Short Review:
Methods for Bayesian Inverse Problems

IGSSE Project: Bayesian Updating of Engineering models with spatially variable properties

Jonas Latz*

February 22, 2017

Problem setting: Let X, Y be separable Banach spaces, where $\dim Y < \infty$, and $\mathcal{G} : X \to Y$ be the *forward response operator*. Let $u^{\text{truth}} \in X$ be the *true model parameter* and $y := \mathcal{G}(u^{\text{truth}}) + \widehat{\eta}$ be the *observed data*, where $\widehat{\eta}$ is an evaluation of $\eta \sim \mathrm{N}(0, \Gamma)$ and Γ is strictly positive definite. Propose a prior distribution μ_0 on $(X, \mathcal{B}(X))$. Let $u \sim \mu_0$ be distributed according to this prior distribution. Given that further assumptions ([5, Chp. 3]) hold, one can apply *Bayes' Theorem* to derive the *posterior distribution*, which is the goal of *Bayesian Inverse Problems*,

$$\mathbb{P}(u \in \cdot | \mathcal{G}(u) + \eta = y) =: \mu^y,$$

where

$$\frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_0} = \frac{1}{Z(y)} \exp(-\Phi(u; y)), \tag{0.1}$$

$$\Phi(u; y) = \frac{1}{2} \|\mathcal{G}(u) - y\|^2_{\Gamma^{-1}}, \tag{0.2}$$

$$Z(y) = \int \exp(-\Phi(v; y)) \mathrm{d}\mu_0(v). \tag{0.3}$$

In the following, several methods are summarised that allow to approximate the posterior distribution or the expected value of some quantity of interest $Q : X \to \mathbb{R}$ with respect to the posterior distribution; that is the conditional expectation $\mathbb{E}[Q(u)|\mathcal{G}(u) + \eta = y]$.

*Technical University of Munich, Department of Mathematics, Chair of Numerical Mathematics, jonas.latz@tum.de

1 Autonormalised Importance Sampling

Importance sampling methods use samples from the prior distribution to approximate integrals wrt the posterior distribution by applying Bayes' rule. In particular, the prior distribution is used as importance sampling proposal distribution and the likelihood acts as the importance sampling weights. The fundamental idea of this approach is the following way to represent an integral wrt μ^y using integrals wrt μ_0:

$$\mathbb{E}_{\mu^y}[Q] = \int Q \mathrm{d}\mu^y = \frac{1}{Z(y)} \int Q(v) \exp(-\Phi(v;y)) \mathrm{d}\mu_0(v) = \frac{\mathbb{E}_{\mu_0}\left[Q \exp(-\Phi(\cdot;y))\right]}{\mathbb{E}_{\mu_0}\left[\exp(-\Phi(\cdot;y))\right]}. \tag{1.1}$$

Let $(u^{(j)} : j \in \{1, ..., J\}) \sim \mu_0^{\otimes J}$ be samples that are independent and identically distributed (iid.) according to the prior distribution. The *importance sampling estimator* is then defined by

$$E^{\mathrm{IS}}_{\mu^y}[Q] := \frac{\sum_{j=1}^J Q(u^{(j)}) \exp(-\Phi(u^{(j)};y))}{\sum_{j=1}^J \exp(-\Phi(u^{(j)};y))}. \tag{1.2}$$

Advantages

- The method is based on iid. samples drawn from the correct distribution. Depending on the variance of the likelihood (i.e. $\mathrm{Var}(\exp(-\Phi(u;y)))$), the importance sampling estimator might converge faster to the true expected value than the sample mean of a comparable Markov Chain Monte Carlo method.
- A variety of quadrature methods can be applied to approximate the integrals on the rhs of Eq. (1.1). (e.g. Multilevel Monte Carlo (MLMC), Quasi Monte Carlo (QMC))
- The expected values $\mathbb{E}_{\mu_0}\left[Q \exp(-\Phi(\cdot;y))\right]$ and $\mathbb{E}_{\mu_0}\left[\exp(-\Phi(\cdot;y))\right]$ can be constructed using the same set of samples and their $\mathcal{G}$-evaluations.

Drawbacks

- No actual samples of the posterior distribution.
- $\mathbb{E}_{\mu_0}\left[\exp(-\Phi(\cdot;y))\right] = Z(y)$ tends to be very small and its approximation might be infeasible in such cases.

References

Importance sampling in general is discussed for instance in [16, Sec. 2.5] and as a method to solve Bayesian Inverse Problems in [1]. Various quadrature methods are given in [4] and [19].

2 Markov Chain Monte Carlo (MCMC)

Idea: Construct an ergodic Markov Chain $(u^{(j)} : j \in \mathbb{N}) \in X^{\mathbb{N}}$ that has μ^y as a stationary distribution. In that case, Markov Chain's ergodicity implies

$$\frac{1}{J}\sum_{j=1}^{J} Q(u^{(j)}) \to \mathbb{E}_{\mu^y}[Q] \quad (J \to \infty, \text{a.s.}). \tag{2.1}$$

The *Markov Chain Monte Carlo estimator* is given by

$$E_{\mu^y}^{\text{MCMC}}[Q] := \frac{1}{J}\sum_{j=1}^{J} Q(u^{(j)}) \tag{2.2}$$

There are plenty of different ways to construct this Markov Chain. One huge family of MCMC algorithms are Metropolis-Hastings methods.

2.1 Metropolis-Hastings methods

The structure of Metropolis-Hastings methods is given in Algorithm 1.

Algorithm 1: Metropolis-Hastings methods

Given an *acceptance probability* $a : X \times X \to [0,1]$ and a *proposal Markov kernel* $R : \mathcal{B}(X) \times X \to [0,1]$, the Markov Chain $(u^{(j)} : j \in \mathbb{N}) \in X^{\mathbb{N}}$ is constructed in the following way:

Pick $u_0 \in X$ arbitrarily;
for $j \in \mathbb{N}_0$ **do**
  Sample $v \sim R(\cdot|u_j)$;
  Sample $A \sim \text{Unif}[0,1]$ (independently of v);
  if $a(v; u_j) \leq A$ **then**
    $u_{j+1} \leftarrow v$;
  else
    $u_{j+1} \leftarrow u_j$
  end
end

Independence Sampler The *independence sampler* considers the prior distribution $\mu_0 =: Q$ as a proposal Markov kernel. Therefore, the proposals are independent of the chain. The name independence sampler is slightly misleading, since dependence is introduced by the acceptance step, in which the acceptance probability is given by

$$a(u; v) = \min\{1, \exp(\Phi(v) - \Phi(u))\}. \tag{2.3}$$

preconditioned Crank-Nicholson MCMC (pCN MCMC) Let $\mu_0 = \mathrm{N}(0, C)$. The pCN MCMC method is a Metropolis-Hastings-type Monte Carlo method, given by the following proposal and acceptance probability:

$$Q(\cdot|u) = \mathrm{N}(\sqrt{1-\beta^2}u, \beta^2 C), \tag{2.4}$$

$$a(u; v) = \min\{1, \exp(\Phi(v) - \Phi(u))\}. \tag{2.5}$$

2.2 Multilevel Markov Chain Monte Carlo (MLMCMC)

Given any Metropolis-Hastings method, a Multilevel version can be constructed based on the given method straight forwardly. This is discussed in [6, Algorithm 2].

Advantages

- Provides samples from the correct posterior
- The derivation of the normalising constant $Z(y)$ is not required
- The method is generally applicable in any BIP setting

Drawbacks

- The convergence analysis of MCMC is generally difficult
- Monte Carlo Markov Chains usually require a burn-in period, which is both computationally expensive and hard to determine correctly.
- The sample mean of positively correlated samples (as they are given in an MCMC method) converges slower than the sample mean of independent samples, since the variance of the sample mean depends on the variance of the samples and their covariances.
- sampling with MCMC can be infeasible if the posterior distribution is multi modal or its support bad-shaped.

References

A general introduction to MCMC is given in [16, Chp. 5]. The methods above are given in [5, Subs. 5.2] and their theoretical basis [5, Subs. 5.1]. Relevant multilevel methods are discussed in [6] and [4].

3 Particle Filters

Idea: Construct a finite sequence of N measures $(\mu_n)_{n=1}^N$, given by the formula

$$\frac{\mathrm{d}\mu_n}{\mathrm{d}\mu_0}(u) = \frac{1}{Z_n(y)} \exp\left(-\frac{n}{N}\Phi(u;y)\right) \tag{3.1}$$

$$Z_n(y) = \int \exp\left(-\frac{n}{N}\Phi(u;y)\right) \mathrm{d}\mu_0(u). \tag{3.2}$$

Obviously, $\mu_N = \mu^y$. Due to mutual absolute continuity within the sequence, the sequence can also be constructed recursively:

$$\frac{\mathrm{d}\mu_{n+1}}{\mathrm{d}\mu_n}(u) := \frac{Z_n(y)}{Z_{n+1}(y)} \exp(-N^{-1}\Phi(u;y)) \tag{3.3}$$

and can be used to approach the posterior distribution successively - which is especially helpful, when prior and posterior distribution are quite different.

The update from $\mu_n \mapsto \mu_{n+1}$ above occurs naturally in *data assimilation*. In that setting, it is called *filter step*. The sequence of measures is usually approximated by $J \in \mathbb{N}$ particles, due to the complexity of the forward response operator $\mathcal{G}$. Therefore, the methods are called *particle filters*.

3.1 Sequential Monte Carlo and Sequential Importance Sampling

Let P_n be a Markov kernel preserving measure μ_n, i.e. $\mu_n = P_n\mu_n$ (e.g. an update of an MCMC having μ_n as a stationary distribution) and let L be the function mapping $\mu_n \mapsto \mu_{n+1}$ (Eq. (3.3)). Then, $\mu_{n+1} = LP_n\mu_n$.

In practice, the distributions $(\mu_n)_{n=1}^N$ are approximated given particles $(v_n^{(j)} : n \in \{1,...,N\}, j \in \{1,...,J\}) \in X^{N\times J}$ with weights $(w_n^{(j)} : n \in \{1,...,N\}, j \in \{1,...,J\}) \in \mathbb{R}^{N\times J}$, i.e. $\mu_n \approx \mu_n^J = \sum_{j=1}^J w_n^{(j)} \delta_{v_n^{(j)}}$. Sequential Monte Carlo initially draws J independent samples $(v_0^{(j)} : j \in \{1,...,J\})$ from the prior distribution μ_0, that are weighted equally. Thereafter, the samples are moved around in the parameter space X, by applying P_0. Then, by reweighing the particles, they are transformed into weighted particles approximating μ_1. This reweighing is performed in an autonormalised importance sampling manner. The results of these initial steps are the ensemble of particles $(v_1^{(j)} : j \in \{1,...,J\})$, their weights $(w_1^{(j)} : j \in \{1,...,J\})$ and the induced approximate measure $\mu_1^J = \sum_{j=1}^J w_1^{(j)} \delta_{v_1^{(j)}}$. After this initial step, the algorithm continues iteratively: Given $\mu_{n-1}^J, (v_{n-1}^{(j)} : j \in \{1,...,J\})$ and $(w_{n-1}^{(j)} : j \in \{1,...,J\})$, the particles are resampled first, i.e. particles with a weight that is too close to 0 are removed from the ensemble of particles and particles with high weight are split into several particles. In practice, this is done by sampling J times independently from μ_{n-1}^J and assigning equal weights to each of the newly sampled particles. This new particles are then updated analogously to the initial step: The Markov Kernel P_n is applied to each of the particles and the particles are then reweighed as above. The algorithm returns $\mu_N^J =: \mu^{y,\mathrm{SMC}}$, which is a measure approximating the posterior. It can then also be used

to approximate the expected value of some quantity of interest:

$$E_{\mu^y}^{\mathrm{SMC}}[Q] = \int Q \mathrm{d}\mu^{y,\mathrm{SMC}} = \sum_{j=1}^{J} w_N^{(j)} Q(v_N^{(j)}) \tag{3.4}$$

The motivational instruction above is called *Sequential Monte Carlo* (SMC) and stated in pseudo code in Algorithm 2.

Algorithm 2: Sequential Monte Carlo

Given the operator L and the Markov kernels P_n. Let $\mu_0^J := \mu_0$.
for $n \in \{0, 1, ..., N-1\}$ **do**
  Sample $(v_n^{(j)} : j \in \{1, ..., J\} \sim (\mu_n^J)^{\otimes J}$;
  $w_n^{(j)} \leftarrow J^{-1}$; $(j \in \{1, ..., J\})$
  $\mu_n^J \leftarrow \sum_{j=1}^J w_n^{(j)} \delta_{v_n^{(j)}}$;
  Sample $(\widehat{v}_{n+1}^{(j)} : j \in \{1, ..., J\} \sim P_n(\cdot, v_n^{(j)})$; $(j \in \{1, ..., J\})$
  $\widehat{w}_{n+1}^{(j)} \leftarrow \exp(-N^{-1}\Phi(\widehat{v}_{n+1}^{(j)}; y)) w_n^{(j)}$;
  $w_{n+1}^{(j)} \leftarrow \widehat{w}_{n+1}^{(j)} / (\sum_{j=1}^J \widehat{w}_{n+1}^{(j)})$; $(j \in \{1, ..., J\})$
  $\mu_{n+1}^J \leftarrow \sum_{j=1}^J w_{n+1}^{(j)} \delta_{\widehat{v}_{n+1}^{(j)}}$;
end
$\mu^{y,\mathrm{SMC}} := \mu_N^J$

The resampling step is often helpful, but in some cases not required. An equivalent method without resampling is called *Sequential Importance Sampling* (SIS). Some papers suggest to only resample, when the *effective sample size* $\mathrm{ESS} := \frac{1}{\sum_{j=1}^J (w_n^{(j)})^2}$ $(n \geq 1)$ drops below some threshold.

Advantages

- generic way to approximate the posterior distribution with iid samples
- while prior and posterior distribution are often far apart in the measure sense, the sequence of measures given above approaches the posterior slowly, which might improve the efficiency of the importance-sampling-type steps that transfer $\mu_n \mapsto \mu_{n+1}$ and thus improve the accuracy of the posterior approximation

Disadvantages

- No generic ways to estimate an efficient sample size J and step size N.
- the method tends to produce posterior approximations the support of which consists of very few particles

References

Sequential Monte Carlo and Sequential Importance Sampling are discussed in several textbooks and papers, e.g. [16, Chp. 3, Chp. 4], [18], [7]. Sequential Monte Carlo in the inverse problem setting is considered in [5] and [13]. A Multilevel version of sequential Monte Carlo is given in [2].

A more fundamental introduction to filtering in data assimilation is given in [15].

3.2 The Ensemble Kalman Filter

The *Ensemble Kalman Filter* (EnKF) is another particle filter, that has been successfully applied in the Inverse Problem setting. There are different ways to introduce it, two popular interpretations of the method's update step are the following:

- update the particles $v_n^{(j)}$ by updating their position rather than their weight. The particle's weights shall remain uniformly distributed throughout the entire algorithm.
- the EnKF update is a Gaussian type approximation, i.e. the method is equivalent to Sequential Monte Carlo, if $(u, \mathcal{G}(u) + \eta)$ are jointly Gaussian distributed.

The initial ensemble of particles is given by $(v_0^{(j)} : j \in \{1, ..., J\}) \sim \mu_0^{\otimes J}$. Since all particles are weighted equally, a particle update is sufficient. This is given by

$$
\begin{aligned}
v_{n+1}^{(j)} &:= v_n^{(j)} + C_n^{vp}(C_n^{pp} + N\Gamma)^{-1}(y_n^{(j)} - \mathcal{G}(v_n^{(j)})) \quad (j \in \{1, ..., J\}) \\
C_n^{vp} &:= \frac{1}{J}\sum_{j=1}^{J}(v_n^{(j)} - \overline{v_n^{(\cdot)}}) \otimes (\mathcal{G}(v_n^{(j)}) - \overline{\mathcal{G}(v_n^{(\cdot)})}) \\
C_n^{pp} &:= \frac{1}{J}\sum_{j=1}^{J}(\mathcal{G}(v_n^{(j)}) - \overline{\mathcal{G}(v_n^{(\cdot)})}) \otimes (\mathcal{G}(v_n^{(j)}) - \overline{\mathcal{G}(v_n^{(\cdot)})}) \\
y_n^{(j)} &:= y + \eta_n^{(j)} \quad (j \in \{1, ..., J\}, (\eta_n^{(j)} : j \in \{1, ..., J\}) \sim \mathrm{N}(0, N\Gamma)^{\otimes J})
\end{aligned}
$$

Again, the sample mean of the particles given in the last ensemble is used to estimate the quantity of interest, which is the posterior mean in this case $\rightarrow$ a generalisation to other quantities of interest is straight forward.

$$
E_{\mu^y}^{\mathrm{EnKF}}[u] := \frac{1}{J}\sum_{j=1}^{J} v_N^{(j)}. \tag{3.5}
$$

Advantages

- The EnKF estimator is robust (?)

- The EnKF has been applied successfully in various inverse problem settings with a relatively small amount of particles J and steps N.

Drawbacks

- The sample mean of the particles approximates the mean of the posterior. Although, the particles do not necessarily provide a good approximation of the posterior.
- In all the other methods stated above, the same ensemble of particles can be used several times to estimate the expected value of various quantities of interest. That is not possible with the mean field approximation of the Ensemble Kalman Filter.
- The convergence of the EnKF for non-linear $\mathcal{G}$ has not been proven yet.

3.2.1 Bayes Linear estimator

Consider the EnKF given that $N = 1$. In this case, $E_{\mu_y}^{\text{EnKF}}[u]$ is the best linear (wrt y) approximation of the conditional expectation of u given $\mathcal{G}(u) + \eta = y$ and called *Bayes Linear estimator*. It is fundamental in *Bayes Linear Statistics*, which is a collection of approximate methods in Bayesian statistics, some of which can also be applied in the inverse problem setting. The update step of the EnKF can also be interpreted as a Bayes Linear type approximation and thus the EnKF as a sequential Bayes Linear method.

3.3 Multilevel Particle Filter

Multilevel versions of SMC (for inverse problems) and the EnKF (for data assimilation) have been proposed.

References

The Ensemble Kalman Filter for inverse problems is fundamentally introduced in [12]. Various properties and ways to enhance it are given in [20], [17], [14], [10], [11], [8]. A multilevel version of the EnKF in data assimilation is given in [3].

Bayes Linear Statistics is fundamentally introduced in [9]. Several ways to apply Bayes Linear in the Bayesian inverse problem framework are proposed and summarised in [14].

References

[1] S. Agapiou, O. Papaspiliopoulos, D. Sanz-Alonso, and A. M. Stuart. Importance sampling: computational complexity and intrinsic dimension. Submitted.

[2] A. Beskos, A. Jasra, K. Law, R. Tempone, and Y. Zhou. Multilevel Sequential Monte Carlo Samplers. *ArXiv e-prints*, March 2015, 1503.07259.

[3] A. Chernov, H. Hoel, K. Law, F. Nobile, and R. Tempone. Multilevel ensemble Kalman filtering for spatially extended models. *ArXiv e-prints*, August 2016, 1608.08558.

[4] K. A. Cliffe, M. B. Giles, R. Scheichl, and A. L. Teckentrup. Multilevel monte carlo methods and applications to elliptic pdes with random coefficients. *Computing and Visualization in Science*, 14(1):3–15, 2011.

[5] M. Dashti and A.M. Stuart. The Bayesian Approach to Inverse Problems. In R. Ghanem, D. Higdon, and H. Owhadi, editors, *Handbook of Uncertainty Quantification*. Springer.

[6] T. J. Dodwell, C. Ketelsen, R. Scheichl, and A. L. Teckentrup. A hierarchical multilevel markov chain monte carlo algorithm with applications to uncertainty quantification in subsurface flow. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):1075–1108, 2015, http://dx.doi.org/10.1137/130915005.

[7] A. Doucet and A.M. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. In *Handbook of Nonlinear Filtering*. Oxford, UK: Oxford University Press, 2011.

[8] Oliver G. Ernst, Björn Sprungk, and Hans-Jörg Starkloff. Analysis of the ensemble and polynomial chaos Kalman filters in Bayesian inverse problems. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):823–851, 2015.

[9] M. Goldstein and D. Wooff. *Bayes linear statistics, theory and methods*, volume 716. John Wiley & Sons, 2007.

[10] M.A. Iglesias. Iterative regularization for ensemble data assimilation in reservoir models. *Computational Geosciences*, 19(1):177–212, 2015.

[11] M.A. Iglesias. A Regularizing Iterative Ensemble Kalman Method for PDE-constrained Inverse Problems. *Inverse Problems*, 32(2):025002, 2016.

[12] M.A. Iglesias, K.J.H. Law, and A.M. Stuart. The Ensemble Kalman filter for Inverse Problems. *Inverse Problems*, 29(4):045001, 2013.

[13] N. Kantas, A. Beskos, and A. Jasra. Sequential Monte Carlo methods for high-dimensional inverse problems: a case study for the Navier–Stokes equations. 2(1):464–489, 2014.

[14] J. Latz. Bayes linear methods for inverse problems. Master's thesis, University of Warwick,

September 2016.

[15] K.J.H. Law, A.M. Stuart, and K.C. Zygalakis. *Data Assimilation: A Mathematical Introduction*. Springer, 2015.

[16] Jun S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer Publishing Company, Incorporated, 2008.

[17] H.G. Matthies, E. Zander, B.V. Rosić, A. Litvinenko, and O. Pajonk. *Inverse Problems in a Bayesian Setting*, pages 245–286. Springer International Publishing, Cham, 2016.

[18] Ajay Jasra Pierre Del Moral, Arnaud Doucet. Sequential monte carlo samplers. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 68(3):411–436, 2006.

[19] R. Scheichl, A. M. Stuart, and A. L. Teckentrup. Quasi-Monte Carlo and Multilevel Monte Carlo Methods for Computing Posterior Expectations in Elliptic Inverse Problems. *ArXiv e-prints*, February 2016, 1602.04704.

[20] C. Schillings and A.M. Stuart. Analysis of the ensemble Kalman filter for inverse problems. *ArXiv e-prints*, February 2016, 1602.02020.