Probabilistic Numerical Linear Solvers

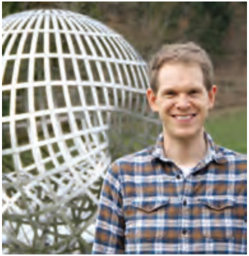
Ilse C.F. Ipsen

NC STATE UNIVERSITY

Raleigh, NC, USA

Research supported in part by NSF DMS: RTG & FRG

Collaborators



Jon Cockayne

Alan Turing Institute, UK

Chris Oates

University of Newcastle upon Tyne, UK

Tim Reid

North Carolina State University, USA

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, termination of iterative methods)

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, termination of iterative methods)

But why??? Want error measures that

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

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, termination of iterative methods)

But why??? Want error measures that

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

Probabilistic numerical methods have been developed for:
Approximation, quadrature, numerical solution of ordinary and partial differential equations, optimization

Here: Computational kernels and linear algebra

Our Focus: Accurate Error Estimation for Iterative Solution of Linear Systems

[Disclaimer: Research still at proof-of-concept stage]

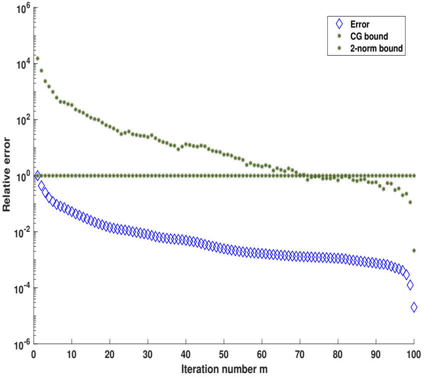
Given: Nonsingular real matrix $\boldsymbol{A}$, vector $\boldsymbol{b}$
Want: Solution $\boldsymbol{x}_*$ of linear system $\boldsymbol{A}\boldsymbol{x}_* = \boldsymbol{b}$

Iterative solver:

From user-specified initial guess $\boldsymbol{x}_0$
computes iterates $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m \rightarrow \boldsymbol{x}_*$

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

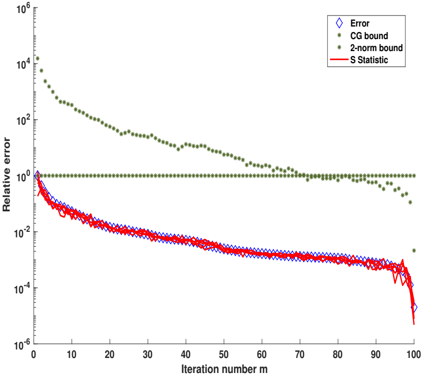
What We Can Do So Far (Details Later)



Exact error $\|\boldsymbol{x}_m - \boldsymbol{x}_*\|/\|\boldsymbol{x}_*\|$

Traditional error bounds ≥ 1, not informative

What We Can Do So Far (Details Later)



Exact error $\|\boldsymbol{x}_m - \boldsymbol{x}_*\| / \|\boldsymbol{x}_*\|$

Traditional error bounds ≥ 1, not informative

Our S statistic from posterior distribution $\approx$ exact error

Overview

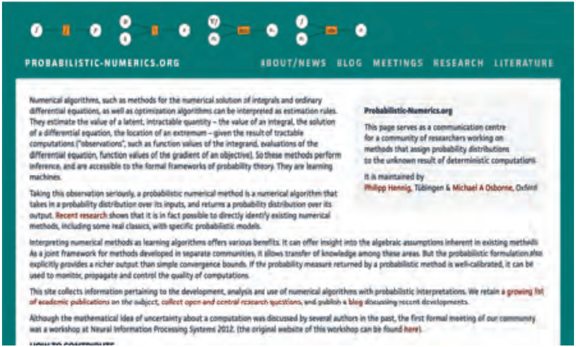
1. Probabilistic Numerics: The Big Picture
2. Gaussian Probability Distributions
3. BayesCG, a Probabilistic Numerical Linear Solver
4. Errors, and Metrics on Probability Distributions
5. Two Priors that Reproduce Conjugate Gradient

    Inverse prior

    Krylov priors

6. Low-Rank Approximations of Krylov Priors

1. Probabilistic Numerics: The Big Picture



Numerical algorithms, such as methods for the numerical solution of integrals and ordinary differential equations, as well as optimization algorithms can be interpreted as estimation rules. They estimate the value of a latent, intractable quantity – the value of an integral, the solution of a differential equation, the location of an extremum – given the result of tractable computations ("observations", such as function values of the integrand, evaluations of the differential equation, function values of the gradient of an objective). So these methods perform inference, and are accessible to the formal frameworks of probability theory. They are learning machines.

Taking this observation seriously, a probabilistic numerical method is a numerical algorithm that takes in a probability distribution over its inputs, and returns a probability distribution over its output. Recent research shows that it is in fact possible to directly identify existing numerical methods, including some real classics, with specific probabilistic models.

Probabilistic-Numerics.org

This page serves as a communication centre for a community of researchers working on methods that assign probability distributions to the unknown result of deterministic computations.

It is maintained by
Philipp Hennig, Tübingen & Michael A Osborne, Oxford

Interpreting numerical methods as learning algorithms offers various benefits. It can offer insight into the algebraic assumptions inherent in existing methods. As a joint framework for methods developed in separate communities, it allows transfer of knowledge among these areas. But the probabilistic formulation also explicitly provides a richer output than simple convergence bounds. If the probability measure returned by a probabilistic method is well-calibrated, it can be used to monitor, propagate and control the quality of computations.

This site collects information pertaining to the development, analysis and use of numerical algorithms with probabilistic interpretations. We retain a growing list of academic publications on the subject, collect open and central research questions, and publish a blog discussing recent developments.

Although the mathematical idea of uncertainty about a computation was discussed by several authors in the past, the first formal meeting of our community was a workshop at Neural Information Processing Systems 2012. (the original website of this workshop can be found here).

HOW TO CONTRIBUTE

`http://probabilistic-numerics.org/`

Early Probabilistic Numerics

- H. Poincaré (1896, 1912): Calcul des Probabilités

- A.V. Sul'din (1959): Wiener Measure and its Application to Approximation Methods, *I. Izv. Vysš. Učben. Zayed. Mat.*
- F. M. Larkin (1972): Gaussian Measure in Hilbert space and Applications in Numerical Analysis, *Rocky Mountain J. Math.*

- G.S. Kimeldorf, G. Wahba (1970): A Correspondence between Bayesian Estimation on Stochastic Processes and Smoothing by Splines, *Ann. Math. Stat.*
- P. Diaconis (1988): Bayesian Numerical Analysis, *Stat. Decis. Theory Relat. Top. IV*
- A. O'Hagan (1992): Some Bayesian Numerical Analysis, *Bayesian. Stat.*

Perspectives on Modern Probabilistic Numerics

- Bayesian inference in inverse problems
  A.M. Stuart (2010): Inverse Problems: A Bayesian Perspective, *Acta Numer.*

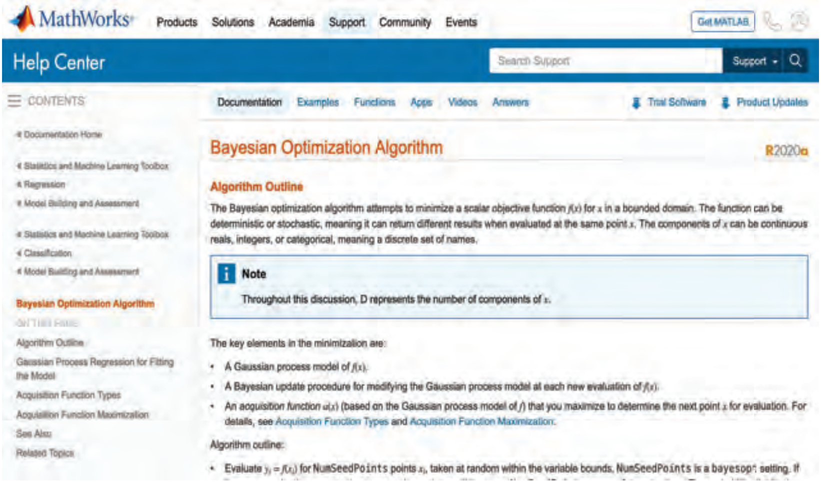
- ‘A call to arms for probabilistic numerical methods’
  P. Hennig, M.A Osborne, M. Girolami (2015): Probabilistic Numerics and Uncertainty in Computations, *Proc. R. Soc. A*

- Statistical foundations for non-linear, non-Gaussian problems
  J. Cockayne, C.J. Oates, T.J. Sullivan, M. Girolami (2019): Bayesian Probabilistic Numerical Methods, *SIAM Rev.*

- Historical development of probabilistic numerics
  C.J. Oates, T.J. Sullivan (2019): A Modern Retrospective on Probabilistic Numerics, *Stat. Comput.*

- Probabilistic Numerics at SIAM Conference on UQ
  `http://probabilistic-numerics.org/meetings/SIAMUQ2020/`

Application: Global Optimization



Bayesian Optimization Algorithm

R2020a

Algorithm Outline

The Bayesian optimization algorithm attempts to minimize a scalar objective function $f(x)$ for x in a bounded domain. The function can be deterministic or stochastic, meaning it can return different results when evaluated at the same point x. The components of x can be continuous reals, integers, or categorical, meaning a discrete set of names.

Note

Throughout this discussion, D represents the number of components of x.

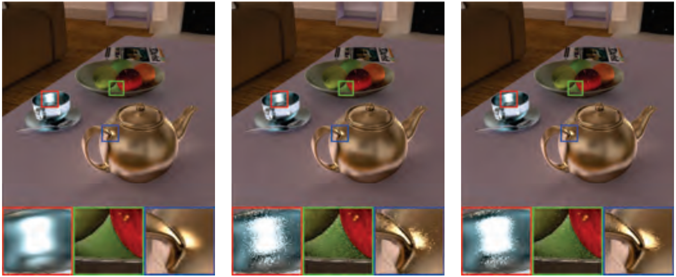
The key elements in the minimization are:

- A Gaussian process model of $f(x)$.
- A Bayesian update procedure for modifying the Gaussian process model at each new evaluation of $f(x)$.
- An *acquisition function* $a(x)$ (based on the Gaussian process model of f) that you maximize to determine the next point x for evaluation. For details, see Acquisition Function Types and Acquisition Function Maximization.

Algorithm outline:

- Evaluate $y_i = f(x_i)$ for `NumSeedPoints` points x_i, taken at random within the variable bounds. `NumSeedPoints` is a `bayesopt` setting. If

Application: Numerical Integration for Rendering of Glossy Surfaces



(a) Reference

(RMSE = 0.039, time = 2m18s)
(b) LDIS

(RMSE = 0.026, time = 2m20s)
(c) BMC

Fig. 1. Indirect radiance component for the Room scene rendered with low discrepancy Monte Carlo Importance Sampling (LDIS, (b)) and BMC (c). The shown images have been multiplied by a factor of 4. The RMSE and the time are computed by considering only the glossy component. 16 ray samples per visible point were used for the materials with a glossy BRDF (teapot, tea cup and fruit-dish), while 64 samples were used for the diffuse BRDFs.

R. Marques, C. Bouville, M. Ribardiere, L.P. Santos, K. Bouatouch (2013), *IEEE Trans. Vis. Comput. Graph.*

Probabilistic Numerics & Other Areas

Connections to

- Information-based complexity
  [Traub, Wasilkowski, Woźniakowski 1983]
- Average-case analysis
  [Novak 1988], [Ritter 2000]
- Data assimilation, Kalman filters
  [Law, Stuart, Zygalakis 2015], [Reich, Cotter 2015]
- Statistical learning
  [Hastie, Tibshirani, Friedman 2009], [Rasmussen, Williams 2006]

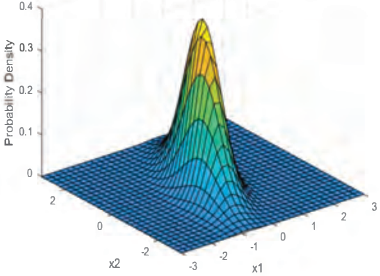
Difference to Uncertainty Quantification [Smith 2014]

- UQ: Problems usually ill-posed, uncertainties from all sources: model, parameter, experimental, measurement
- PN: Problems are well-posed, algorithmic uncertainty due to limited computational resources

Probabilistic Numerics in Linear Algebra

- 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 (2017): Compression, Inversion, and Approximate PCA of Dense Kernel Matrices at Near-Linear Computational Complexity, *arXiv*
- 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.*

2. Gaussian Probability Distributions



[The MathWorks R2020a]

Our Probability Distributions are Gaussians

Gaussian = multi-variate normal distribution
Gaussian random vector $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0)$

Mean $\boldsymbol{\mu}_0 \in \mathbb{R}^d$
Covariance $\Sigma_0 \in \mathbb{R}^{d \times d}$ symmetric positive semi-definite

If Σ_0 also positive definite, then probability density function is

$$p(\boldsymbol{x}) = \frac{1}{\sqrt{\det(\Sigma_0)\,(2\pi)^d}}\ \exp\left(-\frac{1}{2}\|\boldsymbol{x} - \boldsymbol{\mu}_0\|^2_{\Sigma_0^{-1}}\right)$$

where $\|\boldsymbol{x} - \boldsymbol{\mu}_0\|^2_{\Sigma_0^{-1}} = (\boldsymbol{x} - \boldsymbol{\mu}_0)^T \Sigma_0^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_0)$ and T is transpose

Why Gaussians?
Stability: Linear transformations preserve Gaussianity
If $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0)$ then $\boldsymbol{B}\,\boldsymbol{x} + \boldsymbol{z} \sim \mathcal{N}(\boldsymbol{B}\,\boldsymbol{\mu}_0 + \boldsymbol{z},\ \boldsymbol{B}\,\Sigma_0\,\boldsymbol{B}^T)$

[Muirhead 1982], [Stuart 2010]

Stability of Gaussians for Computing Posteriors

If $\boldsymbol{x} \sim \mathcal{N}(\mu_0, \Sigma_0)$ then $\boldsymbol{z} + \boldsymbol{B}\,\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{z} + \boldsymbol{B}\,\mu_0, \boldsymbol{B}\,\Sigma_0\,\boldsymbol{B}^T)$

- Prior: $\boldsymbol{x} \sim \mathcal{N}(\mu_0, \Sigma_0)$
- Condition on linear information $\boldsymbol{y} = \boldsymbol{B}\boldsymbol{x}$
- Bayesian inference

  Posterior: $\boldsymbol{x} \,|\, \boldsymbol{y} \sim \mathcal{N}(\mu, \Sigma)$ with

$$\begin{aligned} \mu &= \mu_0 + \Sigma_0 \boldsymbol{B}^T (\boldsymbol{B}\,\Sigma_0\,\boldsymbol{B}^T)^{-1} (\boldsymbol{y} - \boldsymbol{B}\mu_0) \\ \Sigma &= \Sigma_0 - \Sigma_0 \boldsymbol{B}^T (\boldsymbol{B}\,\Sigma_0\,\boldsymbol{B}^T)^{-1}\,\boldsymbol{B}\,\Sigma_0 \end{aligned}$$

Connections to: Schur complements, projectors

Conditioning Gaussian random vector $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{x}_0, \Sigma_0)$
on linear information $\boldsymbol{y} = \boldsymbol{B}\boldsymbol{x}$
gives another Gaussian random vector $\boldsymbol{x} \,|\, \boldsymbol{y} \sim \mathcal{N}(\mu, \Sigma)$

Stability of Gaussians for Computational Sampling

- Want: Sample $X \sim \mathcal{N}(\mu, \Sigma)$
- Stability: If $\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I})$ then $\mu + \boldsymbol{L}\boldsymbol{x} \sim \mathcal{N}(\mu, \boldsymbol{L}\boldsymbol{L}^T)$
- Possible factorizations $\Sigma = \boldsymbol{L}\boldsymbol{L}^T$

  Square root $\boldsymbol{L} = \Sigma^{1/2}$
  Cholesky factorization, if Σ nonsingular
  Thin Cholesky: $\boldsymbol{L}$ has $\mathrm{rank}(\Sigma)$ columns

- Matlab

$$X = \mu + \boldsymbol{L} * \underbrace{\mathtt{randn}(\mathtt{size}(\boldsymbol{L}, 2), 1)}_{\text{vector } \sim \mathcal{N}(0, \boldsymbol{I})}$$

3. BayesCG, a Probabilistic Numerical Linear Solver

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 iterates $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m \to \boldsymbol{x}_*$

Solution-based inference

Probability distribution over solution space $\boldsymbol{x} \in \mathbb{R}^d$
Keeping it simple: $\boldsymbol{x}_m = \boldsymbol{\mu}_m$, $m \geq 0$
Iterates coincide with 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 $\mathcal{N}(\boldsymbol{x}_m, \Sigma_m)$
  Posteriors reflect knowledge about $\boldsymbol{x}_*$ after m iterations

Computing the Posteriors in Iteration $m \geq 1$

[Cockayne, Oates, Sullivan, Girolami 2019]

Information about $\boldsymbol{x}_*$ provided by m search directions

$$\boldsymbol{S}_m = \begin{pmatrix} \boldsymbol{s}_1 & \cdots & \boldsymbol{s}_m \end{pmatrix} \in \mathbb{R}^{d \times m} \qquad \mathrm{rank}(\boldsymbol{S}_m) = m$$

Condition on $\boldsymbol{y}_m = \boldsymbol{S}_m^T \boldsymbol{A} \boldsymbol{x}_* = \boldsymbol{S}_m^T \boldsymbol{b}$

Exists unique Bayesian method that outputs posterior distribution

$$\boldsymbol{x} \mid \boldsymbol{y}_m \sim \mathcal{N}(\boldsymbol{x}_m, \Sigma_m)$$

with

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

Choose $\boldsymbol{A}\Sigma_0\boldsymbol{A}$-orthogonal search directions: $\boldsymbol{S}_m^T \boldsymbol{A} \Sigma_0 \boldsymbol{A} \boldsymbol{S}_m = \boldsymbol{I}_m$

Bayesian Conjugate Gradient Method (BayesCG)

[Cockayne, Oates, Ipsen, Girolami 2019]

```
$\boldsymbol{r}_0 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0$    {initial residual}
$\boldsymbol{s}_1 = \boldsymbol{r}_0 / \sqrt{\boldsymbol{r}_0{}^T \boldsymbol{A}\,\Sigma_0 \boldsymbol{A}\,\boldsymbol{r}_0}$    {initial search direction}
$m = 1$
while not converged do
    $\Sigma_m = \Sigma_{m-1} - (\Sigma_0 \boldsymbol{A}\boldsymbol{s}_m)\,(\Sigma_0 \boldsymbol{A}\boldsymbol{s}_m)^T$    {next posterior}
    $\boldsymbol{x}_m = \boldsymbol{x}_{m-1} + \Sigma_0 \boldsymbol{A}\,\boldsymbol{s}_m\,(\boldsymbol{r}_{m-1}^T \boldsymbol{s}_m)$    {next iterate}
    $\boldsymbol{r}_m = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_m$    {next residual}
    $\tilde{\boldsymbol{s}}_{m+1} = \boldsymbol{r}_m - \boldsymbol{s}_m\,(\boldsymbol{r}_m^T \boldsymbol{A}\,\Sigma_0 \boldsymbol{A}\,\boldsymbol{s}_m)$    {next search direction}
    $\boldsymbol{s}_{m+1} = \tilde{\boldsymbol{s}}_{m+1} / \sqrt{\tilde{\boldsymbol{s}}_{m+1}^T \boldsymbol{A}\,\Sigma_0 \boldsymbol{A}\,\tilde{\boldsymbol{s}}_{m+1}}$    {normalize search dir}
end while
```

Properties of BayesCG

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

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

- Error minimization in Σ_0^{-1} norm

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

- Convergence in Σ_0^{-1} norm depends on conditioning of $\Sigma_0\, \boldsymbol{A}^2$

$$\|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\Sigma_0^{-1}} \leq 2 \left(\frac{\sqrt{\kappa_2(\Sigma_0\, \boldsymbol{A}^2)} - 1}{\sqrt{\kappa_2(\Sigma_0\, \boldsymbol{A}^2)} + 1}\right)^m \|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_{\Sigma_0^{-1}}$$

- Prior: If $\Sigma_0 = \boldsymbol{A}^{-1}$ then BayesCG = CG
  CG = Bayesian inference with prior $\mathcal{N}(\boldsymbol{x}_0, \boldsymbol{A}^{-1})$

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 $\boldsymbol{x}_0$ identical to initial guess for iterates
- Output: Gaussian posterior $\mathcal{N}(\boldsymbol{x}_m, \Sigma_m)$ at iteration m
  Models uncertainty about solution $\boldsymbol{x}_*$ after m iterations
  Mean identical to iterate $\boldsymbol{x}_m \approx \boldsymbol{x}_*$

Next: How to use the BayesCG posteriors?

4. Errors, and
Metrics on Probability Distributions

Metric on Set of Probability Distributions

Measure distance between two probability distributions

- Any two Gaussian distributions
  $\mathcal{G}_m = \mathcal{N}(\boldsymbol{x}_m, \Sigma_m)$ and $\mathcal{G}_* = \mathcal{N}(\boldsymbol{x}_*, \Sigma_*)$
- Wasserstein 2-norm metric: Distance between $\mathcal{G}_m$ and $\mathcal{G}_*$

$$[W_2(\mathcal{G}_m,\, \mathcal{G}_*)]^2 = \|\boldsymbol{x}_m - \boldsymbol{x}_*\|_2^2 + \mathrm{trace}\left(\Sigma_m + \Sigma_* - 2\,(\Sigma_m \Sigma_*)^{1/2}\right)$$

[Dowson, Landau 1982]

- Extension to general weighted norms ($\boldsymbol{B}$ is spd)

$$\begin{aligned}[W_{\boldsymbol{B}}(\mathcal{G}_m,\, \mathcal{G}_*)]^2 = {} & \|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{B}}^2 \\ & + \mathrm{trace}\left(\boldsymbol{B}(\Sigma_m + \Sigma_*) - 2\,(\boldsymbol{B}^{1/2}\Sigma_m \boldsymbol{B} \Sigma_* \boldsymbol{B}^{1/2})^{1/2}\right)\end{aligned}$$

Application of Wasserstein Metric to BayesCG

Input: Prior $\mathcal{G}_0 = \mathcal{N}(\boldsymbol{x}_0, \Sigma_0)$
Output: Posteriors $\mathcal{G}_m = \mathcal{N}(\boldsymbol{x}_m, \Sigma_m)$
Errors: $\|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\Sigma_0^{-1}}$

Represent solution $\boldsymbol{x}_*$ as point distribution $\mathcal{N}(\boldsymbol{x}_*, 0)$

$$\begin{aligned}\left[W_{\Sigma_0^{-1}}(\mathcal{G}_m,\, \boldsymbol{x}_*)\right]^2 &= \|\boldsymbol{x}_m - \boldsymbol{x}_*\|^2_{\Sigma_0^{-1}} + \mathrm{trace}\left(\Sigma_0^{-1}\,\Sigma_m\right) \\ &= \|\boldsymbol{x}_m - \boldsymbol{x}_*\|^2_{\Sigma_0^{-1}} + (d - m)\end{aligned}$$

Σ_0^{-1} distance of posterior $\mathcal{G}_m$ to solution $\boldsymbol{x}_*$ equals
Σ_0^{-1} norm error $+$ dimension of unexplored space

Computational error estimation with “S-statistic”

$$\|\boldsymbol{x}_m - \boldsymbol{x}_*\|^2_{\Sigma_0^{-1}} \approx \|\boldsymbol{x}_m - X\|^2_{\Sigma_0^{-1}} \qquad \text{where} \quad X \sim \mathcal{G}_m$$

Possible Prior Distributions

- "Noninformative": $\Sigma_0 = \boldsymbol{I}_d$
- Inverse: $\Sigma_0 = \boldsymbol{A}^{-1}$, BayesCG = CG
- Natural: $\Sigma_0 = \boldsymbol{A}^{-2}$, convergence in 1 iteration
- Preconditioner: $\Sigma_0 = (\boldsymbol{P}^T\boldsymbol{P})^{-1} \approx \boldsymbol{A}^{-2}$
- Krylov[1]: $\Sigma_0 = \boldsymbol{V}\Phi\boldsymbol{V}^T$, 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[1]

Impractical "academic" priors that already contain all information to compute solution immediately

But: We study them to calibrate our expectations

In order to: Develop practical, low-rank approximations

[1] Different from Krylov prior in [Cockayne, Oates, Ipsen, Girolami 2019]

5a. Priors that Reproduce CG
Inverse Prior

Wasserstein Metric for BayesCG under Inverse Prior

- Prior $\mathcal{G}_0 = \mathcal{N}(\boldsymbol{x}_0, \Sigma_0)$ with $\Sigma_0 = \boldsymbol{A}^{-1}$
- BayesCG under inverse prior = CG + posteriors
- Posteriors $\mathcal{G}_m = \mathcal{N}(\boldsymbol{x}_m, \Sigma_m)$ with $\Sigma_m = \Sigma_{m-1} - \boldsymbol{s}_m \boldsymbol{s}_m^T$
  Down dating with search directions

Distance between posterior and solution

$$[W_{\boldsymbol{A}}(\mathcal{G}_m,\, \boldsymbol{x}_*)]^2 = \|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2 + (d - m)$$

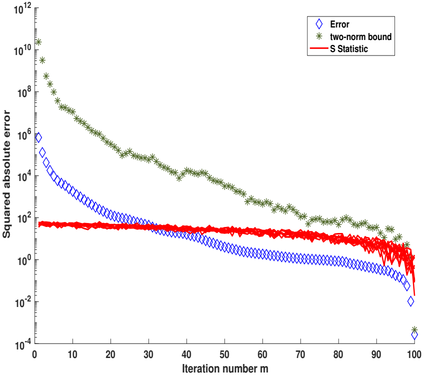
Error estimation with S statistic

$$\|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2 \approx \|\boldsymbol{x}_m - X\|_{\boldsymbol{A}}^2 \qquad \text{where} \quad X \sim \mathcal{G}_m$$

Experiments: $\boldsymbol{A}\boldsymbol{x}_* = \boldsymbol{b}$

Dimension $d = 100$, $\boldsymbol{x}_* = \mathbb{1}$, $\kappa_2(\boldsymbol{A}) \approx 4 \cdot 10^4$
S statistic samples per iteration: 10

BayesCG under Inverse Prior
Absolute Error, Two-norm Bound, S Statistic



Squared absolute error $\|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2$

S statistic samples

Two-norm bound squared

BayesCG under Inverse Prior
Preliminary Conclusions

Pros

- Iterates and convergence identical to that of CG
- Samples from S statistic more accurate than norm-wise forward error bounds

Cons

- Preserving semi-definiteness of posterior during down dates $\Sigma_m = \Sigma_{m-1} - \boldsymbol{s}_m \boldsymbol{s}_m^T$
- Prior is not practical

Existing Work on $\boldsymbol{A}$-Norm Errors in CG

Estimation approaches:

Ritz values, Gauss-Radau rules, incremental norm estimation, maximal attainable accuracy estimates

Golub, Strakoš 1994
Greenbaum 1997
Meurant 1998
Golub, Meurant 1997
Strakoš, Tichý 2002
Strakoš, Tichý 2005
Meurant, Tichý 2013
Liesen, Strakoš 2016
Meurant, Tichý 2019

Probabilistic numerical solvers:

Designed to capture computational uncertainty

5b. Priors that Reproduce CG
Krylov Priors

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

Columns of $\boldsymbol{V}$ are $\boldsymbol{A}$-orthonormal basis

$$\{\boldsymbol{V}^T\boldsymbol{A}\boldsymbol{V} = \boldsymbol{I}_n, \text{ normalized CG search directions}\}$$

for Krylov space of maximal dimension $n \leq d$

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

Φ is any diagonal matrix Φ with positive diagonal

Advantages

- BayesCG under Krylov prior = CG + posteriors
- Trailing submatrices: No explicit downdating of posteriors

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

Γ_0 singular for $n < d$

$\Longrightarrow$ cannot use Γ_0 to define norm for Wasserstein distance

Wasserstein Metric for BayesCG under Specific Krylov Prior

Specific Krylov prior: $\Gamma_0 = \boldsymbol{V}\Phi\boldsymbol{V}^T$ has diagonal elements

$$\Phi_{mm} = (\boldsymbol{v}_m^T\boldsymbol{r}_0)^2 \qquad 1 \leq m \leq n$$

{Computed automatically in our CG implementation}

- $\boldsymbol{A}$-norm error with specific Krylov prior

$$\|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2 = \mathrm{trace}\,(\boldsymbol{A}\Gamma_m)$$

- Wasserstein $\boldsymbol{A}$-norm metric for BayesCG under Γ_0

$$[W_{\boldsymbol{A}}(\mathcal{G}_m,\, \boldsymbol{x}_*)]^2 = \|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2 + \mathrm{trace}\,(\boldsymbol{A}\,\Gamma_m)$$

Wasserstein $\boldsymbol{A}$-norm metric $\triangleq$ $\boldsymbol{A}$-norm error in iterate

$$[W_{\boldsymbol{A}}(\mathcal{G}_m,\, \boldsymbol{x}_*)]^2 = 2\,\|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2$$

S Statistic for Specific Krylov Prior

Krylov posterior $\mathcal{G}_m = \mathcal{N}(\boldsymbol{x}_m, \Gamma_m)$

- Distance of posterior to solution $\triangleq$ absolute error

$$[W_{\boldsymbol{A}}(\mathcal{G}_m,\, \boldsymbol{x}_*)]^2 = 2\,\|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2$$

- Absolute error $\triangleq$ mean of S statistic

$$2\,\|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2 = \mathbb{E}_{X\sim\mathcal{G}_m}\left[\|\boldsymbol{x}_m - X\|_{\boldsymbol{A}}^2\right]$$

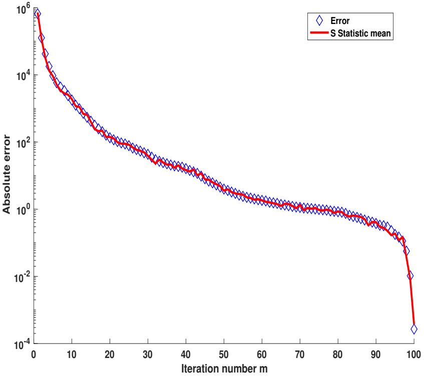
- Empirical mean of S statistic from 10 samples

$$\mathbb{E}_{X\sim\mathcal{G}_m}\left[\|\boldsymbol{x}_m - X\|_{\boldsymbol{A}}^2\right] \approx \tfrac{1}{10}\sum_{j=1}^{10} \tfrac{1}{2}\|\boldsymbol{x}_m - X_j\|_{\boldsymbol{A}}^2 \qquad X_j \sim \mathcal{G}_m$$

Experiments: $\boldsymbol{A}\boldsymbol{x}_* = \boldsymbol{b}$

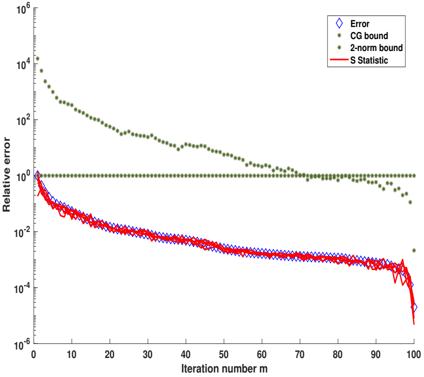
Dimension $d = 100$, $\boldsymbol{x}_* = \mathbb{1}$, same $\boldsymbol{A}$ as before
Condition number $\kappa_2(\boldsymbol{A}) \approx 4 \cdot 10^4$
S statistic samples per iteration: 10

BayesCG under Specific Krylov Prior
Empirical Mean of S Statistic $\approx$ Absolute Error



Squared absolute error $\|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2$
Empirical mean of S statistic from 10 samples

BayesCG under Specific Krylov Prior
Relative Error, Bounds, S Statistic



Exact error $\|\mathbf{x}_m - \mathbf{x}_*\|_{\mathbf{A}} / \|\mathbf{x}_*\|_{\mathbf{A}}$

S statistic samples (square-root, normalized) $\approx$ exact error

CG bound, and general 2-norm error bound

BayesCG under Specific Krylov Prior
Preliminary Conclusions

Pros:

- Errors in iterates $\triangleq$ distances of posteriors to solution
- Errors in iterates $\triangleq$ mean of S statistic
- S statistic produces exact magnitude of errors

Cons: BayesCG under Krylov priors requires

1. Running maximal number n of CG iterations
2. Storing all n search directions V (for factored form $\Gamma_0 = \boldsymbol{V}\Phi\boldsymbol{V}^T$)
3. Matvecs with $\Gamma_m \in \mathbb{R}^{d\times(n-m+1)}$ in iteration m

Again, this is not practical

Next: Low-rank approximations of Krylov priors are promising

6. Low-Rank Approximations of Krylov Priors

Low-Rank Approximation via Look-Ahead

- Run k iterations of CG
- Look-ahead: Run $\ell = 5$ more CG iterations
- Prior has rank $k + \ell = k + 5$
- Estimate errors $e_m = \|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2$ at iteration $1 \le m \le k$ with empirical mean of 5 samples from S statistic

$$\sigma_m = \tfrac{1}{5} \sum_{j=1}^{5} \tfrac{1}{2} \|\boldsymbol{x}_m - X_{mj}\|_{\boldsymbol{A}}^2 \qquad X_{mj} \sim \mathcal{N}(\boldsymbol{x}_m, \hat{\Gamma}_m)$$

Error of empirical mean in last iteration is $|\sigma_k - e_k|/e_k$

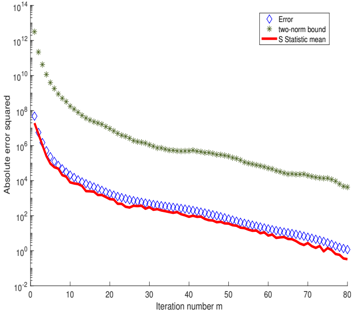
3 numerical examples: $\boldsymbol{A}\boldsymbol{x}_* = \boldsymbol{b}$ of dimension d

$\boldsymbol{A} = \boldsymbol{Q}\Lambda\boldsymbol{Q}^T$ with $\boldsymbol{Q}$ random orthogonal [Stewart 1980]

Eigenvalue distributions from [Liesen, Strakoš 2013]

Uniformly Distributed Eigenvalues, Dimension $d = 10^3$
$k = 80$ Iterations, Prior of Rank 85

$\kappa_2(\boldsymbol{A}) = 10^5, \qquad \text{Eigenvalues } \lambda_j = 1 + \frac{j-1}{d-1}\,(10^5 - 1), \;\; 1 \le j \le d$

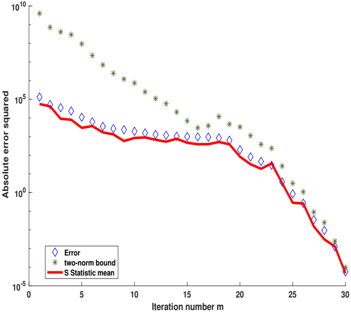


Squared absolute error $\|\boldsymbol{x}_m - \boldsymbol{x}_*\|^2_{\boldsymbol{A}}$

Empirical mean of S statistic, mean error in last iteration $\le .72$

Large Eigenvalue Cluster, Dimension $d = 10^3$
 $k = 30$ Iterations, Prior of Rank 35

$\kappa_2(\boldsymbol{A}) = 10^5$, Eigenvalues $\lambda_j = 1 + \frac{j-1}{d-1}\,(10^5 - 1)\,0.65^{d-j}$

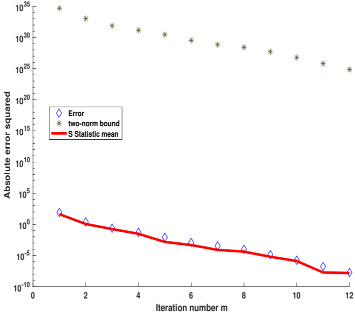


Squared absolute error $\|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2$

Empirical mean of S statistic, mean error in last iteration $\leq .18$

Smoothly Increasing Eigenvalues, Dimension $d = 100$
$k = 12$ Iterations, Prior of Rank 17

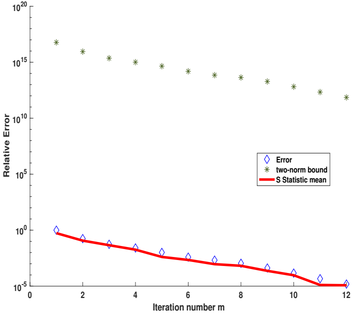
$\kappa_2(\boldsymbol{A}) \approx 7 \cdot 10^{16}$,   Eigenvalues $\lambda_j = \log(j)/log(d)$, $1 \leq j \leq d$



Squared absolute error $\|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2$

Empirical mean of S statistic, mean error in last iteration $\leq .2$

Relative Errors for Previous Problem



Relative error $\|\boldsymbol{x}_m - \boldsymbol{x}_*\|_{\boldsymbol{A}} / \|\boldsymbol{x}_*\|_{\boldsymbol{A}}$, empirical mean of S statistic, 2-norm bound

Last iteration

Two-norm bound: No accuracy $(\kappa_2(\boldsymbol{A}) \approx 7 \cdot 10^{16})$

Relative error: $\|\boldsymbol{x}_{12} - \boldsymbol{x}_*\|_{\boldsymbol{A}} / \|\boldsymbol{x}_*\|_{\boldsymbol{A}} \approx 1.5405 \cdot 10^{-5}$

S statistic empirical mean: $\sigma_{12} \approx 1.1956 \cdot 10^{-5}$

Accuracy of Low-Rank Prior Approximations

Experiments: Error estimates have correct magnitude
especially where traditional bounds are not informative

Exact prior $\mathcal{G}_0 = \mathcal{N}(\boldsymbol{x}_0, \Gamma_0)$

$$\Gamma_0 = \boldsymbol{V}\,\Phi\,\boldsymbol{V}^T \qquad \mathrm{rank}(\Gamma_0) = n$$

Rank $k+\ell$ approximation $\hat{\mathcal{G}}_0 = \mathcal{N}(\boldsymbol{x}_0, \hat{\Gamma}_0)$

$$\hat{\Gamma}_0 = \boldsymbol{V}_{1:k+\ell}\,\Phi_{1:k+\ell,1:k+\ell}\,\boldsymbol{V}_{1:k+\ell}^T \qquad \mathrm{rank}(\hat{\Gamma}_0) = k+\ell$$

Distance between low-rank and exact distributions

$$[W_{\boldsymbol{A}}(\mathcal{G}_m, \hat{\mathcal{G}}_m)]^2 = \|\boldsymbol{V}_{k+\ell+1:n}^T \boldsymbol{r}_0\|_2^2 \qquad 0 \le m \le k$$

Distance $\approx$ contribution of $\boldsymbol{r}_0$ in search directions ignored by $\hat{\Gamma}_0$

Practical Procedure: Error Estimate for Last Iterate

1. Run CG until convergence, at some iteration k
2. Run ℓ more CG iterations, store ℓ search directions $\boldsymbol{V}_{k+1:k+\ell}$
3. Normalize factor for low-rank posterior $\hat{\Gamma}_k = \boldsymbol{L}_k \boldsymbol{L}_k^T$

$$\boldsymbol{L}_k = \boldsymbol{V}_{k+1:k+\ell}\, \Phi_{k+1:k+\ell}^{1/2} \in \mathbb{R}^{d\times\ell}$$

4. S statistic samples $\sigma_j = \|\boldsymbol{x}_k - X_j\|_{\boldsymbol{A}}^2, \;\; 1 \leq j \leq n_s$

$$X_j = \boldsymbol{x}_k + \boldsymbol{L}_k\, \mathtt{randn}(\ell, 1)$$

5. Error estimate = empirical mean of S statistic samples

$$\frac{1}{n_s} \sum_{j=1}^{n_s} \frac{1}{2}\sigma_j \approx \|\boldsymbol{x}_k - \boldsymbol{x}_*\|_{\boldsymbol{A}}^2$$

Work in addition to CG: ℓ iterations, $c\, n_s$ matvecs

Additional storage: ℓ search directions

Take-Home Message

Solution of $\boldsymbol{A}\boldsymbol{x}_* = \boldsymbol{b}$ with real symmetric positive definite matrix $\boldsymbol{A}$

- BayesCG: Probabilistic extension of Conjugate Gradient (CG)
  BayesCG under low-rank Krylov prior = CG
- Accurate error estimates with a few more iterations & matvecs
  S statistic: Produces error estimates of correct magnitude
  especially where traditional bounds are not informative
- Distance of posterior to solution $\triangleq$ absolute error in iterate

Not discussed here

- Numerical implementation
  (reducing re-orthogonalizations, numerical accuracy of posteriors)
- Rigorous & meaningful statistical setting
  (nonlinear dependence of CG on solution, degenerate distributions)

Future Work

- Systematic low-rank approximation of priors
- Error bars/variance for S statistic and other statistics
- Hardwiring termination criteria into posteriors

- Probabilistic numerical Krylov solvers for indefinite/non-symmetric systems, and least squares
- Priors and posteriors designed for capturing numerical uncertainty (roundoff)
- Disintegration: Non-Gaussian distributions