

# **Bayesian inference for spatial GP models:**

## **Part 2**

---

Abhi Datta

Department of Biostatistics, Bloomberg School of Public Health, Johns Hopkins University, Baltimore, Maryland

## Review of last lecture

- Gibbs sampler: Generate samples from  $p(X_1, X_2)$  by iteratively generating samples from  $p(X_1^{(t)}|X_2^{(t-1)})$  and  $p(X_2^{(t)}|X_1^{(t)})$
- Implementing our own Gibbs sampler for the **unmarginalized model**:  $y \sim N(X\beta + w, \tau^2 I)$ ,  $w \sim N(0, \sigma^2 R(\phi))$ 
  - Assume  $\phi$  is known
  - **Priors:**  $\sigma^2 \sim IG(a_\sigma, b_\sigma)$ ,  $\tau^2 \sim IG(a_\tau, b_\tau)$  and  $\beta \sim N(\mu, V)$
  - Full conditionals:

$$\beta | \sigma^2, \tau^2, w, y \sim N(\mu^*, V^*)$$

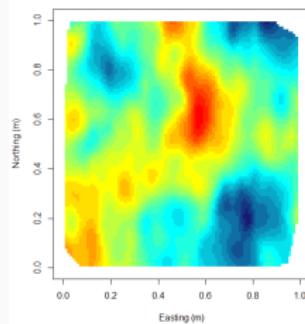
$$w | \sigma^2, \tau^2, \beta, y \sim N(m, C^*)$$

$$\sigma^2 | \beta, \tau^2, w, y \sim IG(a_\sigma^*, b_\sigma^*)$$

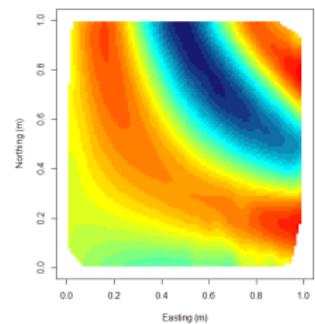
$$\tau^2 | \beta, \sigma^2, w, y \sim IG(a_\tau^*, b_\tau^*)$$

# Data analysis

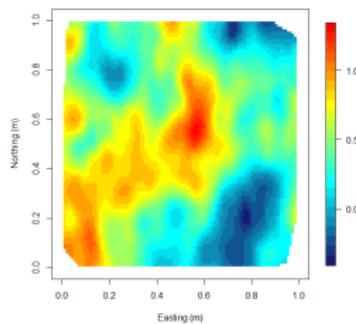
- Dataset 3 from Lecture 1
- True model:  $y(s) \sim N(0.2 - 0.3x(s) + w(s), 0.01)$ ,  
 $w(s) \sim GP$ ,  $\text{Cov}(w(s_i), w(s_j)) = 0.25 * \exp(-2||s_i - s_j||)$



$y(s)$



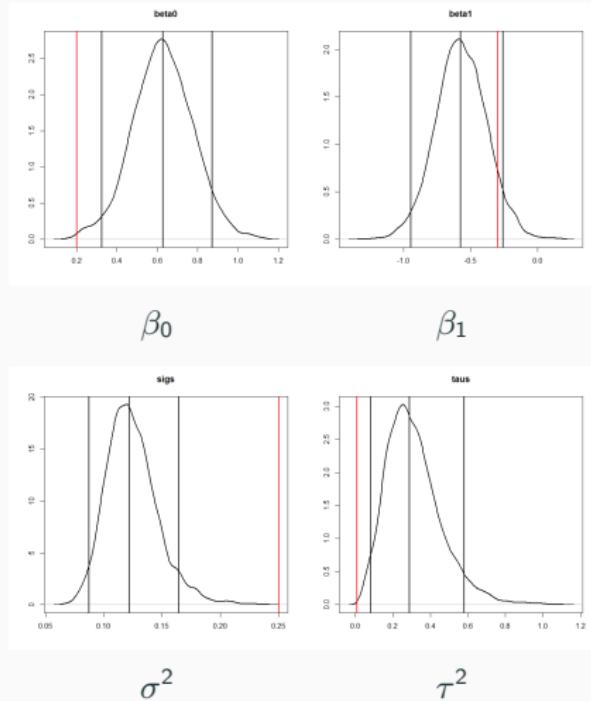
$x(s)$



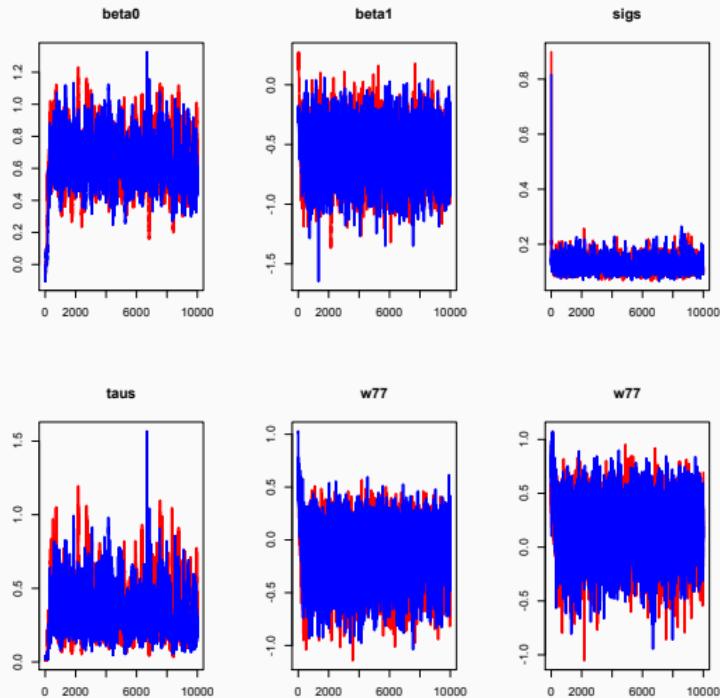
$w(s)$

# Parameter posteriors

- $\phi$  is kept fixed at 4.23 (estimated value from variogram fitting)
- Gibbs sampler for  $w$ ,  $\beta$ ,  $\sigma^2$  and  $\tau^2$



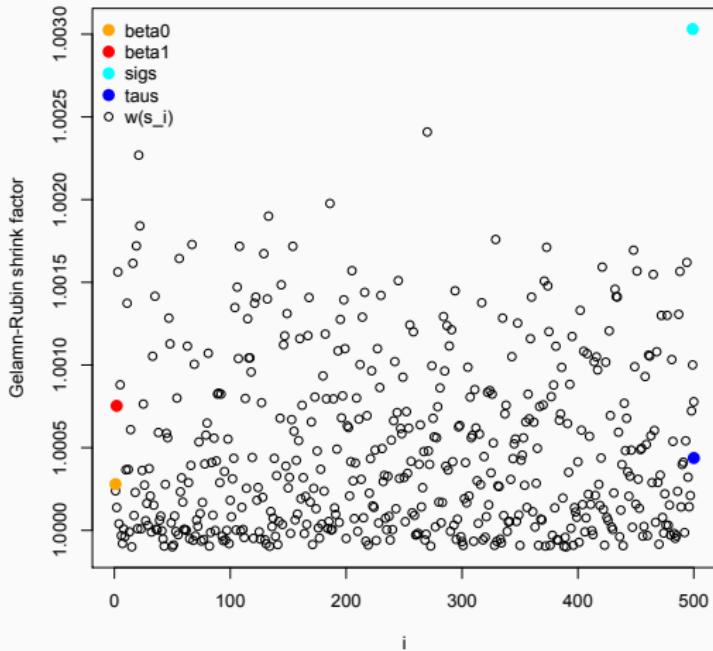
# Convergence diagnostics: Trace plots



## Convergence diagnostics: Gelman-Rubin shrink factor

- Run chains of length  $N$  with overdispersed initial values
- Discard the first  $N_b$  draws of each chain as burn-in
- For each variable  $\theta$ , calculate the **within-chain** variance
$$W = \frac{1}{m} \sum_{j=1}^m \frac{1}{N-N_b-1} \sum_{i=N_b+1}^N (\theta^{(ij)} - \bar{\theta}_j)^2$$
- For each variable  $\theta$ , calculate the **between-chain** variance
$$B = \frac{N-N_b}{m-1} \sum_{j=1}^m (\bar{\theta}_j - \bar{\theta})^2 \text{ where } \bar{\theta} = \frac{1}{m} \sum_{j=1}^m \bar{\theta}_j$$
- Calculate the **Gelman-Rubin shrink factor** as
$$R = \sqrt{\frac{(1 - \frac{1}{N-N_b})W + \frac{1}{N-N_b}B}{W}}$$
- $R > 1.1$  or  $1.2$  indicates lack of convergence
- **coda** package in R gives the GR-shrink factors for each variable

# Convergence diagnostics: Gelman-Rubin shrink factor



**Figure:** GR shrink factor for dataset 3

# Convergence diagnostics: Plots of Gelman-Rubin shrink factor

- Plots of GR shrink factor against MCMC iteration number can help determine the burn-in

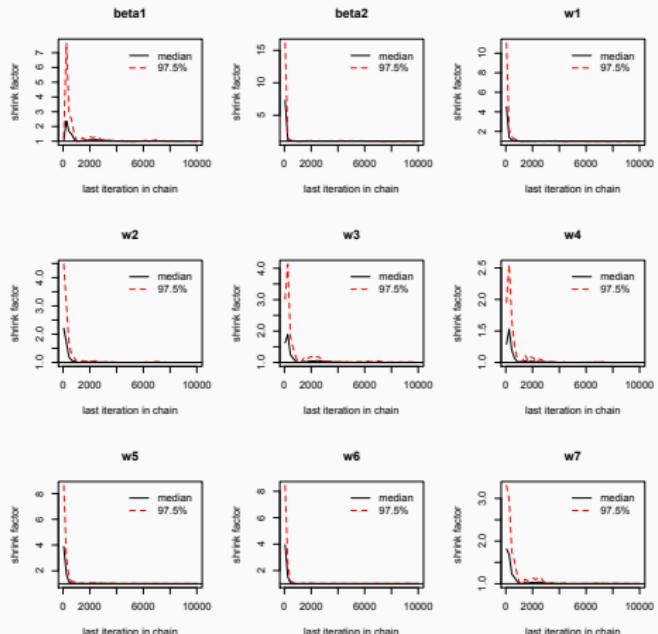


Figure: GR shrink factor a a function of MCMC iteration for dataset 3

# Convergence diagnostics: Plots of Gelman-Rubin shrink factor

- Plots of GR shrink factor against MCMC iteration number can help determine the burn-in

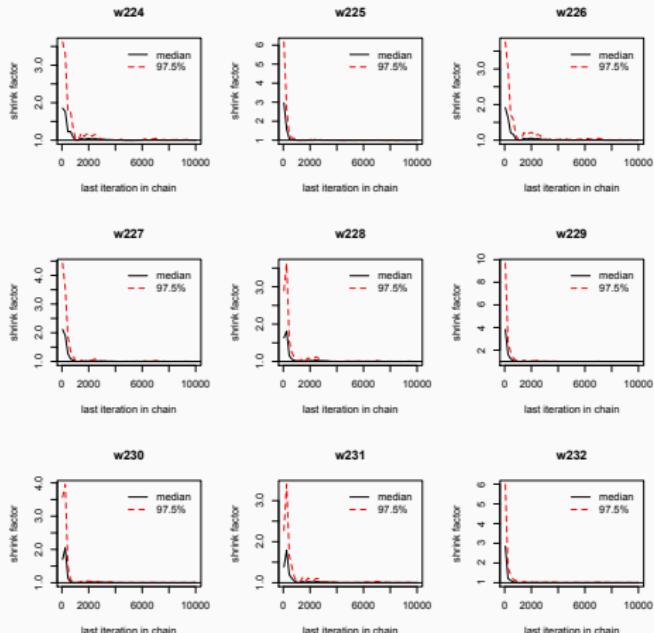
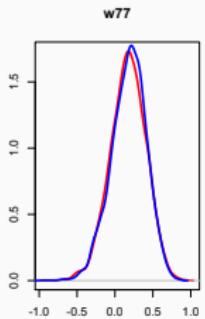
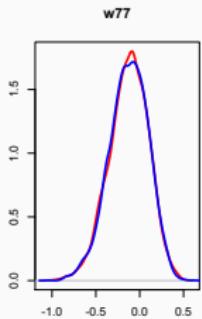
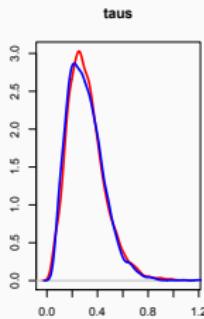
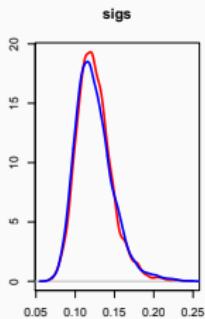
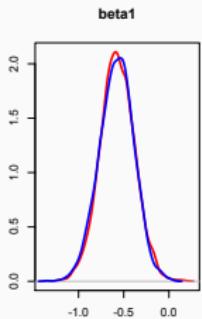
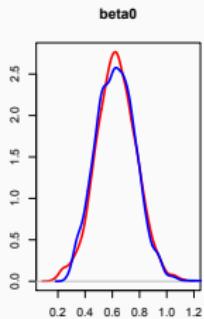


Figure: GR shrink factor a a function of MCMC iteration for dataset 3

# Convergence diagnostics: Density plots



## Model comparison using Bayesian output

- With holdout data we can use RMSPE (using posterior means or medians), out-of-sample CP (coverage probability) and CIW (confidence interval width) based on posterior quantiles

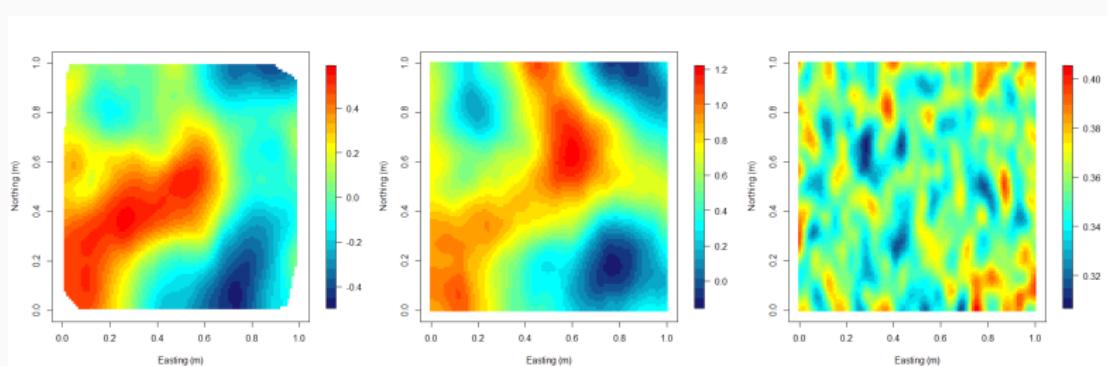
## Model comparison using Bayesian output

- Deviance Information Criterion (DIC) (Spiegelhalter, 2002)  
uses the posterior samples for model comparison
- DIC for the model  $I(y|\theta)$  is based on the deviance  
$$D(y, \theta) = -2 \log I(y|\theta)$$
- $\bar{D}(y) = E(D(y, \theta)|\theta) \approx \frac{1}{N} \sum_{i=1}^N D(y, \theta^{(i)})$
- $p_D = \bar{D}(y) - D(y, \bar{\theta})$  where  $\bar{\theta} = E(\theta|y) \approx \frac{1}{N} \sum_{i=1}^N \theta^{(i)}$
- $DIC = \bar{D}(y) + p_D$
- The  $p_D$  term can be interpreted as effective number of parameters in the model and hence penalizes more complex models (similarity to AIC and BIC)

## Posterior predictive distributions

- For the unmarginalized model, posterior samples for  $w(s)$  are already generated for all  $s$  in the training data locations  $S$
- Posterior predictive distributions  $\tilde{y}(s)$  can be obtained using composition sampling:
  - If  $s_0 \notin S$ , generate samples from  $w(s_0) | y$  using  $w(s_0)^{(j)} | \cdot \sim N(r(s_0)'R^{-1}w^{(j)}, \sigma^{2(j)}(1 - r(s_0)'R^{-1}r(s_0)))$
  - $r(s_0) = \text{cor}(w(s_0), w)$  and  $R = \text{cor}(w)$
  - If  $\phi$  was also sampled, replace  $r$  and  $R$  by  $r^{(j)}$  and  $R^{(j)}$
  - For any  $s$ , generate  $\tilde{y}(s)^{(j)} = N(x(s)'\beta^{(j)} + w(s)^{(j)}, \tau^{2(j)})$

# Posterior surfaces



$w(s) | y$

$\tilde{y}(s) | y$

$\text{var}(\tilde{y}(s) | y)$

## Marginalized model

- Unmarginalized model has  $n$  additional parameters ( $w$ )
- May lead to slow MCMC convergence
- Marginalized model:  $y \sim N(X\beta, \sigma^2 R(\phi) + \tau^2 I)$
- **Pros:** Only  $p + 3$  parameters
- **Cons:** Even the full conditionals are not useful (except for  $\beta$ )
- How to do MCMC?

## Metropolis algorithm

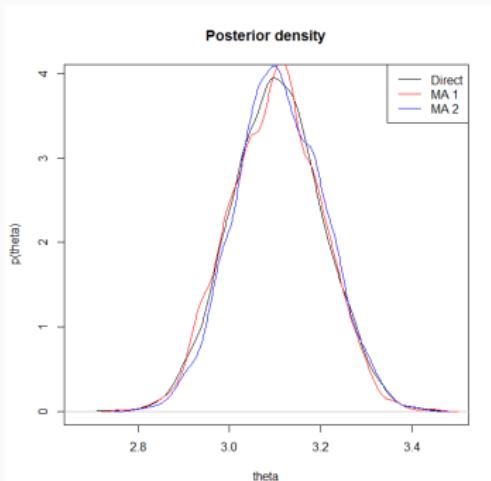
- We want to draw sample from a density  $p(\theta) = f(\theta)/K$
- Begin with an initial  $\theta^0$
- Choose a function  $q(x | y)$  such that
  - $q(x | y)$  is a valid density function in  $x$  for every value of  $y$
  - $q(x | y) = q(y | x)$
  - e.g.  $q(x | y) \sim N(x | y, \lambda) = \frac{1}{\sqrt{2\pi\lambda}} \exp(-\frac{1}{2\lambda}(x - y)^2)$
  - If  $\theta$  is multivariate one can choose  $q(x | y) \sim N(x | y, \Sigma)$
- $q$  is called the **proposal density**
- If  $\theta$  is multivariate, choose  $q$  to be a multivariate proposal density

## Metropolis algorithm

- At the  $i^{th}$  iteration, generate  $\theta^*$  from  $q(\cdot | \theta_{i-1})$
- Calculate the ratio  $r = f(\theta^*)/f(\theta_{i-1})$
- If  $r \geq 1$ , accept the new value i.e  $\theta_i = \theta^*$
- If  $r < 1$ :
  - Accept the new value i.e  $\theta_i = \theta^*$  with probability  $r$
  - Keep the old value i.e  $\theta_i = \theta_{i-1}$  with probability  $1 - r$
- The sample  $(\theta_i)_{i=N_b}^N$  is a sample from  $p(\theta)$  where  $N_b$  is a burn-in period used
- An overall rate of acceptance around 30% – 50% is desirable (controlled by the **tuning** parameter  $\lambda$ )

## Example

- $Y_i \stackrel{\text{iid}}{\sim} N(\theta, \sigma^2)$  for  $i = 1, \dots, n$  where  $\theta = 3$  (unknown),  $\sigma^2 = 1$ (known) and  $n = 100$
- Prior:  $\theta \sim N(\mu, \tau^2)$  where  $\mu = 0$  and  $\tau^2 = 10$
- Metropolis algorithm:  
$$p(\theta | Y) \propto \exp\left(-\frac{n}{2\sigma^2} (\bar{y} - \theta)^2 - \frac{1}{2\tau^2} (\theta - \mu)^2\right)$$
- Direct approach:  $\theta | Y \sim N\left(\frac{\frac{n\bar{y}}{\sigma^2} + \frac{\mu}{\tau^2}}{\frac{n}{\sigma^2} + \frac{1}{\tau^2}}, \frac{1}{\frac{n}{\sigma^2} + \frac{1}{\tau^2}}\right)$



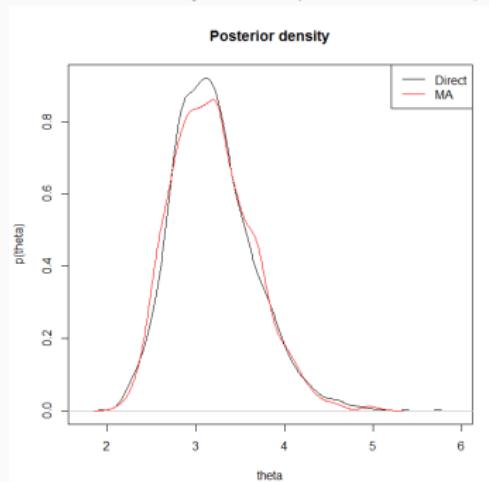
## Jacobian adjustment

- Often the parameter of interest  $\theta$  is not supported on the entire real line but on a part of it e.g.  $[0, 1]$ ,  $(0, \infty)$  etc.
- The normal proposal density is easy to use but has the entire real line as support
- One can choose a transformation  $g$  such that  $\eta = g(\theta)$  is supported on the real line
- Generate new  $\eta^*$  using the normal proposal density
- Use the inverse transformation to obtain  $\theta^* = g^{-1}(\eta^*)$
- The likelihood for  $\eta$  will be given by  $p(\eta) = p(\theta)/|g'(\theta)|$
- Calculate

$$r = p(\eta^*)/p(\eta_{i-1}) = p(\theta^*)/p(\theta_{i-1}) \times |g'(\theta_{i-1})|/|g'(\theta^*)|$$

## Example

- $Y_i \stackrel{\text{iid}}{\sim} N(0, \sigma^2)$  for  $i = 1, \dots, n$  where  $\sigma^2 = 4$  (unknown)
- $\sigma^2$  is supported on  $(0, \infty)$ . So, we use log transformation
- Prior:  $\sigma^2 \sim \text{IG}(\alpha, \beta)$  where  $\alpha = 2$  and  $\beta = 1$
- Metropolis algorithm:  
$$p(\sigma^2 | Y) \propto (\sigma^2)^{-1-\alpha-n/2} \exp(-(\beta + \sum_{i=1}^n y_i^2 / 2) / \sigma^2)$$
- Direct approach:  
$$\sigma^2 | Y \sim \text{Inverse Gamma}(\alpha + n/2, \beta + \sum_{i=1}^n y_i^2 / 2)$$



# Metropolis-Hastings Algorithm

- Allows for asymmetric proposal densities
- We want to draw sample from a density  $p(\theta) = f(\theta)/K$
- Let  $q(x | y)$  denote the proposal density
- Calculate the ratio  $r = \frac{f(\theta^*)q(\theta_{i-1} | \theta^*)}{f(\theta_{i-1})q(\theta^* | \theta_{i-1})}$
- Useful if  $f$  is asymmetric
- Reduces to Metropolis algorithm if  $q$  is symmetric

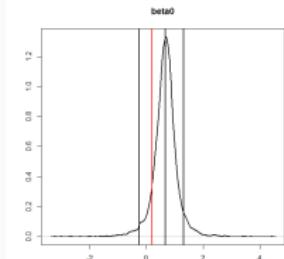
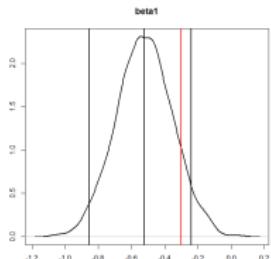
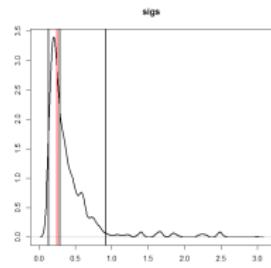
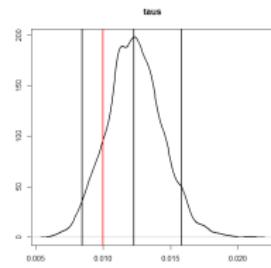
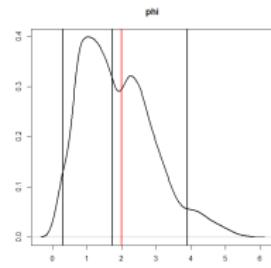
## Metropolis within Gibbs

- For the marginalized model, doing MH for the entire vector  $(\beta', \sigma^2, \tau^2, \phi)'$  may be slow if  $p$  is large
- Also,  $\beta$  has nice normal full conditionals
- One can use a **Metropolis Random Walk (RW) step** for the univariate full conditional target densities inside the Gibbs sampler
- Example: MCMC steps for the marginalized model:
  - (a) Gibbs for  $\beta$ :  $\beta^{(j)} \sim N((X'X)^{-1}X'y, \tau^{2(j-1)}(X'X)^{-1})$
  - (b) RW for  $\phi$  from target density  
$$N(y | X\beta^{(j)}, \sigma^{2(j-1)}R(\phi) + \tau^{2(j-1)}I) \times p(\phi)$$
  - (c) RW for  $\sigma^2$  from  $N(y | X\beta^{(j)}, \sigma^2R(\phi^{(j)}) + \tau^{2(j-1)}I) \times p(\sigma^2)$
  - (d) RW for  $\tau^2$  from target density  
$$N(y | X\beta^{(j)}, \sigma^{2(j)}R(\phi^{(j)}) + \tau^2I) \times p(\phi)$$

## *Nimble* package

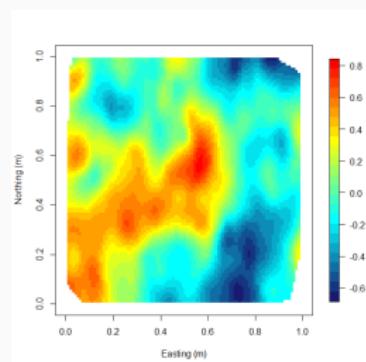
- <https://r-nimble.org/>
- Implements the MCMC for you
- You only need to specify the model and initialize the MCMC !
- We run the MCMC for the marginalized model for dataset 3 in Nimble

# Parameter posteriors

 $\beta_0$  $\beta_1$  $\sigma^2$  $\tau^2$  $\phi$

## Recovering $w$

- The marginalized model integrates out the  $w$ 's
- We can recover them after the MCMC
- $w | y, \beta, \sigma^2, \tau^2, \phi \sim N(V_w(y - X\beta)/\tau^2, V_w)$  where  
 $V_w = (I/\tau^2 + R(\phi)^{-1}/\sigma^2)^{-1}$
- Use composition sampling



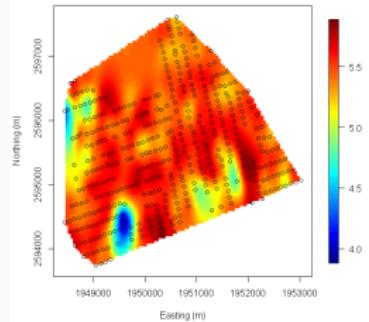
**Figure:**  $w(s) | y$

## Predictions for the marginalized model

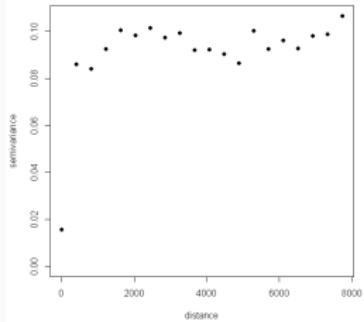
- Two ways to do predict  $\tilde{y}(s) | y$  using composition sampling
- If you have already recovered  $w$ 
  - Similar to the unmarginalized model
  - Generate  $w(s_0) | w, \text{params}$  and then  $\tilde{y}(s_0) | w(s_0), \text{params}$
- Direct approach (not requiring samples of  $w$ ):
  - $c(s_0) = \text{cov}(w(s_0), w)$  and  $\Sigma = \sigma^2 R(\phi) + \tau^2 I$
  - Generate samples of  $\tilde{y}(s_0) | y, \text{params} \sim N(x(s_0)' \beta + c(s_0)' \Sigma^{-1} (y - X\beta), \sigma^2 + \tau^2 - c(s_0)' \Sigma^{-1} c(s_0))$

# BEF data analysis in spBayes

- Dataset available in spBayes on long-term research studies on the Bartlett Experimental Forest, Bartlett, NH
- Forest inventory data for 437 locations
- Variables include species specific basal area and biomass; inventory plot coordinates; slope; elevation; and tasseled cap brightness (TC1), greenness (TC2), and wetness (TC3) components from spring, summer, and fall 2002 Landsat images



log biomass



Variogram

## MCMC free Bayesian inference

- The marginalized model can be reparametrized as:  
 $N(y, X\beta, \sigma^2(R(\phi) + \alpha I))$  where  $\alpha = \tau^2 / \sigma^2$
- If  $\phi$  and  $\alpha$  is fixed, we can do **exact conjugate sampling**
- bayesGeostatExact*** does that
- Fixed values of  $\phi$  and  $\alpha$  can be chosen from the variogram

	2.5%	25%	50%	75%	97.5%
(Intercept)	-0.624	0.267	0.728	1.182	2.079
Elevation	0.000	0.001	0.001	0.001	0.001
Slope	-0.017	-0.013	-0.011	-0.008	-0.004
Brightness	-0.001	0.006	0.010	0.013	0.021
Greenness	0.000	0.004	0.007	0.009	0.014
Wetness	0.015	0.021	0.024	0.028	0.034
$\sigma^2$	0.072	0.079	0.083	0.087	0.094
$\tau^2$	0.014	0.016	0.016	0.017	0.019

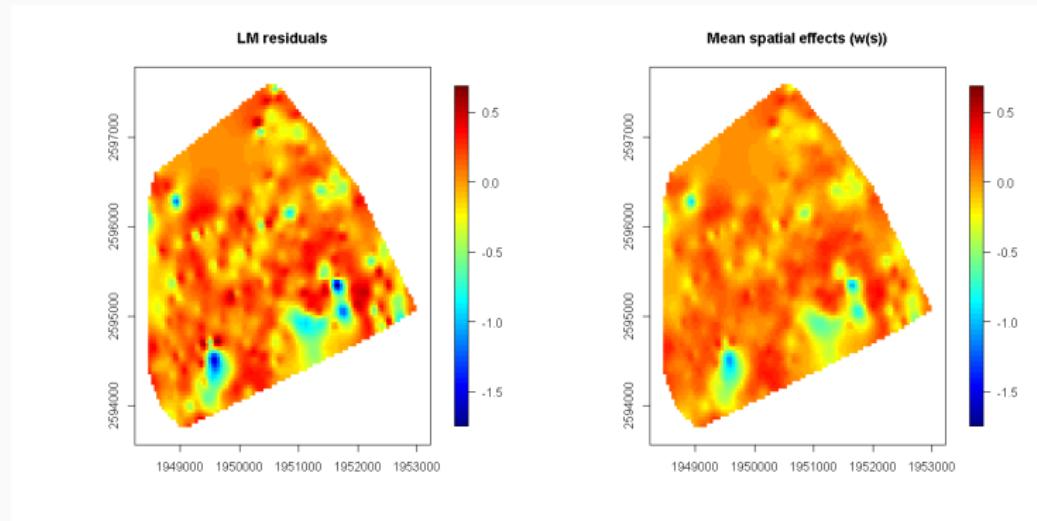
# Full Bayesian inference

- *spLM* function
- Even marginalizes out  $\beta$  to make the chain only 3 dimensional

	2.5%	25%	50%	75%	97.5%
(Intercept)	-0.253	0.937	1.586	2.069	3.189
Elevation	0.000	0.000	0.000	0.001	0.001
Slope	-0.017	-0.011	-0.008	-0.005	0.002
Brightness	-0.005	0.006	0.010	0.015	0.025
Greenness	-0.005	0.003	0.005	0.008	0.014
Wetness	0.007	0.015	0.019	0.023	0.032
$\sigma^2$	0.042	0.074	0.086	0.095	0.108
$\tau^2$	0.005	0.010	0.015	0.030	0.063
$\phi$	0.004	0.008	0.010	0.012	0.016

## Recovery of $w(s)$

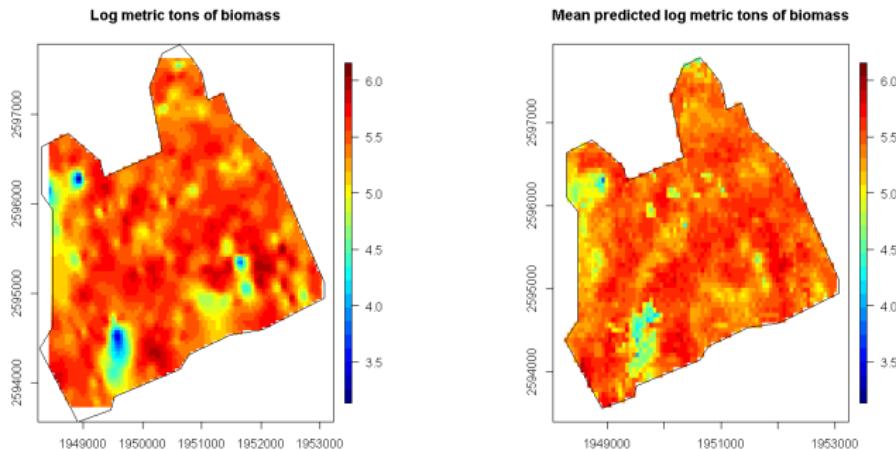
- *spRecover* function recovers both  $\beta$  and  $w$



- *spDiag* calculates the DIC after recovery of  $w$

# Kriging

- *spPredict* function

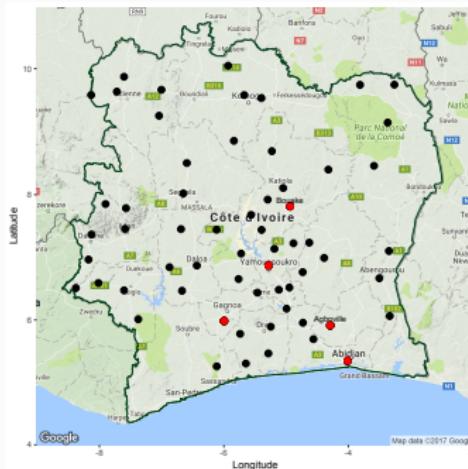


## spBayes vs Nimble

- Nimble is a general package for running MCMC
- spBayes specializes in spatial GP regression and should be the preferred choice if the spatial linear model is the suitable choice for the data analysis
- However, Nimble is a great tool for more complex Bayesian models involving spatial data

# A hierarchical modeling example

- Goal: Prediction of MSM population size at 61 regions of Côte d'Ivoire
- Data on MSM population (as proportion of the relevant male population) available for **only 5 cities** (red dots)
- Need to predict MSM population size at the **remaining 56 regions** (black dots) using a **linear regression** model



## A hierarchical modeling example

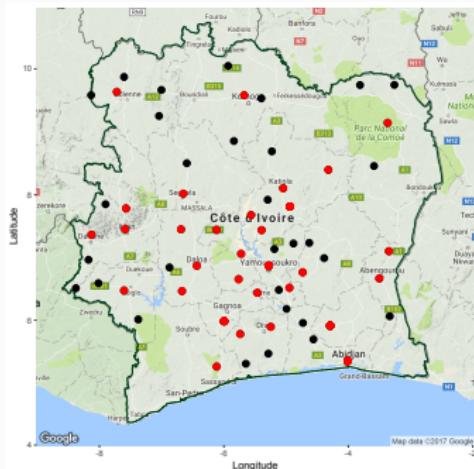
- Goal: Prediction of MSM population size at 61 regions of Côte d'Ivoire
- Data on MSM population (as proportion of the relevant male population) available for only 5 cities (red dots)
- Need to predict MSM population size at the remaining 56 regions (black dots) using a linear regression model
- Multiple datapoints available for each of the 5 cities (from multiple surveys)
- Preliminary analysis suggests important covariates are log of male population and HIV prevalence
- Details at *Datta et al., “Bayesian estimation of MSM population in Côte d'Ivoire”, Statistics and Public Policy, 2018 (In press)*.

## Predicting MSM in Côte d'Ivoire

- This seems to be a straight-forward regression model,  
 $\log(\text{MSM \%}) = \beta_0 + \beta_1 \log(\text{male popn.}) + \beta_2 \text{ HIV prev.} + \text{error}$

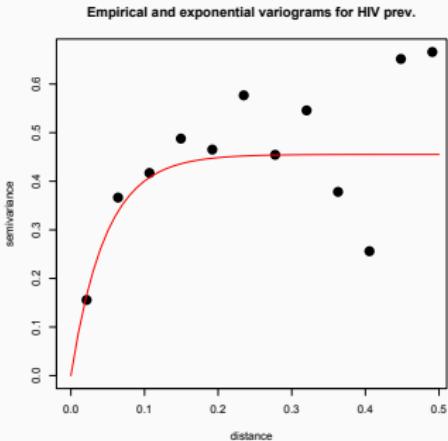
# Predicting MSM in Côte d'Ivoire

- This seems to be a straight-forward regression model,  
 $\log(\text{ MSM \%}) = \beta_0 + \beta_1 \log(\text{male popn.}) + \beta_2 \text{ HIV prev.} + \text{error}$
- HIV prevalence is **missing** in **nearly 50%** of the regions where we want to predict



# Spatial model for HIV prevalence

- Empirical variogram of HIV prevalence suggests spatial dependence



- We can use kriging to impute missing HIV prevalence
- Leave-one-out cross validation suggests kriging offers 20% improved predictive accuracy than simple mean imputation for HIV prevalence

## Hierarchical Bayesian model

- $y_j(s_i)$  is the estimate of population proportion for the  $i^{th}$  region based on the  $j^{th}$  survey
- $N(s_i)$  is male population,  $H(s_i)$  is HIV prevalence

$$\prod_{i=1}^5 \prod_j N(y_j(s_i) | \beta_0 + \beta_1 \log\{N(s_i)\} + \beta_2 H(s_i), \tau^2 w_{ij}) \times$$

← Regression model

$$N(H(S) | \mu_1, \Sigma(\sigma^2, \phi)) \times$$

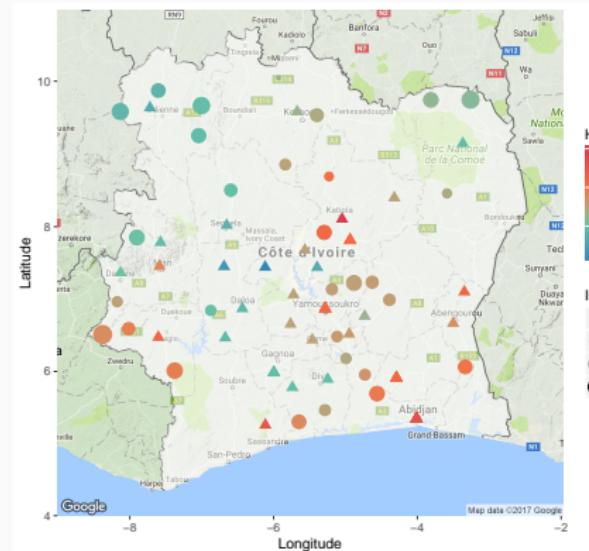
← Spatial model for HIV prevalence

$$N(\beta | 0, 10^6 I) \times N(\mu | 0, 10^6) \times \text{Unif}(\phi | 0, 10) \times$$
$$\text{Gamma}(1/\tau^2 | 0.01, 0.01) \times \text{Gamma}(1/\sigma^2 | 2, 1)$$

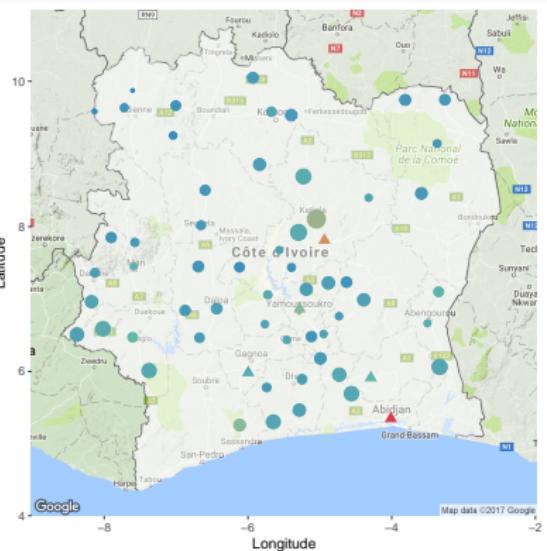
← Priors

- We use a mean constant mean parameter in the GP model for HIV prevalence
- spBayes cannot implement hierarchical models like this
- We use Nimble to run the MCMC

# Predictions and uncertainty estimates



HIV



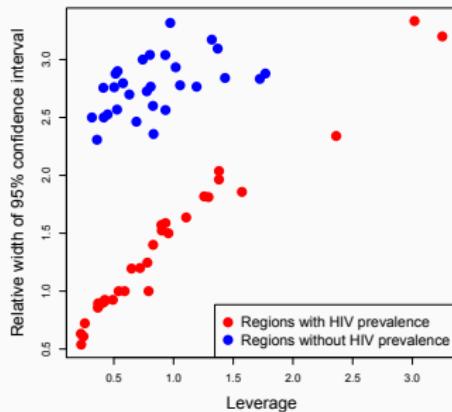
Population size

## Value of a fully Bayesian model

- We can simply use a two step model where we first use kriging to predict HIV and use the predicted HIV in the regression predictions.
- Why do we need the fully Bayesian model that does everything together?

# Value of a fully Bayesian model

- We can simply use a two step model where we first use kriging to predict HIV and use the predicted HIV in the regression predictions.
- Why do we need the fully Bayesian model that does everything together?



- Proper uncertainty quantification: Using the Bayesian model, regions with predicted HIV has higher uncertainty

# Summary

- Convergence diagnostics
- Model comparison using Bayesian output (DIC)
- MH algorithm
- Writing your own MCMC
- Using Nimble package to run the MCMC
- Spatial predictions from Bayesian output
- MCMC-free and fully Bayesian spatial analysis using spBayes package

## References

- Expository article on Gibbs sampler: Casella, G. and George, E.I. (1992), Explaining the Gibbs Sampler, *The American Statistician*, 46, 167-174.
- Expository article on MH algorithm: Chib, S. and Greenberg, E. (1995), Understanding the Metropolis-Hastings Algorithm, *The American Statistician*, 49, 327-335.
- DIC Spiegelhalter, D. J., and Best, N. G., and Carlin, B. P., and van der Linde, A. (2002). *Bayesian measures of model complexity and fit* Journal of the Royal Statistical Society, Series B. 64 (4), 583-63
- Great slides on convergence diagnostics of Markov Chains  
[http://patricklam.org/teaching/convergence\\_print.pdf](http://patricklam.org/teaching/convergence_print.pdf)
- Gelfand, A. E. , and Smith. A. F. M. (1990). *Sampling-Based Approaches to Calculating Marginal Densities*. *Journal of the American Statistical Association*, 85(410), 398–409.
- Liu, J. S., Wong, W. H., and Kong, A. (1994). *Covariance structure of the gibbs sampler with applications to the comparisons of estimators and augmentation schemes*. *Biometrika*, 81(1), 27–40.