}))^{1/2}$$

Amount by which $\mathrm{trace}(\boldsymbol{A}\,\widehat{\Gamma}_k)$ underestimates true error

Numerical Experiments: BayesCG under approximate Krylov prior

- Matrix $\boldsymbol{A} \in \mathbb{R}^{d \times d}$ with $d = 11{,}948$ and $\kappa(\boldsymbol{A}) \approx 2 \cdot 10^6$
- Iterations $1 \leq m \leq 2{,}500$ of BayesCG under Krylov prior
- Krylov posteriors $\widehat{\Gamma}_m$ of ranks $\ell = 1$ and $\ell = 50$
- Plots show

True error from posterior means: $\|\boldsymbol{x}_* - \boldsymbol{x}_m\|_{\boldsymbol{A}}^2$
Pointwise error from posterior covariances:

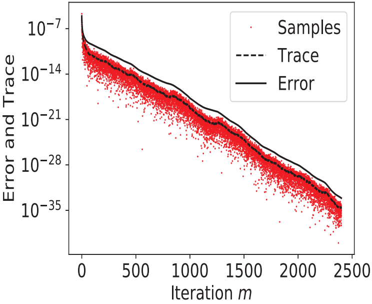
$$\mathrm{trace}(\boldsymbol{A}\,\widehat{\Gamma}_m) = \mathbb{E}[\|X_m - \boldsymbol{x}_m\|_{\boldsymbol{A}}^2]$$

Errors of samples from posterior

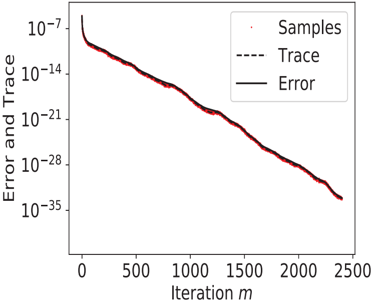
$$\|X_m - \boldsymbol{x}_m\|_{\boldsymbol{A}}^2 \text{ where } X_m \sim \mathcal{N}(\boldsymbol{x}_m, \widehat{\Gamma}_m)$$

‘Normalize’ $\boldsymbol{L}_m = \boldsymbol{V}_{m+1:m+\ell}\,\Phi_{m+1:m+\ell}^{1/2}$
Compute samples with matvecs: $X_m = \boldsymbol{x}_m + \boldsymbol{L}_m\,\mathtt{randn}(\ell, 1)$

BayesCG under approximate Krylov posterior



Rank $\ell = 1$



Rank $\ell = 50$

Pointwise error estimates $\mathrm{trace}(\boldsymbol{A}\widehat{\Gamma}_m) < \|\boldsymbol{x}_* - \boldsymbol{x}_m\|_{\boldsymbol{A}}^2$

Errors from rank-50 posterior samples $\|X_m - \boldsymbol{x}_m\|_{\boldsymbol{A}}^2 \approx \|\boldsymbol{x}_* - \boldsymbol{x}_m\|_{\boldsymbol{A}}^2$

Posterior rank $\ell \leq 0.5\%$ of matrix dimension

Summary: BayesCG under approximate Krylov posterior

Solution of $d \times d$ spd system $\boldsymbol{A}\boldsymbol{x}_* = \boldsymbol{b}$

- Choose posterior rank $\ell \lesssim 0.005 \cdot d$
- ℓ additional CG iterations after convergence

  Computation of rank-ℓ posterior $\widehat{\Gamma}_m$ same work as CG with delay

  Computation of posterior samples: matvecs with $d \times \ell$ matrix
- Pointwise estimates $\mathrm{trace}(\boldsymbol{A}\widehat{\Gamma}_m)$ underestimate true error

  Underestimate = distance between $\mathcal{N}(\boldsymbol{x}_m, \widehat{\Gamma}_m)$ and $\mathcal{N}(\boldsymbol{x}_m, \Gamma_m)$
- Errors $\|X_m - \boldsymbol{x}_m\|_{\boldsymbol{A}}^2$ from posterior samples $X_m \sim \mathcal{N}(\boldsymbol{x}_m, \widehat{\Gamma}_m)$ represent accurate surrogates for true error $\|\boldsymbol{x}_* - \boldsymbol{x}_m\|_{\boldsymbol{A}}^2$
- Can use $\mathcal{N}(\boldsymbol{x}_m, \widehat{\Gamma}_m)$ to generate and propagate uncertainty in approximation $\boldsymbol{x}_m$ due to early termination of CG

6. Application:
UQ in PDE-constrained optimization

Hyper-differential sensitivity analysis of uncertain parameters in PDE-constrained optimization[4]

Compute Singular Value Decomposition (SVD) of

$$\boldsymbol{M} = \boldsymbol{M}_1 \boldsymbol{A}^{-1} \boldsymbol{M}_2$$

$\boldsymbol{A}$ is spd, and accessible only through matvecs: $\boldsymbol{A} \cdot \text{vector}$
Each matvec requires several PDE solves

```
Extremely simplified algorithm (1 loop of subspace iteration)
Solve (1):  $\boldsymbol{Y} := \boldsymbol{M}\Omega$
        CG solves with $\boldsymbol{A}$ produce posteriors $\Gamma^{(1)}$
Solve (2):  $\boldsymbol{X} := \boldsymbol{M}^T \boldsymbol{Y}$
        CG solves with $\boldsymbol{A}$ produce posteriors $\Gamma^{(2)}$
Compute singular values of $\boldsymbol{X}$  (downstream computation)
```

[4] [Hart, Van Bloemen Waanders, Herzog 2020], [Saibaba, Hart, van Bloemen Waanders 2020]

Algorithm that propagates uncertainty due to early termination of CG

```
Input: Random matrix $\Omega$ with $n$ columns, $N$ samples
  for $k_1 = 1 : N$ do
    for $j = 1 : n$ do
      {Solve system (1) for column $j$ of sample $\boldsymbol{Y}_{k_1}$}
      $\boldsymbol{Y}^{(j)} := \boldsymbol{M}\Omega^{(j)}$, sample $Y^{(j)}$ from posterior $\Gamma_1^{(j)}$
    end for
    Set $\boldsymbol{Y}_{k_1} := \begin{bmatrix} Y^{(1)} & \dots & Y^{(n)} \end{bmatrix}$
    for $k_2 = 1 : N$ do
      for $i = 1 : n$ do
        {Solve system (2) for column $i$ of sample $\boldsymbol{X}_{k_1,k_2}$}
        $\boldsymbol{X}^{(i)} := \boldsymbol{M}^T \boldsymbol{Y}_{k_1}^{(i)}$, sample $X^{(i)}$ from posterior $\Gamma_2^{(i)}$
      end for
      Compute singular values of $\boldsymbol{X}_{k_1,k_2} := \begin{bmatrix} X^{(1)} & \dots & X^{(n)} \end{bmatrix}$
    end for
  end for
```

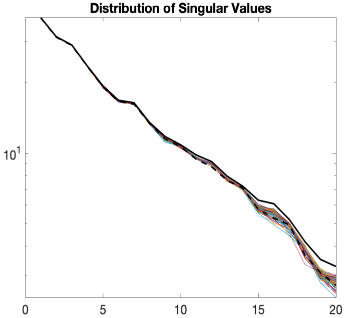
Which solve is more important for accurate singular values?

Piping posteriors from 1. linear solve into 2. solve
Probe uncertainty in downstream singular values

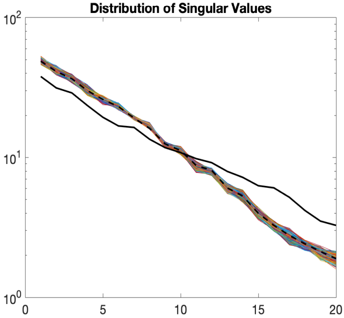
- $\boldsymbol{A}$ has dimension $d = 225$
- Posteriors have rank $\ell = 10 = .05\, d$
- Ω, $\boldsymbol{X}$, $\boldsymbol{Y}$ have 40 columns,
- Number of samples for each solve $N = 100$
- Accurate solve has stopping tolerance $\eta = 10^{-6}$
  Other solve performs 1 iteration
- Compute 20 largest singular values for each of the $N^2 = 10^4$ samples $\boldsymbol{X}_{k_1,k_2}$

20 largest singular values from each of the 10^4 samples

Black line: ‘exact’ singular values



Accurate solve (2)



Accurate solve (1)

??????

5. Take-Home Message

- BayesCG: Probabilistic extension of Conjugate Gradient (CG)

  For solution of symmetric positive definite linear systems

- Efficient low-rank Krylov posteriors for BayesCG

  No explicit prior, implicit prior customized to linear system
  Cheap to compute (a few more iterations & matvecs)
  Numerically accurate (maintained in factored form)
  Deliver accurate surrogates of the error
  Generate and propagate uncertainty due to early termination of CG

Issues

- Empirical uncertainty propagation requires combinatorial number of samples
- A potential alternative: Analytical uncertainty propagation by adding Gaussian noise to posteriors????
- Numerically stable implementations non-trivial
- Rigorous & meaningful statistical setting