Accounting for Atomic Data Uncertainty: A Fully Bayesian Approach Using Stan

Nathan Stein

Department of Statistics, The Wharton School, University of Pennsylvania

ISSI Workshop, Bern, Switzerland, April 2016

Data

- Lines with wavelengths $\Lambda = \{\lambda_1, \ldots, \lambda_J\}$
- Have a collection of M atomic data curves sampled from their prior distribution:
$$A_{i\lambda}(n), \quad \lambda \in \Lambda, \quad i = 1, \ldots, M,$$
- Observe intensities $\hat{I}_\lambda$ with known uncertainties σ_λ, $\lambda \in \Lambda$

Statistical Model

Prior distributions:

$$\log n \sim \text{Uniform}(\min = 7,\ \max = 12)$$
$$\log \text{d}s \sim \text{Cauchy}(\text{center} = 9,\ \text{scale} = 5)^a$$
$$Z \sim \text{DiscreteUniform}\left(\{1, \ldots, M\}\right) \quad \text{(index of atomic data curve)}$$

Likelihood:

$$\hat{I}_\lambda \mid Z, n, \text{d}s \sim \text{Normal}\left(A_{Z\lambda}(n)n^2\text{d}s,\ \sigma_\lambda^2\right), \quad \lambda \in \Lambda$$

[a]Note: a flat prior, $p(\log \text{d}s) \propto 1$, yields an improper posterior distribution because likelihood $\rightarrow$ constant > 0 as $\log \text{d}s \rightarrow -\infty$

Interested in...

- Parameter $\theta = (\log n, \log \mathrm{d}s)$
- Data $y = (I_{\lambda_1}, \ldots, I_{\lambda_J})$

Posterior distribution:

$$p(\theta \mid y)$$

- *Fully Bayesian*, as opposed to *pragmatic Bayesian*
- Averages over (posterior) uncertainty in atomic data curves

Posterior probability of ith atomic data curve:

$$\Pr(Z = i \mid y), \quad i = 1, \ldots, M$$

- Data allowed to determine more/less likely atomic data curves

Estimation

Obtain a sample

$$\theta^{(1)}, \ldots, \theta^{(L)} \sim p(\theta \mid y)$$

- Can compute

$$\Pr(Z = i \mid \theta^{(\ell)}, y) \propto \frac{1}{M} p(y \mid Z = i, \theta^{(\ell)})$$

Estimate

$$\widehat{\Pr}(Z = i \mid y) = \frac{1}{L} \sum_{\ell=1}^{L} \Pr(Z = i \mid \theta^{(\ell)}, y)$$

Computation

- Use **Stan** to sample $\theta^{(1)}, \ldots, \theta^{(L)} \sim p(\theta \mid y)$

Stan...

- is a probabilistic modeling language developed by Andrew Gelman and collaborators
- uses Hamiltonian Monte Carlo (HMC) to sample from posterior distributions
- cannot accommodate discrete parameters

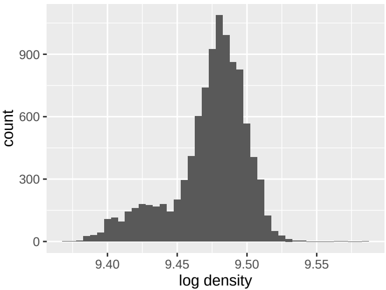
  (so we must analytically marginalize over Z)
- interfaces with R, Python, etc.
- is available at `mc-stan.org`

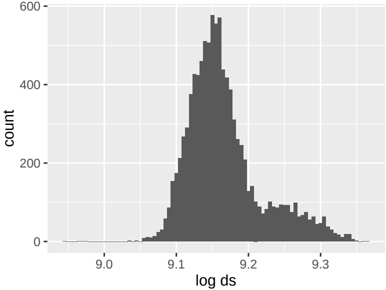
Results

- Data: Fe XIII intensities
- $M = 1000$ curves
- Run 5 chains, 4000 iterations each
- Discard first 2000 iterations of each chain as burn-in
- Total time: 14 minutes (on 1.7 GHz i7 MacBook Air)

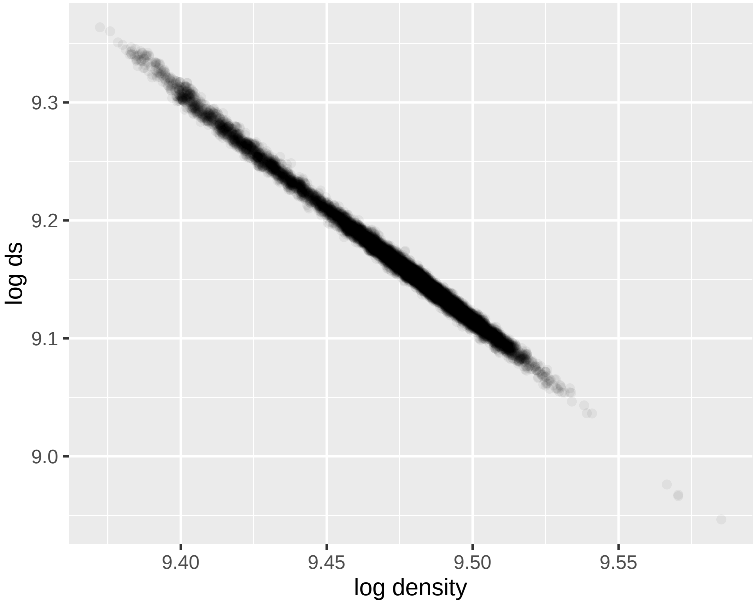
Results

	Posterior median	0.025 quantile	0.975 quantile
$\log n$	9.48	9.40	9.51
$\log ds$	9.16	9.09	9.30

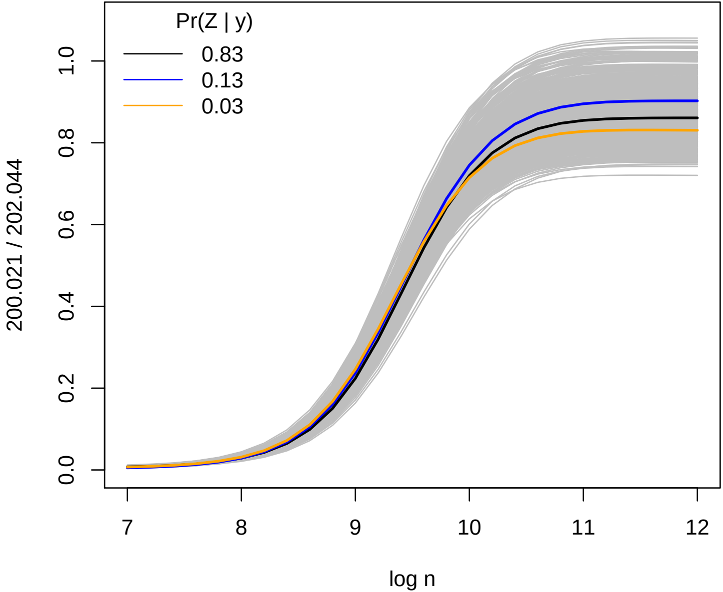




Results



Results


Pr(Z | y)
0.83
0.13
0.03
200.021 / 202.044
0.0
0.2
0.4
0.6
0.8
1.0
7
8
9
10
11
12
log n

Details

$$p(y \mid \theta, Z = i) = \prod_{j=1}^{J} \mathcal{N}\left(\hat{I}_\lambda \;\middle|\; A_{i\lambda}(n) n^2 \mathrm{d}s,\ \sigma_\lambda^2\right)$$

$$p(\theta, Z = i) = p(\theta)\,\frac{1}{M}$$

Notation: $\mathcal{N}(x \mid \mu, \sigma^2)$ is the density of a normal distribution with mean μ and variance σ^2 evaluated at x.

Details

Marginalize over Z:

$$
\begin{aligned}
p(y \mid \theta) &= \sum_{i=1}^{M} p(y \mid \theta, Z = i) \ \Pr(Z = i \mid \theta) \\
&= \sum_{i=1}^{M} \frac{1}{M} \, p(y \mid \theta, Z = i) \\
&= \sum_{i=1}^{M} \exp(c_i)
\end{aligned}
$$

$$
\text{where} \quad c_i = -\log(M) + \sum_{j=1}^{J} \log \mathcal{N}\left(\hat{I}_\lambda \;\middle|\; A_{i\lambda}(n) n^2 \mathrm{d}s, \ \sigma_\lambda^2\right)
$$

Fully Bayesian posterior distribution:

$$
p(\theta \mid y) \propto p(y \mid \theta) \ p(\theta)
$$