Towards automating MCMC algorithms for a class of spatial models

Murali Haran

Department of Statistics
Penn State University

Collaborators: J.M.Flegal (U.C.Riverside), G.L.Jones (U.Minnesota), M.M.Tibbits, J.L.Liechty (Penn State), L.Tierney (U.Iowa).

Duke University

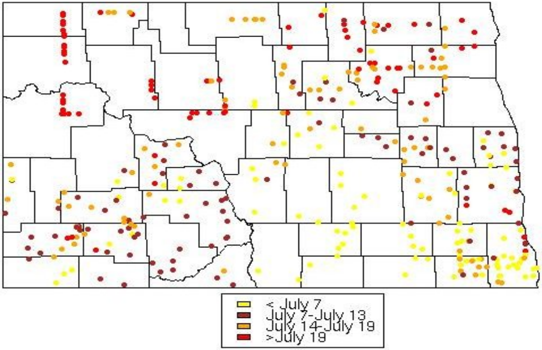
October 2009

Gaussian random fields

- Gaussian random field are very popular models for spatial data.
- Let $\mathbf{s}$ vary over index set $D \subset \mathbb{R}^d$ so the associated spatial process is $\{Z(\mathbf{s}) : \mathbf{s} \in D\}$.
- Here I will be concerned with:
  - Geostatistics: D is a fixed subset of $\mathbb{R}^d$. Process is infinite-dimensional (locations vary continuously in space) but observed at a finite set of locations. e.g. pollutant levels across Pennsylvania only observed at monitoring stations.
  - Areal/lattice data: D is a finite set of locations in $\mathbb{R}^d$, used to represent data often observed on or aggregated up to arbitrary spatial units such as census tracts, counties. e.g. cancer rates by county across Minnesota.

Geostatistics

Wheat flowering dates in North Dakota (for studying blight epidemics).

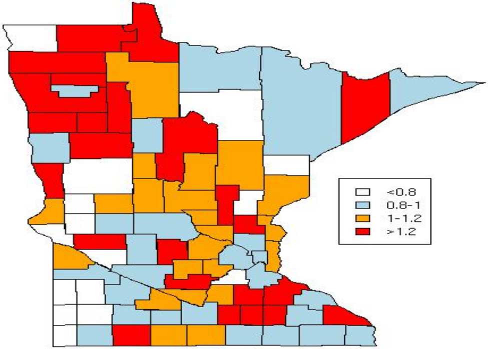


Courtesy Plant Pathology, PSU and North Dakota State.

(Haran, Bhat, Molineros, DeWolf, 2008)

Areal data

Minnesota cancer rates by county: $\frac{\text{observed}}{\text{expected}}$ counts



Courtesy MN Cancer Surveillance System, Dept. of Health

(Haran, Hodges, Carlin, 2003)

Gaussian random field models

- Very widely used by statisticians and non-statisticians, including in many non-spatial contexts. E.g. Gaussian process-based models used for emulating complex computer experiments, machine learning, nonparametric regression, classification.
- Designing efficient Markov chain Monte Carlo (MCMC) algorithms for such models can be challenging.
- Automated, reliable algorithms for even a few specific models will be very useful for people who want to fit these models routinely.

Basic Gaussian random field (linear) model

- Spatial process at location $\mathbf{s}$ is $Z(\mathbf{s}) = \mu(\mathbf{s}) + w(\mathbf{s})$ where:
  - $\mu(\mathbf{s})$ is the mean. Often $\mu(\mathbf{s}) = X(\mathbf{s})\boldsymbol{\beta}$, $X(\mathbf{s})$ are covariates at $\mathbf{s}$ and $\boldsymbol{\beta}$ is a vector of coefficients.
- Model dependence among spatial random variables by imposing it on the errors (the $w(\mathbf{s})$'s).
- For n locations, $\mathbf{s}_1, \ldots, \mathbf{s}_n$, $\mathbf{w} = (w(\mathbf{s}_1), \ldots, w(\mathbf{s}_n))^T$ can be jointly modeled via a zero mean Gaussian process (GP) for geostatistics, or Gaussian Markov random field (GMRF) for areal/lattice data.
- Gaussian Process (GP): Let Θ be the parameters for covariance matrix $\Sigma(\Theta)$. Let $\mathbf{Z} = (Z(\mathbf{s}_1), \ldots, Z(\mathbf{s}_n))^T$. Then:

$$\mathbf{Z}|\Theta, \boldsymbol{\beta} \sim N(\mathbf{X}\boldsymbol{\beta}, \Sigma(\Theta))$$

Spatial linear model (contd.)

- Gaussian Markov Random field (GMRF): Let Θ be the parameters for precision matrix $Q(\Theta)$. Then:

$$\mathbf{Z}|\Theta, \boldsymbol{\beta} \sim N(\mathbf{X}\boldsymbol{\beta}, Q^{-1}(\Theta))$$

- For some popular forms of the Gaussian Markov random field the precision matrix is singular so:

$$f(\mathbf{Z}|\Theta, \boldsymbol{\beta}) \propto c(\Theta) \exp\left(-\frac{1}{2}(\mathbf{Z} - \mathbf{X}\boldsymbol{\beta})^T Q(\Theta)(\mathbf{Z} - \mathbf{X}\boldsymbol{\beta})\right).$$

- For spatial linear model, once priors for $\Theta, \boldsymbol{\beta}$ specified, inference is based on posterior $\pi(\Theta, \boldsymbol{\beta} \mid \mathbf{Z})$.
- Key observation: Θ typically has low dimensions (2-5) for both GP and GMRF models, while dimensions of $\mathbf{Z}$ can be large.

Spatial generalized linear model

What if data generating mechanism is non-Gaussian (Diggle et al., 1998):

- Stage 1: Model $Z(\mathbf{s}_i)$ conditionally independent with distribution f given parameters $\boldsymbol{\beta}, \Theta$, spatial errors $w(\mathbf{s}_i)$

  $$f(Z(\mathbf{s}_i)|\boldsymbol{\beta}, \Theta, w(\mathbf{s}_i)),$$

  where $g(E(Z(\mathbf{s}_i))) = \eta(\mathbf{s}_i) = X(\mathbf{s}_i)\boldsymbol{\beta} + w(\mathbf{s}_i), \eta$ is a canonical link function (for example the logit link).
- Stage 2: Again $\mathbf{w} = (w(\mathbf{s}_1), \ldots, w(\mathbf{s}_n))^T$. Model $\mathbf{w}$ as spatially dependent either via a GP or GMRF.
- Stage 3: Priors for $\Theta, \boldsymbol{\beta}$.
- Inference based on $\pi(\Theta, \boldsymbol{\beta}, \mathbf{w} \mid \mathbf{Z})$.

MCMC for posterior inference

Goal: estimate $E_\pi g$ for real valued functions g.

MCMC: Construct a Harris-ergodic Markov chain $X_1, X_2, \ldots$ with stationary distribution π so that if $E_\pi |g(x)| < \infty$:

$$\bar{g}_n = \sum_{i=1}^{n} g(X_i)/n \to E_\pi g$$

When simulating from π careful practitioners face several (time consuming) issues:

- Devising/tuning the Metropolis-Hastings algorithm.
- Starting values?
- How long to run the Markov chain?
- Accuracy of the estimator is hard to estimate.

Automation of MCMC

Ideally (the holy grail):

- Automated approach for constructing algorithm. No tuning necessary.
- Generate appropriate starting values automatically.
- Have a rigorous criteria for determining when to stop the chain that is *related to inferential goals*. E.g. How accurate do you want your estimates to be?
- Some theoretical guarantees regarding all of the above.

Options

1. Exact sampling:
    - *Perfect* draws using a Markov chain (Propp-Wilson, 1996).
    - Make classical (old fashioned) Monte Carlo methods such as rejection sampling practical.
2. Construct Metropolis-Hastings so Markov chain sampler mixes well (e.g. uniformly or geometrically ergodic):
    - Rigorous approach for estimating Monte Carlo standard errors and stopping rules.

Option (1): very hard for Gaussian random field models. When achievable, far less efficient than corresponding MCMC sampler.

Option (2): hard to achieve, challenging analytical work.

Efficient MCMC for spatial linear models

- Closed form for low-dimensional (usually 2-8) marginal posterior, $\pi(\Theta, \boldsymbol{\beta} \mid \mathbf{Z})$. Slice samplers (Agarwal, Gelfand 2005; Yan et al., 2007) involve univariate updates.
- To improve mixing, can use block updates: multivariate slice sampling (Tibbits, Haran, Liechty, 2009) resulting in a faster mixing algorithm. Since search for proposals at each step of the algorithm is very expensive, this is done in parallel on a graphical processing unit (GPU).
- Aside: matrix operations at each iteration are expensive for large data sets. Need to take advantage of some form of sparsity (banded matrices, tapering, reduced-rank approaches).

Efficient MCMC for spatial generalized linear models

Computing for SGLMs is more challenging:

- Higher dimensional posterior, $\pi(\Theta, \boldsymbol{\beta}, \mathbf{w} \mid \mathbf{Z})$.
- Strong dependence among components (e.g. spatial random effects) results in slow mixing chains when using univariate update MCMC approaches. Univariate updates are also more computationally expensive than block updates when matrices are dense.
- Block sampling can solve these issues.
- Challenges with block sampling: (1) hard to construct good block proposals, (2) matrix operations at each iteration can be very expensive.

Efficient MCMC for SGLMs [cont'd]

- Block sampling approaches are generally based on a multivariate normal approximation for at least part of the distribution.
- Two routes for constructing MCMC updates:
  - Approximate SGLM by linear spatial model.
  - Langevin-Hastings (Roberts and Tweedie, 1996)

Linearization of an SGLM

- Target posterior of SGLM is $\pi(\Theta, \boldsymbol{\beta}, \mathbf{w} \mid \mathbf{Z})$.
- Approximate the SGLM by a linear spatial model:
  - Transform data **Z** to **Y** and use approximation of form:

    $$\mathbf{Y} \mid \Theta, \boldsymbol{\beta}, \mathbf{w} \sim N(\mathbf{X}\boldsymbol{\beta} + \mathbf{w}, C).$$

  - As before $\mathbf{w} \mid \Theta \sim N(0, \Sigma(\Theta))$. Denote posterior for this approximate model by $S(\Theta, \boldsymbol{\beta}, \mathbf{w})$.
- Analytically integrate: $S_1(\Theta, \boldsymbol{\beta}) = \int S(\Theta, \boldsymbol{\beta}, \mathbf{w}) d\mathbf{w}$.
- From $S(\Theta, \boldsymbol{\beta}, \mathbf{w})$, can obtain approximate conditional distribution of spatial random effects, $S_2(\mathbf{w} \mid \Theta, \boldsymbol{\beta})$ (multivariate normal). Then, we have

  $$S(\Theta, \boldsymbol{\beta}, \mathbf{w}) = S_1(\Theta, \boldsymbol{\beta}) S_2(\mathbf{w} \mid \Theta, \boldsymbol{\beta}).$$

[Haran (2003), Haran and Tierney (2009)]

Approximation

Construct heavy-tailed approximation $\hat{\pi}(\Theta, \boldsymbol{\beta}, \mathbf{w})$:

- We have: $S_1(\Theta, \boldsymbol{\beta}) S_2(\mathbf{w} \mid \Theta, \boldsymbol{\beta}) \approx \pi(\Theta, \boldsymbol{\beta}, \mathbf{w} | Y)$.
- Find heavy-tailed approximation to $S_1(\Theta, \boldsymbol{\beta})$: $\hat{\pi}_1(\Theta, \boldsymbol{\beta})$.
- Find heavy-tailed (multi-t) approximation to $S_2(\mathbf{w} | \Theta, \boldsymbol{\beta})$, $\hat{\pi}_2(\mathbf{w} | \Theta, \boldsymbol{\beta})$. Easy: multivariate-t with same mean and variance as the multivariate normal $S_2(\mathbf{w} | \Theta, \boldsymbol{\beta})$.

A joint proposal distribution

- $\hat{\pi}(\Theta, \boldsymbol{\beta}, \mathbf{w})$: proposal for Monte Carlo algorithms.
- Can sample sequentially from $\hat{\pi}$:
  1. Sample $\Theta, \boldsymbol{\beta} \sim \hat{\pi}_1(\Theta, \boldsymbol{\beta})$.
  2. Sample $\mathbf{w} \mid \Theta, \boldsymbol{\beta} \sim \hat{\pi}_2(\mathbf{w} \mid \Theta, \boldsymbol{\beta})$.
- Note:
  - Step 1 is easy (fast) since low-dimensional.
  - Step 2: Expensive matrix operations involved when generating proposal, evaluating Met-Hastings ratio. Fast if sparse/reduced-rank: GMRFs (banded; Rue, 2001), covariance tapering GPs (Furrer et al. 2006, Kaufman et al., 2008), reduced rank, process convolutions GPs (Higdon, 1998; Cressie and Johanessen, 2008; Banerjee et al., 2008).

Example: disease mapping

Besag, York, Mollie (1991) model:

- $Z(\mathbf{s}_i)|\mu(\mathbf{s}_i) \sim$ Poisson$(E(\mathbf{s}_i)e^{\mu(\mathbf{s}_i)}),\ i = 1, ..., N$, where
  - $\mu(\mathbf{s}_i)$, log-relative risk: $\mu(\mathbf{s}_i) = \theta(\mathbf{s}_i) + \phi(\mathbf{s}_i)$.
  - $E(\mathbf{s}_i)$: expected number of events in region i (known).
  - $\theta(\mathbf{s}_i)$'s are non-spatial $\theta(\mathbf{s}_i)|\tau_h \overset{iid}{\sim} N(0, 1/\tau_h)$.
  - $\phi(\mathbf{s}_i)$'s form a GMRF.
    $f(\phi \mid \tau_c) \propto \tau_c^{(N-1)/2} \exp\left(-\frac{1}{2}\phi^T Q(\tau_c)\phi\right)$, where
    $\boldsymbol{\phi} = (\phi(\mathbf{s}_1), \ldots, \phi(\mathbf{s}_N))$ and $Q(\tau_c)$ is an adjacency matrix.
  - Add priors for the precision parameters τ_h, τ_c, Inverse Gammas.

Posterior: $\pi(\boldsymbol{\theta}, \boldsymbol{\phi}, \tau_h, \tau_c|\mathbf{Z})$, of $2N + 2$ dims.

Approximation for disease mapping model

Recap:

- Our goal is to construct an automated sampler for Gaussian random field models.
- We have a general approach for constructing a heavy-tailed approximation to $\pi(\boldsymbol{\theta}, \phi, \tau_h, \tau_c | \mathbf{Z})$.

For disease mapping example:

- Set $Y_i = \log(Z(\mathbf{s}_i)/E_i), i = 1, \ldots, N$, to obtain transformed vector $\mathbf{Y}$.
- Can use a normal approximation for $\mathbf{Y}$ and utilize approach outlined before to obtain heavy-tailed approximation, $\hat{\pi}(\boldsymbol{\theta}, \phi, \tau_h, \tau_c \mid \mathbf{Y})$.

Automated MCMC for disease mapping example

- Generate starting values from $\hat{\pi}$: genuinely overdispersed with respect to π (cf. Gelman and Rubin, 1992).
- Construct a Metropolis-Hastings ‘independence sampler’ (cf. Tierney, 1994): propose every M-H update from $\hat{\pi}$.
    - The resulting sampler is provably fast mixing: it is *uniformly ergodic* (Haran and Tierney, 2009).
    - The sampler is very easy to parallelize (‘embarrassingly parallel’).
    - Since GMRF, can use sparse matrix algorithms for fast computing (Rue, 2001).

Stopping rules, estimating standard errors

Since the Markov chain is uniformly ergodic, can obtain rigorous estimates of standard errors for expectations based on MCMC runs as well as rigorous stopping rules for MCMC.

- Under mild moment conditions, Central Limit theorem holds for estimate of expectations based on these samplers (cf. Roberts and Rosenthal, 2004).
- Consistent batch means (Jones, Haran, Caffo, Neath, 2006) provides a *consistent* estimate of the Monte Carlo standard error. Simple stopping rule (**'fixed width' approach**): When estimated standard error is below a desired level, stop the sampler. Works well in practice (Flegal, Haran, Jones, 2008; Jones et al., 2006).

Data examples

- Minnesota cancer data sets: 176 parameters. Infant mortality: 910 parameters.
- Note: can also use rejection sampler/E-sup rejection sampler (Caffo et al., 2001) with same proposal.
- Stop algorithms when Monte Carlo standard errors are below same threshold for parameters.

data set	sample size		time taken	
	rejection	I-MH	rejection	I-MH
breast cancer	4,118	29,241	2,663s	183s
colo-rectal cancer	4,735	27,225	543s	170s
infant mortality	—	97,721	—	10,066s

Disease mapping: summary of results

- Both samplers: good estimates, similar inference. (surprising that rejection sampler works in some cases.)
- I-MH is vastly superior to rejection sampler. For large data set (910-dimensional posterior): I-MH is still practical while rejection sampler is not. Timing for I-MH can be reduced linearly according to number of processors available.
- Almost like iid Monte Carlo: know that CLT holds (generally not true for MCMC), have consistent standard error estimates, easy to determine starting values, stopping rule.

Criticism: Approximation tries to match entire posterior distribution using a multivariate normal approximation for one stage in the hierarchical model. May work poorly in some examples.

Example: zero-inflated SGLM for ecology

- Colorado Potato Beetles counts data (PSU Ecology).
- Of interest: (a) determining spatial field corresponding to incidence — binary outcomes, (b) determining spatial field corresponding to prevalence — counts.
- A two-stage model (Recta, Haran, Rosenberger, 2009) based on knowledge of data generating mechanism:
  - Incidence: $U(\mathbf{s})$, binary with logit link and latent GP.
  - Prevalence: $V(\mathbf{s})$ is defined only when $U(\mathbf{s}) = 1$. $V(\mathbf{s})$ has spatial truncated Poisson distribution with log link and latent GP. Observe: $Z(\mathbf{s})$ is $V(\mathbf{s})$ if $U(\mathbf{s}) = 1$, else 0.
  - Separate regressions/predictors for incidence and prevalence, separate posterior predictive distribution for each process.

Efficient MCMC for zero-inflated SGLM

- Normal approximation will generally not work well here.
- Better off exploring *local* normal approximations instead. e.g. Langevin-Hastings MCMC.
- Langevin-Hastings MCMC: construct joint local approximations using gradients at current state of Markov chain (Roberts and Tweedie, 1996; Christensen, Roberts, Sköld, 2006).
- The resulting Markov chain is geometrically ergodic under certain conditions (Christensen, Møller, Waagepetersen, 2001).
- Can obtain rigorous standard error estimates and stopping rules, as before.
- This works well in practice for our SGLM.

Observations from applications

- Independence Metropolis-Hastings with heavy-tailed proposal, Langevin-Hastings algorithm are useful for real data examples. Can be rigorous about errors and automated stopping rules.
- Even when we have not established mixing rates, automation, fixed width approach can be very useful:
  - Can use it for simulation studies — e.g. comparing posterior quantities for a model over randomly generated data sets.
  - Automatically runs longer when Markov chain mixes poorly, or expectation is challenging (e.g. tail probabilities). Shorter if we do not want much accuracy, chain is fast mixing.
  - More reliable than some other convergence diagnostics (Flegal, Haran, Jones, 2008).

Summary

- Spatial linear and generalized linear models are a flexible, useful class of models.
- Possible to construct efficient MCMC algorithms using approximations and recent theoretical developments:
  - Consistent estimate of standard errors.
  - Known mixing properties (uniformly ergodic).
  - Automatically generate starting values.
  - Simple, rigorous stopping rule (‘fixed width’): when desired accuracy is attained, stop.
  - Useful to take advantage of sparsity when possible.
  - Parallel computing can make an impractical algorithm/model practical.

Select references

- Christensen, Møller, Waagepetersen (2001), “Geometric ergodicity of Metropolis-Hastings algorithms for conditional simulation in generalized linear mixed models,” Methodology and Computing in Applied Prob.
- Haran, M. and Tierney, L. (2009) “Automated Markov chain Monte Carlo for a class of spatial models.”.
- Haran, M. (2009) “Gaussian random field models for spatial data.” *Handbook of Markov chain Monte Carlo* (to appear).
- Tibbits, M.M., Haran, M., Liechty, J.L. (2009) “Parallel multivariate slice sampling.”
- Flegal, J., Haran, M., and Jones, G.L. (2008) "Markov chain Monte Carlo: Can we trust the third significant figure?" *Stat. Sci.*, `www.stat.psu.edu/~mharan/batchmeans.R`
- Jones, G.L., Haran, M., Caffo, B.S. and Neath, R. (2006). “Fixed Width Output Analysis for Markov chain Monte Carlo,” *JASA*.
- Recta, V.L., Haran, M., Rosenberger, J.L. (2009) “A two-stage model for incidence and prevalence in point-level spatial count data.”