Some alternatives to {MC,S,}MC

Håvard Rue
Norwegian University of Science and Technology
Trondheim, Norway

March 18, 2009

Some alternatives to {MC,S,}MC

- {MC,S,}MC is not the only computational device for doing Bayesian analysis, but its the only *general* device.
- Generality has it costs; for specific classes of models, we can do "better"
  - Provide faster answers to commonly asked questions
  - Highly specialised approaches which is often much more involved
- These alternative approaches are not to be seen as competitors to the general algorithms, but as alternatives

Some alternatives to {MC,S,}MC

- {MC,S,}MC is not the only computational device for doing Bayesian analysis, but its the only *general* device.
- Generality has it costs; for specific classes of models, we can do "better"
  - Provide faster answers to commonly asked questions
  - Highly specialised approaches which is often much more involved
- These alternative approaches are not to be seen as competitors to the general algorithms, but as alternatives

Some alternatives to {MC,S,}MC

- {MC,S,}MC is not the only computational device for doing Bayesian analysis, but its the only *general* device.
- Generality has it costs; for specific classes of models, we can do "better"
  - Provide faster answers to commonly asked questions
  - Highly specialised approaches which is often much more involved
- These alternative approaches are not to be seen as competitors to the general algorithms, but as alternatives

Classes of models where there are alternatives

- Latent Gaussian Models (LGM)
- Change-point Models
- Latent binary Markov Random Fields models

(I do not cover Variational-Bayes.)

Overview of this talk

- LGM and INLA
- LGM in practice using R
- Some newer results
  - LGM and Survival models
  - Geostatistics and GMRFs
- Latent Binary MRFs models
- Change-point models
- Parallel computing

Latent Gaussian models

Stage 1 Observed data $\mathbf{y} = (y_i)$,

$$y_i \mid \mathbf{x}, \boldsymbol{\theta} \sim \pi(y_i | x_i, \boldsymbol{\theta})$$

Stage 2 Latent Gaussian field

$$\mathbf{x} \mid \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{Q}(\boldsymbol{\theta})^{-1}), \qquad \mathbf{A}\mathbf{x} = \mathbf{0}$$

Stage 3 Priors for the hyperparameters

$$\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta})$$

Unify many of the most used models in statistics

Latent Gaussian models

Stage 1 Observed data $\mathbf{y} = (y_i)$,

$$y_i \mid \mathbf{x}, \boldsymbol{\theta} \sim \pi(y_i | x_i, \boldsymbol{\theta})$$

Stage 2 Latent Gaussian field

$$\mathbf{x} \mid \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{Q}(\boldsymbol{\theta})^{-1}), \qquad \mathbf{A}\mathbf{x} = \mathbf{0}$$

Stage 3 Priors for the hyperparameters

$$\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta})$$

Unify many of the most used models in statistics

Latent Gaussian models

Stage 1 Observed data $\mathbf{y} = (y_i)$,

$$y_i \mid \mathbf{x}, \boldsymbol{\theta} \sim \pi(y_i | x_i, \boldsymbol{\theta})$$

Stage 2 Latent Gaussian field

$$\mathbf{x} \mid \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{Q}(\boldsymbol{\theta})^{-1}), \qquad \mathbf{Ax} = \mathbf{0}$$

Stage 3 Priors for the hyperparameters

$$\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta})$$

Unify many of the most used models in statistics

Latent Gaussian models

Stage 1 Observed data $\mathbf{y} = (y_i)$,

$$y_i \mid \mathbf{x}, \boldsymbol{\theta} \sim \pi(y_i | x_i, \boldsymbol{\theta})$$

Stage 2 Latent Gaussian field

$$\mathbf{x} \mid \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{Q}(\boldsymbol{\theta})^{-1}), \qquad \mathbf{A}\mathbf{x} = \mathbf{0}$$

Stage 3 Priors for the hyperparameters

$$\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta})$$

Unify many of the most used models in statistics

Structured additive regression models

Linear predictor

$$\eta_i = \sum_k \beta_k z_{ki} + \sum_j w_{ji} f_j(z_{ji}) + \epsilon_i$$

- Linear effects of covariates $\{z_{ki}\}$
- Effects of $f_j(\cdot)$
  - Fixed weights $\{w_{ji}\}$
  - Commonly: $f_j(z_{ji}) = f_{j,z_{ji}}$
  - Account for smooth response
    - Temporal or spatially indexed covariates
    - Unstructured terms ("random effects")
  - Depend on some parameters $\boldsymbol{\theta}$

Structured additive regression models

Linear predictor

$$\eta_i = \sum_k \beta_k z_{ki} + \sum_j w_{ji} f_j(z_{ji}) + \epsilon_i$$

- Linear effects of covariates $\{z_{ki}\}$
- Effects of $f_j(\cdot)$
  - Fixed weights $\{w_{ji}\}$
  - Commonly: $f_j(z_{ji}) = f_{j,z_{ji}}$
  - Account for smooth response
    - Temporal or spatially indexed covariates
    - Unstructured terms ("random effects")
  - Depend on some parameters $\boldsymbol{\theta}$

Structured additive regression models

Linear predictor

$$\eta_i = \sum_k \beta_k z_{ki} + \sum_j w_{ji} f_j(z_{ji}) + \epsilon_i$$

- Linear effects of covariates $\{z_{ki}\}$
- Effects of $f_j(\cdot)$
  - Fixed weights $\{w_{ji}\}$
  - Commonly: $f_j(z_{ji}) = f_{j,z_{ji}}$
  - Account for smooth response
    - Temporal or spatially indexed covariates
    - Unstructured terms ("random effects")
  - Depend on some parameters $\boldsymbol{\theta}$

Structured additive regression models

Linear predictor

$$\eta_i = \sum_k \beta_k z_{ki} + \sum_j w_{ji} f_j(z_{ji}) + \epsilon_i$$

- Linear effects of covariates $\{z_{ki}\}$
- Effects of $f_j(\cdot)$
  - Fixed weights $\{w_{ji}\}$
  - Commonly: $f_j(z_{ji}) = f_{j,z_{ji}}$
  - Account for smooth response
    - Temporal or spatially indexed covariates
    - Unstructured terms ("random effects")
  - Depend on some parameters $\boldsymbol{\theta}$

Structured additive regression models

Linear predictor

$$\eta_i = \sum_k \beta_k z_{ki} + \sum_j w_{ji} f_j(z_{ji}) + \epsilon_i$$

- Linear effects of covariates $\{z_{ki}\}$
- Effects of $f_j(\cdot)$
  - Fixed weights $\{w_{ji}\}$
  - Commonly: $f_j(z_{ji}) = f_{j,z_{ji}}$
  - Account for smooth response
    - Temporal or spatially indexed covariates
    - Unstructured terms ("random effects")
  - Depend on some parameters $\boldsymbol{\theta}$

Structured additive regression models

Linear predictor

$$\eta_i = \sum_k \beta_k z_{ki} + \sum_j w_{ji} f_j(z_{ji}) + \epsilon_i$$

- Linear effects of covariates $\{z_{ki}\}$
- Effects of $f_j(\cdot)$
  - Fixed weights $\{w_{ji}\}$
  - Commonly: $f_j(z_{ji}) = f_{j,z_{ji}}$
  - Account for smooth response
    - Temporal or spatially indexed covariates
    - Unstructured terms ("random effects")
  - Depend on some parameters $\boldsymbol{\theta}$

Structured additive regression models

Linear predictor

$$\eta_i = \sum_k \beta_k z_{ki} + \sum_j w_{ji} f_j(z_{ji}) + \epsilon_i$$

- Linear effects of covariates $\{z_{ki}\}$
- Effects of $f_j(\cdot)$
  - Fixed weights $\{w_{ji}\}$
  - Commonly: $f_j(z_{ji}) = f_{j,z_{ji}}$
  - Account for smooth response
    - Temporal or spatially indexed covariates
    - Unstructured terms ("random effects")
  - Depend on some parameters $\boldsymbol{\theta}$

Structured additive regression models

Linear predictor

$$\eta_i = \sum_k \beta_k z_{ki} + \sum_j w_{ji} f_j(z_{ji}) + \epsilon_i$$

- Linear effects of covariates $\{z_{ki}\}$
- Effects of $f_j(\cdot)$
  - Fixed weights $\{w_{ji}\}$
  - Commonly: $f_j(z_{ji}) = f_{j,z_{ji}}$
  - Account for smooth response
    - Temporal or spatially indexed covariates
    - Unstructured terms ("random effects")
  - Depend on some parameters $\boldsymbol{\theta}$

Structured additive regression models

$$\eta_i = \sum_k \beta_k z_{ki} + \sum_j w_{ji} f_j(z_{ji}) + \epsilon_i$$

- Observations $\mathbf{y}$

$$\pi(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}) = \prod_i \pi(y_i \mid \eta_i, \boldsymbol{\theta})$$

  from an (f.ex) exponential family with mean $\mu_i = g^{-1}(\eta_i)$.
- *Latent Gaussian model* if

$$\mathbf{x} = (\{\beta_k\}, \{f_{ji}\}, \{\eta_i\}) \mid \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{Q}(\boldsymbol{\theta})^{-1})$$

Structured additive regression models

$$\eta_i = \sum_k \beta_k z_{ki} + \sum_j w_{ji} f_j(z_{ji}) + \epsilon_i$$

- Observations $\mathbf{y}$

$$\pi(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}) = \prod_i \pi(y_i \mid \eta_i, \boldsymbol{\theta})$$

from an (f.ex) exponential family with mean $\mu_i = g^{-1}(\eta_i)$.

- *Latent Gaussian model* if

$$\mathbf{x} = (\{\beta_k\}, \{f_{ji}\}, \{\eta_i\}) \mid \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{Q}(\boldsymbol{\theta})^{-1})$$

Structured additive regression models

$$\eta_i = \sum_k \beta_k z_{ki} + \sum_j w_{ji} f_j(z_{ji}) + \epsilon_i$$

- Observations $\mathbf{y}$

$$\pi(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}) = \prod_i \pi(y_i \mid \eta_i, \boldsymbol{\theta})$$

  from an (f.ex) exponential family with mean $\mu_i = g^{-1}(\eta_i)$.
- *Latent Gaussian model* if

$$\mathbf{x} = (\{\beta_k\}, \{f_{ji}\}, \{\eta_i\}) \mid \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{Q}(\boldsymbol{\theta})^{-1})$$

Examples

- Dynamic linear models
- Stochastic volatility
- Generalised linear (mixed) models
- Generalised additive (mixed) models
- Spline smoothing
- Semiparametric regression
- Space-varying (semiparametric) regression models
- Disease mapping
- Log-Gaussian Cox-processes
- Model-based geostatistics (*)
- Spatio-temporal models

Examples

- Dynamic linear models
- Stochastic volatility
- Generalised linear (mixed) models
- Generalised additive (mixed) models
- Spline smoothing
- Semiparametric regression
- Space-varying (semiparametric) regression models
- Disease mapping
- Log-Gaussian Cox-processes
- Model-based geostatistics (*)
- Spatio-temporal models

Examples

- Dynamic linear models
- Stochastic volatility
- Generalised linear (mixed) models
- Generalised additive (mixed) models
- Spline smoothing
- Semiparametric regression
- Space-varying (semiparametric) regression models
- Disease mapping
- Log-Gaussian Cox-processes
- Model-based geostatistics (*)
- Spatio-temporal models

Examples

- Dynamic linear models
- Stochastic volatility
- Generalised linear (mixed) models
- Generalised additive (mixed) models
- Spline smoothing
- Semiparametric regression
- Space-varying (semiparametric) regression models
- Disease mapping
- Log-Gaussian Cox-processes
- Model-based geostatistics (*)
- Spatio-temporal models

Examples

- Dynamic linear models
- Stochastic volatility
- Generalised linear (mixed) models
- Generalised additive (mixed) models
- Spline smoothing
- Semiparametric regression
- Space-varying (semiparametric) regression models
- Disease mapping
- Log-Gaussian Cox-processes
- Model-based geostatistics (*)
- Spatio-temporal models

Examples

- Dynamic linear models
- Stochastic volatility
- Generalised linear (mixed) models
- Generalised additive (mixed) models
- Spline smoothing
- Semiparametric regression
- Space-varying (semiparametric) regression models
- Disease mapping
- Log-Gaussian Cox-processes
- Model-based geostatistics (*)
- Spatio-temporal models

Examples

- Dynamic linear models
- Stochastic volatility
- Generalised linear (mixed) models
- Generalised additive (mixed) models
- Spline smoothing
- Semiparametric regression
- Space-varying (semiparametric) regression models
- Disease mapping
- Log-Gaussian Cox-processes
- Model-based geostatistics (*)
- Spatio-temporal models

Examples

- Dynamic linear models
- Stochastic volatility
- Generalised linear (mixed) models
- Generalised additive (mixed) models
- Spline smoothing
- Semiparametric regression
- Space-varying (semiparametric) regression models
- Disease mapping
- Log-Gaussian Cox-processes
- Model-based geostatistics (*)
- Spatio-temporal models

Examples

- Dynamic linear models
- Stochastic volatility
- Generalised linear (mixed) models
- Generalised additive (mixed) models
- Spline smoothing
- Semiparametric regression
- Space-varying (semiparametric) regression models
- Disease mapping
- Log-Gaussian Cox-processes
- Model-based geostatistics (*)
- Spatio-temporal models

Examples

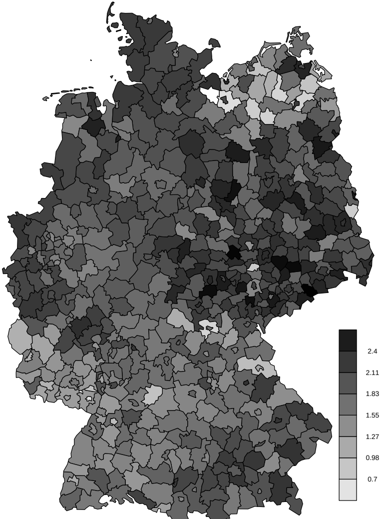
- Dynamic linear models
- Stochastic volatility
- Generalised linear (mixed) models
- Generalised additive (mixed) models
- Spline smoothing
- Semiparametric regression
- Space-varying (semiparametric) regression models
- Disease mapping
- Log-Gaussian Cox-processes
- Model-based geostatistics (*)
- Spatio-temporal models

Examples

- Dynamic linear models
- Stochastic volatility
- Generalised linear (mixed) models
- Generalised additive (mixed) models
- Spline smoothing
- Semiparametric regression
- Space-varying (semiparametric) regression models
- Disease mapping
- Log-Gaussian Cox-processes
- Model-based geostatistics (*)
- Spatio-temporal models

Example: Disease mapping (BYM-model)

- Data $y_i \sim \text{Poisson}(E_i exp(\eta_i))$
- Log-relative risk $\eta_i = u_i + v_i + \boldsymbol{\beta}^T \mathbf{z}_i$
- Structured component $\mathbf{u}$
- Unstructured component $\mathbf{v}$
- Covariates $\mathbf{z}_i$
- $\boldsymbol{\theta}$ are the Log-precisions $\log \kappa_u$ and $\log \kappa_v$



Characteristic features

- Large dimension of the latent Gaussian field: $10^2 - 10^5$
- A lot of conditional independence in the latent Gaussian field
- Few hyperparameters $\boldsymbol{\theta}$: $\dim(\boldsymbol{\theta})$ between 1 and 5
- Non-Gaussian data

Characteristic features

- Large dimension of the latent Gaussian field: $10^2 - 10^5$
- A lot of conditional independence in the latent Gaussian field
- Few hyperparameters $\boldsymbol{\theta}$: $\dim(\boldsymbol{\theta})$ between 1 and 5
- Non-Gaussian data

Characteristic features

- Large dimension of the latent Gaussian field: $10^2 - 10^5$
- A lot of conditional independence in the latent Gaussian field
- Few hyperparameters $\boldsymbol{\theta}$: $\dim(\boldsymbol{\theta})$ between 1 and 5
- Non-Gaussian data

Characteristic features

- Large dimension of the latent Gaussian field: $10^2 - 10^5$
- A lot of conditional independence in the latent Gaussian field
- Few hyperparameters $\boldsymbol{\theta}$: $\dim(\boldsymbol{\theta})$ between 1 and 5
- Non-Gaussian data

Main task

- Compute the posterior marginals for the latent field

$$\pi(x_i \mid \mathbf{y}), \qquad i = 1, \ldots, n$$

- Compute the posterior marginals for the hyperparameters

$$\pi(\theta_j \mid \mathbf{y}), \qquad j = 1, \ldots, \dim(\boldsymbol{\theta})$$

- Today's "standard" approach, is to make use of MCMC
- Main difficulties
  - CPU-time
  - Additive MC-errors

Main task

- Compute the posterior marginals for the latent field

$$\pi(x_i \mid \mathbf{y}), \qquad i = 1, \ldots, n$$

- Compute the posterior marginals for the hyperparameters

$$\pi(\theta_j \mid \mathbf{y}), \qquad j = 1, \ldots, \dim(\boldsymbol{\theta})$$

- Today's "standard" approach, is to make use of MCMC
- Main difficulties
  - CPU-time
  - Additive MC-errors

Main task

- Compute the posterior marginals for the latent field

$$\pi(x_i \mid \mathbf{y}), \qquad i = 1, \ldots, n$$

- Compute the posterior marginals for the hyperparameters

$$\pi(\theta_j \mid \mathbf{y}), \qquad j = 1, \ldots, \dim(\boldsymbol{\theta})$$

- Today's "standard" approach, is to make use of MCMC
- Main difficulties
  - CPU-time
  - Additive MC-errors

Main task

- Compute the posterior marginals for the latent field

$$\pi(x_i \mid \mathbf{y}), \qquad i = 1, \ldots, n$$

- Compute the posterior marginals for the hyperparameters

$$\pi(\theta_j \mid \mathbf{y}), \qquad j = 1, \ldots, \dim(\boldsymbol{\theta})$$

- Today's "standard" approach, is to make use of MCMC
- Main difficulties
  - CPU-time
  - Additive MC-errors

Integrated nested Laplace approximations (INLA) [With N.Chopin and S.Martino]

- Utilise of the latent Gaussian field
  - Laplace approximations
- Utilise the conditional independence properties of the latent Gaussian field
  - Numerical algorithms for sparse matrices
- Utilise small dim(θ)
  - *Integrated Nested* Laplace approximations

Integrated nested Laplace approximations (INLA) [With N.Chopin and S.Martino]

- Utilise of the latent Gaussian field
    - Laplace approximations
- Utilise the conditional independence properties of the latent Gaussian field
    - Numerical algorithms for sparse matrices
- Utilise small dim(θ)
    - *Integrated Nested* Laplace approximations

Integrated nested Laplace approximations (INLA) [With N.Chopin and S.Martino]

- Utilise of the latent Gaussian field
  - Laplace approximations
- Utilise the conditional independence properties of the latent Gaussian field
  - Numerical algorithms for sparse matrices
- Utilise small dim(θ)
  - *Integrated Nested* Laplace approximations

Integrated nested Laplace approximations (INLA) [With N.Chopin and S.Martino]

- Utilise of the latent Gaussian field
    - Laplace approximations
- Utilise the conditional independence properties of the latent Gaussian field
    - Numerical algorithms for sparse matrices
- Utilise small dim($\boldsymbol{\theta}$)
    - *Integrated Nested* Laplace approximations

Integrated nested Laplace approximations (INLA) [With N.Chopin and S.Martino]

- Utilise of the latent Gaussian field
    - Laplace approximations
- Utilise the conditional independence properties of the latent Gaussian field
    - Numerical algorithms for sparse matrices
- Utilise small dim($\boldsymbol{\theta}$)
    - *Integrated Nested* Laplace approximations

Integrated nested Laplace approximations (INLA) [With N.Chopin and S.Martino]

- Utilise of the latent Gaussian field
  - Laplace approximations
- Utilise the conditional independence properties of the latent Gaussian field
  - Numerical algorithms for sparse matrices
- Utilise small dim($\boldsymbol{\theta}$)
  - *Integrated Nested* Laplace approximations

Main results

- HUGE improvement in both speed and accuracy compared to MCMC alternatives
- Relative error
- Practically "exact" results[1]
- Extensions: Marginal likelihood, DIC, Cross-validation, ...

INLA enable us to treat Bayesian latent Gaussian models properly and bring these models from the research communities to the end-users

[1] Can construct counter-examples

Main results

- HUGE improvement in both speed and accuracy compared to MCMC alternatives
- Relative error
- Practically "exact" results[1]
- Extensions: Marginal likelihood, DIC, Cross-validation, ...

INLA enable us to treat Bayesian latent Gaussian models properly and bring these models from the research communities to the end-users

[1] Can construct counter-examples

Main results

- HUGE improvement in both speed and accuracy compared to MCMC alternatives
- Relative error
- Practically "exact" results[1]
- Extensions: Marginal likelihood, DIC, Cross-validation, ...

INLA enable us to treat Bayesian latent Gaussian models properly and bring these models from the research communities to the end-users

[1] Can construct counter-examples

Main results

- HUGE improvement in both speed and accuracy compared to MCMC alternatives
- Relative error
- Practically "exact" results[1]
- Extensions: Marginal likelihood, DIC, Cross-validation, ...

INLA enable us to treat Bayesian latent Gaussian models properly and bring these models from the research communities to the end-users

[1] Can construct counter-examples

Main results

- HUGE improvement in both speed and accuracy compared to MCMC alternatives
- Relative error
- Practically "exact" results[1]
- Extensions: Marginal likelihood, DIC, Cross-validation, ...

INLA enable us to treat Bayesian latent Gaussian models properly and bring these models from the research communities to the end-users

[1] Can construct counter-examples

Main ideas (I)

$$\pi(z) = \frac{\pi(x, z)}{\pi(x|z)} \qquad \text{leading to} \qquad \widetilde{\pi}(z) = \frac{\pi(x, z)}{\widetilde{\pi}(x|z)}\bigg|_{\text{mode}(z)}$$

- When $\widetilde{\pi}(x|z)$ is the Gaussian-approximation, this is the Laplace-approximation
- Want $\pi(x|z)$ to be "almost Gaussian".

Main ideas (I)

$$\pi(z) = \frac{\pi(x, z)}{\pi(x|z)} \qquad \text{leading to} \qquad \widetilde{\pi}(z) = \frac{\pi(x, z)}{\widetilde{\pi}(x|z)}\bigg|_{\mathrm{mode}(z)}$$

- When $\widetilde{\pi}(x|z)$ is the Gaussian-approximation, this is the Laplace-approximation
- Want $\pi(x|z)$ to be "almost Gaussian".

Main ideas (I)

$$\pi(z) = \frac{\pi(x, z)}{\pi(x|z)} \qquad \text{leading to} \qquad \widetilde{\pi}(z) = \frac{\pi(x, z)}{\widetilde{\pi}(x|z)}\Big|_{\text{mode}(z)}$$

- When $\widetilde{\pi}(x|z)$ is the Gaussian-approximation, this is the Laplace-approximation
- Want $\pi(x|z)$ to be “almost Gaussian”.

Main ideas (II)

Posterior

$$\pi(\mathbf{x}, \boldsymbol{\theta} \mid \mathbf{y}) \propto \pi(\boldsymbol{\theta})\, \pi(\mathbf{x} \mid \boldsymbol{\theta}) \prod_{i \in \mathcal{I}} \pi(y_i \mid x_i, \boldsymbol{\theta})$$

Do the integration wrt $\boldsymbol{\theta}$ numerically

Main ideas (II)

Posterior

$$\pi(\mathbf{x}, \boldsymbol{\theta} \mid \mathbf{y}) \propto \pi(\boldsymbol{\theta})\, \pi(\mathbf{x} \mid \boldsymbol{\theta}) \prod_{i \in \mathcal{I}} \pi(y_i \mid x_i, \boldsymbol{\theta})$$

Do the integration wrt $\boldsymbol{\theta}$ numerically

$$\pi(x_i \mid \mathbf{y}) = \int \pi(\boldsymbol{\theta} \mid \mathbf{y})\, \pi(x_i \mid \boldsymbol{\theta}, \mathbf{y})\, d\boldsymbol{\theta}$$

$$\pi(\theta_j \mid \mathbf{y}) = \int \pi(\boldsymbol{\theta} \mid \mathbf{y})\, d\boldsymbol{\theta}_{-j}$$

Main ideas (II)

Posterior

$$\pi(\mathbf{x}, \boldsymbol{\theta} \mid \mathbf{y}) \propto \pi(\boldsymbol{\theta})\, \pi(\mathbf{x} \mid \boldsymbol{\theta}) \prod_{i \in \mathcal{I}} \pi(y_i \mid x_i, \boldsymbol{\theta})$$

Do the integration wrt $\boldsymbol{\theta}$ numerically

$$\widetilde{\pi}(x_i \mid \mathbf{y}) = \int \widetilde{\pi}(\boldsymbol{\theta} \mid \mathbf{y})\, \widetilde{\pi}(x_i \mid \boldsymbol{\theta}, \mathbf{y})\, d\boldsymbol{\theta}$$

$$\widetilde{\pi}(\theta_j \mid \mathbf{y}) = \int \widetilde{\pi}(\boldsymbol{\theta} \mid \mathbf{y})\, d\boldsymbol{\theta}_{-j}$$

Remarks

1. Expect $\widetilde{\pi}(\boldsymbol{\theta}|\mathbf{y})$ to be accurate, since
   - $\mathbf{x}|\boldsymbol{\theta}$ is *a priori* Gaussian
   - Likelihood models are 'well-behaved' so

     $$\pi(\mathbf{x} \mid \boldsymbol{\theta}, \mathbf{y})$$

     is *almost* Gaussian.
2. There are no distributional assumptions on $\boldsymbol{\theta}|\mathbf{y}$
3. Similar remarks are valid to

   $$\widetilde{\pi}(x_i \mid \boldsymbol{\theta}, \mathbf{y})$$

Remarks

1. Expect $\widetilde{\pi}(\boldsymbol{\theta}|\mathbf{y})$ to be accurate, since
   - $\mathbf{x}|\boldsymbol{\theta}$ is *a priori* Gaussian
   - Likelihood models are 'well-behaved' so

     $$\pi(\mathbf{x} \mid \boldsymbol{\theta}, \mathbf{y})$$

     is *almost* Gaussian.
2. There are no distributional assumptions on $\boldsymbol{\theta}|\mathbf{y}$
3. Similar remarks are valid to

   $$\widetilde{\pi}(x_i \mid \boldsymbol{\theta}, \mathbf{y})$$

Remarks

1. Expect $\widetilde{\pi}(\boldsymbol{\theta}|\mathbf{y})$ to be accurate, since
   - $\mathbf{x}|\boldsymbol{\theta}$ is *a priori* Gaussian
   - Likelihood models are 'well-behaved' so

     $$\pi(\mathbf{x} \mid \boldsymbol{\theta}, \mathbf{y})$$

     is *almost* Gaussian.
2. There are no distributional assumptions on $\boldsymbol{\theta}|\mathbf{y}$
3. Similar remarks are valid to

   $$\widetilde{\pi}(x_i \mid \boldsymbol{\theta}, \mathbf{y})$$

Remarks

1. Expect $\widetilde{\pi}(\boldsymbol{\theta}|\mathbf{y})$ to be accurate, since
   - $\mathbf{x}|\boldsymbol{\theta}$ is *a priori* Gaussian
   - Likelihood models are 'well-behaved' so

     $$\pi(\mathbf{x} \mid \boldsymbol{\theta}, \mathbf{y})$$

     is *almost* Gaussian.
2. There are no distributional assumptions on $\boldsymbol{\theta}|\mathbf{y}$
3. Similar remarks are valid to

   $$\widetilde{\pi}(x_i \mid \boldsymbol{\theta}, \mathbf{y})$$

Remarks

1. Expect $\widetilde{\pi}(\boldsymbol{\theta}|\mathbf{y})$ to be accurate, since
   - $\mathbf{x}|\boldsymbol{\theta}$ is *a priori* Gaussian
   - Likelihood models are 'well-behaved' so

     $$\pi(\mathbf{x} \mid \boldsymbol{\theta}, \mathbf{y})$$

     is *almost* Gaussian.
2. There are no distributional assumptions on $\boldsymbol{\theta}|\mathbf{y}$
3. Similar remarks are valid to

   $$\widetilde{\pi}(x_i \mid \boldsymbol{\theta}, \mathbf{y})$$

Computational issues

- Our approach in its "raw" form is not computational feasible
- Main issue is the large dimension, n, of the latent field
- Various strategies/tricks are required for obtaining a practical good solution

Computational issues

- Our approach in its "raw" form is not computational feasible
- Main issue is the large dimension, n, of the latent field
- Various strategies/tricks are required for obtaining a practical good solution

Computational issues

- Our approach in its “raw” form is not computational feasible
- Main issue is the large dimension, n, of the latent field
- Various strategies/tricks are required for obtaining a practical good solution

Gaussian Markov random fields

Make use of the conditional independence properties in the latent field

$$x_i \perp x_j \mid \mathbf{x}_{-ij} \quad \iff \quad Q_{ij} = 0$$

where $\mathbf{Q}$ is the precision matrix (inverse covariance)

Use numerical methods for sparse matrices!

Gaussian Markov random fields

Make use of the conditional independence properties in the latent field

$$x_i \perp x_j \mid \mathbf{x}_{-ij} \quad \Longleftrightarrow \quad Q_{ij} = 0$$

where $\mathbf{Q}$ is the precision matrix (inverse covariance)

Use numerical methods for sparse matrices!

Simplified Laplace Approximation

Expand the Laplace approximation of $\pi(x_i|\boldsymbol{\theta}, \mathbf{y})$:

$$\log \widetilde{\pi}(x_i|\boldsymbol{\theta}, \mathbf{y}) = -\frac{1}{2}x_i^2 + b_i x_i + \frac{1}{6}d_i \; x_i^3 + \cdots$$

Remarks

- Correct the Gaussian approximation for error in shift and skewness through b_i and d_i
- Fit a skew-Normal density

$$2\phi(x)\Phi(ax)$$

- Computational fast
- Sufficient accurate for most applications

Simplified Laplace Approximation

Expand the Laplace approximation of $\pi(x_i|\boldsymbol{\theta}, \mathbf{y})$:

$$\log \widetilde{\pi}(x_i|\boldsymbol{\theta}, \mathbf{y}) = -\frac{1}{2}x_i^2 + b_i x_i + \frac{1}{6}d_i\, x_i^3 + \cdots$$

Remarks

- Correct the Gaussian approximation for error in shift and skewness through b_i and d_i
- Fit a skew-Normal density

$$2\phi(x)\Phi(ax)$$

- Computational fast
- Sufficient accurate for most applications

Simplified Laplace Approximation

Expand the Laplace approximation of $\pi(x_i|\boldsymbol{\theta}, \mathbf{y})$:

$$\log \widetilde{\pi}(x_i|\boldsymbol{\theta}, \mathbf{y}) = -\frac{1}{2}x_i^2 + b_i x_i + \frac{1}{6}d_i\; x_i^3 + \cdots$$

Remarks

- Correct the Gaussian approximation for error in shift and skewness through b_i and d_i
- Fit a skew-Normal density

$$2\phi(x)\Phi(ax)$$

- Computational fast
- Sufficient accurate for most applications

Simplified Laplace Approximation

Expand the Laplace approximation of $\pi(x_i|\boldsymbol{\theta}, \mathbf{y})$:

$$\log \widetilde{\pi}(x_i|\boldsymbol{\theta}, \mathbf{y}) = -\frac{1}{2}x_i^2 + b_i x_i + \frac{1}{6}d_i\, x_i^3 + \cdots$$

Remarks

- Correct the Gaussian approximation for error in shift and skewness through b_i and d_i
- Fit a skew-Normal density

$$2\phi(x)\Phi(ax)$$

- Computational fast
- Sufficient accurate for most applications

Simplified Laplace Approximation

Expand the Laplace approximation of $\pi(x_i|\boldsymbol{\theta}, \mathbf{y})$:

$$\log \widetilde{\pi}(x_i|\boldsymbol{\theta}, \mathbf{y}) = -\frac{1}{2}x_i^2 + b_i x_i + \frac{1}{6}d_i\, x_i^3 + \cdots$$

Remarks

- Correct the Gaussian approximation for error in shift and skewness through b_i and d_i
- Fit a skew-Normal density

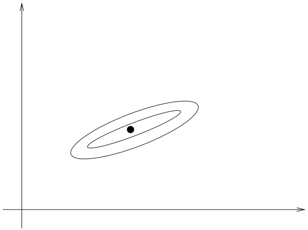
$$2\phi(x)\Phi(ax)$$

- Computational fast
- Sufficient accurate for most applications

The integrated nested Laplace approximation (INLA) I

Step I Explore $\widetilde{\pi}(\boldsymbol{\theta}|\mathbf{y})$

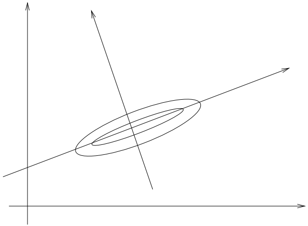
- Locate the mode
- Use the Hessian to construct new variables
- Grid-search



The integrated nested Laplace approximation (INLA) I

Step I Explore $\widetilde{\pi}(\boldsymbol{\theta}|\mathbf{y})$

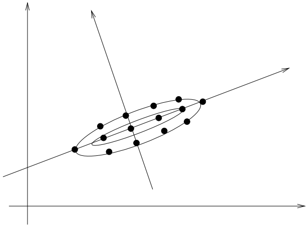
- Locate the mode
- Use the Hessian to construct new variables
- Grid-search



The integrated nested Laplace approximation (INLA) I

Step I Explore $\widetilde{\pi}(\boldsymbol{\theta}|\mathbf{y})$

- Locate the mode
- Use the Hessian to construct new variables
- Grid-search



The integrated nested Laplace approximation (INLA) II

Step II For each $\boldsymbol{\theta}_j$

- For each i, compute the (simplified) Laplace approximation for x_i

The integrated nested Laplace approximation (INLA) III

Step III Sum out $\boldsymbol{\theta}_j$

- For each i, sum out $\boldsymbol{\theta}$

$$\widetilde{\pi}(x_i \mid \mathbf{y}) \propto \sum_j \widetilde{\pi}(x_i \mid \mathbf{y}, \boldsymbol{\theta}_j) \times \widetilde{\pi}(\boldsymbol{\theta}_j \mid \mathbf{y})$$

- Build a log-spline corrected Gaussian

$$\mathcal{N}(x_i;\ \mu_i, \sigma_i^2) \times \exp(\text{spline})$$

to represent $\widetilde{\pi}(x_i \mid \mathbf{y})$.

The integrated nested Laplace approximation (INLA) III

Step III Sum out $\boldsymbol{\theta}_j$

- For each i, sum out $\boldsymbol{\theta}$

$$\widetilde{\pi}(x_i \mid \mathbf{y}) \propto \sum_j \widetilde{\pi}(x_i \mid \mathbf{y}, \boldsymbol{\theta}_j) \times \widetilde{\pi}(\boldsymbol{\theta}_j \mid \mathbf{y})$$

- Build a log-spline corrected Gaussian

$$\mathcal{N}(x_i;\ \mu_i, \sigma_i^2) \times \exp(\text{spline})$$

to represent $\widetilde{\pi}(x_i \mid \mathbf{y})$.

How can we assess the errors in the approximations?

Important, but asymptotic arguments are difficult:

$$\dim(\mathbf{y}) = \mathcal{O}(n) \quad \text{and} \quad \dim(\mathbf{x}) = \mathcal{O}(n)$$

Errors in the approximations of $\pi(x_i|\mathbf{y})$

Compare a sequence of improved approximations

1. Gaussian approximation
2. Simplified Laplace
3. Laplace

Compute the full Laplace-approximation for $\pi(x_i|\mathbf{y}, \boldsymbol{\theta}_j)$ only if the Gaussian and the Simplified Laplace approximation disagree.

Errors in the approximations of $\pi(x_i|\mathbf{y})$

Compare a sequence of improved approximations

1. Gaussian approximation
2. Simplified Laplace
3. Laplace

Compute the full Laplace-approximation for $\pi(x_i|\mathbf{y}, \boldsymbol{\theta}_j)$ only if the Gaussian and the Simplified Laplace approximation disagree.

Overall check: Equivalent number of replicates

Tool 3: Estimate the "effective" number of parameters

- From the *Deviance Information Criteria*:

$$p_{\mathrm{D}}(\boldsymbol{\theta}) \approx n - \mathrm{trace}\left(\mathbf{Q}_{\mathrm{prior}}(\boldsymbol{\theta})\, \mathbf{Q}_{\mathrm{post.}}(\boldsymbol{\theta})^{-1}\right)$$

- Compare with the number of observations:

$$\#\mathrm{observations}/p_{\mathrm{D}}(\boldsymbol{\theta})$$

  high ratio is good

- Theoretical justification

Overall check: Equivalent number of replicates

Tool 3: Estimate the "effective" number of parameters

- From the *Deviance Information Criteria*:

$$p_{\mathrm{D}}(\boldsymbol{\theta}) \approx n - \mathrm{trace}\left(\mathbf{Q}_{\mathrm{prior}}(\boldsymbol{\theta})\, \mathbf{Q}_{\mathrm{post.}}(\boldsymbol{\theta})^{-1}\right)$$

- Compare with the number of observations:

$$\#\mathrm{observations}/p_{\mathrm{D}}(\boldsymbol{\theta})$$

  high ratio is good

- Theoretical justification

Overall check: Equivalent number of replicates

Tool 3: Estimate the "effective" number of parameters

- From the *Deviance Information Criteria*:

$$p_{\mathrm{D}}(\boldsymbol{\theta}) \approx n - \mathrm{trace}\left(\mathbf{Q}_{\mathrm{prior}}(\boldsymbol{\theta})\, \mathbf{Q}_{\mathrm{post.}}(\boldsymbol{\theta})^{-1}\right)$$

- Compare with the number of observations:

$$\#\mathrm{observations}/p_{\mathrm{D}}(\boldsymbol{\theta})$$

  high ratio is good

- Theoretical justification

Marginal likelihood

Marginal likelihood is the normalising constant for $\pi(\boldsymbol{\theta}|\mathbf{y})$

Deviance Information Criteria

$$D(\mathbf{x};\boldsymbol{\theta}) = -2\sum_i \log(y_i \mid x_i, \boldsymbol{\theta})$$

$$DIC = 2 \times \mathrm{Mean}\left(D(\mathbf{x};\boldsymbol{\theta})\right) - D(\mathrm{Mean}(\mathbf{x});\boldsymbol{\theta}^*)$$

Cross-validation

- Based on

$$\pi(x_i|\mathbf{y}_{-i},\boldsymbol{\theta}) \propto \frac{\pi(x_i|\mathbf{y},\boldsymbol{\theta})}{\pi(y_i|x_i,\boldsymbol{\theta})}$$

  we can compute

$$\pi(y_i \mid \mathbf{y}_{-i})$$

- Similar with $\pi(\boldsymbol{\theta}|\mathbf{y}_{-i})$
- Keep the integration points $\{\boldsymbol{\theta}_j\}$ fixed.
- Detect "surprising" observations:

$$\mathrm{PIT}_i = \mathrm{Prob}(y_i^{\mathrm{new}} \leq y_i \mid \mathbf{y}_{-i})$$

- To some extent, validate the model looking at $\{\mathrm{PIT}_i\}$

Cross-validation

- Based on

$$\pi(x_i|\mathbf{y}_{-i}, \boldsymbol{\theta}) \propto \frac{\pi(x_i|\mathbf{y}, \boldsymbol{\theta})}{\pi(y_i|x_i, \boldsymbol{\theta})}$$

  we can compute

$$\pi(y_i \mid \mathbf{y}_{-i})$$

- Similar with $\pi(\boldsymbol{\theta}|\mathbf{y}_{-i})$
- Keep the integration points $\{\boldsymbol{\theta}_j\}$ fixed.
- Detect "surprising" observations:

$$\text{PIT}_i = \text{Prob}(y_i^{\text{new}} \leq y_i \mid \mathbf{y}_{-i})$$

- To some extent, validate the model looking at $\{\text{PIT}_i\}$

Cross-validation

- Based on

$$\pi(x_i|\mathbf{y}_{-i}, \boldsymbol{\theta}) \propto \frac{\pi(x_i|\mathbf{y}, \boldsymbol{\theta})}{\pi(y_i|x_i, \boldsymbol{\theta})}$$

  we can compute

$$\pi(y_i \mid \mathbf{y}_{-i})$$

- Similar with $\pi(\boldsymbol{\theta}|\mathbf{y}_{-i})$
- Keep the integration points $\{\boldsymbol{\theta}_j\}$ fixed.
- Detect "surprising" observations:

$$\mathrm{PIT}_i = \mathrm{Prob}(y_i^{\mathrm{new}} \leq y_i \mid \mathbf{y}_{-i})$$

- To some extent, validate the model looking at $\{\mathrm{PIT}_i\}$

Cross-validation

- Based on
$$\pi(x_i|\mathbf{y}_{-i},\boldsymbol{\theta}) \propto \frac{\pi(x_i|\mathbf{y},\boldsymbol{\theta})}{\pi(y_i|x_i,\boldsymbol{\theta})}$$
we can compute
$$\pi(y_i \mid \mathbf{y}_{-i})$$
- Similar with $\pi(\boldsymbol{\theta}|\mathbf{y}_{-i})$
- Keep the integration points $\{\boldsymbol{\theta}_j\}$ fixed.
- Detect "surprising" observations:
$$\mathrm{PIT}_i = \mathrm{Prob}(y_i^{\text{new}} \leq y_i \mid \mathbf{y}_{-i})$$
- To some extent, validate the model looking at $\{\mathrm{PIT}_i\}$

Cross-validation

- Based on

$$\pi(x_i|\mathbf{y}_{-i},\boldsymbol{\theta}) \propto \frac{\pi(x_i|\mathbf{y},\boldsymbol{\theta})}{\pi(y_i|x_i,\boldsymbol{\theta})}$$

  we can compute

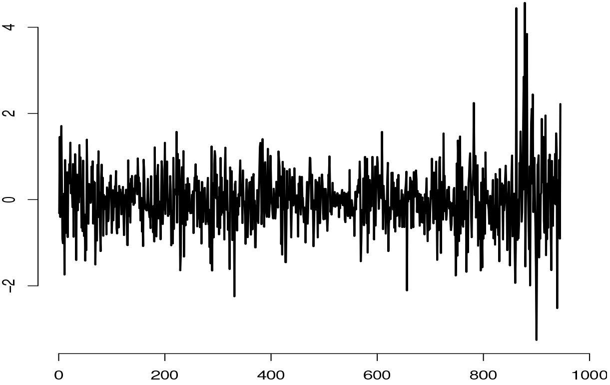
$$\pi(y_i \mid \mathbf{y}_{-i})$$

- Similar with $\pi(\boldsymbol{\theta}|\mathbf{y}_{-i})$
- Keep the integration points $\{\boldsymbol{\theta}_j\}$ fixed.
- Detect "surprising" observations:

$$\mathrm{PIT}_i = \mathrm{Prob}(y_i^{\mathrm{new}} \leq y_i \mid \mathbf{y}_{-i})$$

- To some extent, validate the model looking at $\{\mathrm{PIT}_i\}$

Stochastic Volatility model (I)



Log of the daily difference of the pound-dollar exchange rate from October 1st, 1981, to June 28th, 1985.

Stochastic Volatility model (II)

$$\begin{aligned} \eta_t &= \mu + f_t \\ f_t \mid f_1, \ldots, f_{t-1}, \tau, \phi &\sim \mathcal{N}\left(\phi f_{t-1}, 1/\tau\right), \quad |\phi| < 1 \end{aligned}$$

Observations

$$y_t \mid \eta_t \sim \mathcal{N}(0,\ \exp(\eta_t))$$

$$\boldsymbol{\theta} = (\phi, \tau)$$

Stochastic Volatility model (II)

$$\begin{aligned} \eta_t &= \mu + f_t \\ f_t \mid f_1, \ldots, f_{t-1}, \tau, \phi &\sim \mathcal{N}\left(\phi f_{t-1}, 1/\tau\right), \quad |\phi| < 1 \end{aligned}$$

Observations

$$y_t \mid \eta_t \sim \mathcal{N}(0, \exp(\eta_t))$$

$$\boldsymbol{\theta} = (\phi, \tau)$$

Stochastic Volatility model (II)

$$\begin{aligned} \eta_t &= \mu + f_t \\ f_t \mid f_1, \ldots, f_{t-1}, \tau, \phi &\sim \mathcal{N}\left(\phi f_{t-1}, 1/\tau\right), \quad |\phi| < 1 \end{aligned}$$

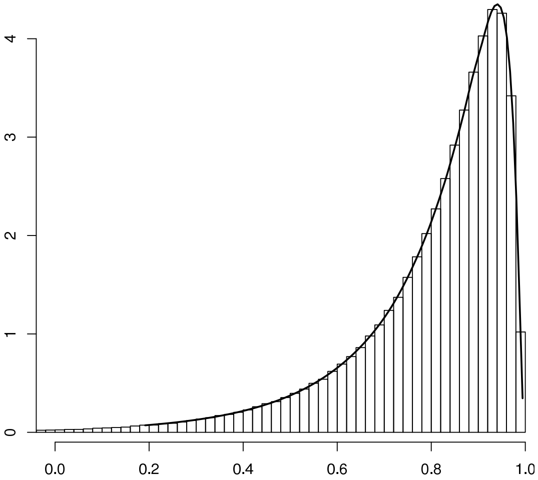
Observations

$$y_t \mid \eta_t \sim \mathcal{N}(0, \ \exp(\eta_t))$$

$$\boldsymbol{\theta} = (\phi, \tau)$$

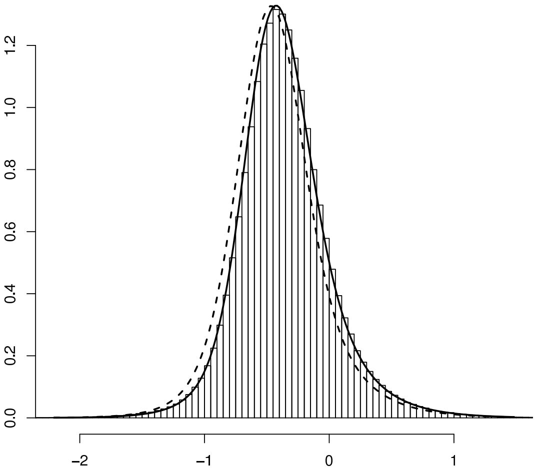
Results: 50 *first data*

Posterior marginal ϕ



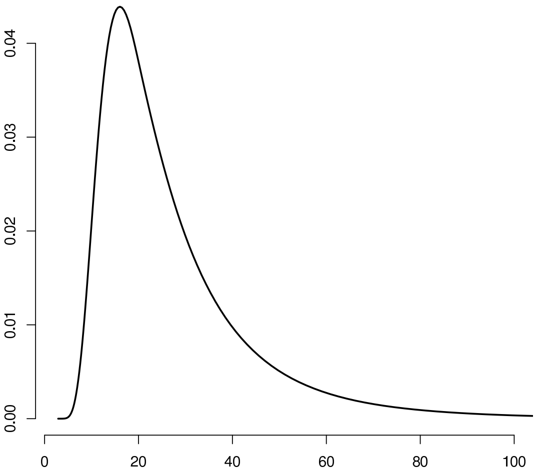
Results: 50 *first data*

Posterior marginal for μ



Revised model: all data

Observations: $y_t \mid \eta_t \ \sim \ \exp(\eta_t/2) \ \times$ Student-t_ν



Longitudinal mixed effects model: Epileptic-example from OpenBUGS (I)

Observations are the number of seizures during the two weeks before each of the four clinic visits

$$y_{ij} \sim \text{Poisson}(\exp(\eta_{ij}))$$

Linear predictor ($i = 1, \ldots, 59$ and $j = 1, \ldots, 4$)

$$\begin{aligned}
\eta_{ij} &= \beta_0 + \beta_{\text{Base}} \log(\text{Baseline}_j/4) \\
&+ \beta_{\text{Trt}} \text{Trt}_j + \beta_{\text{Trt}\times\text{Base}} \text{Trt}_j \times \log(\text{Baseline}_j/4) \\
&+ \beta_{\text{Age}} \text{Age}_j + \beta_{\text{V4}} \text{V4}_j + \epsilon_i + \nu_{ij}
\end{aligned}$$

Unstructured terms

$$\{\epsilon_i\} \overset{\text{iid}}{\sim} \mathcal{N}(0, 1/\tau_\epsilon) \qquad \{\nu_{ij}\} \overset{\text{iid}}{\sim} \mathcal{N}(0, 1/\tau_\nu) \qquad \boldsymbol{\theta} = (\tau_\epsilon, \tau_\nu)$$

Longitudinal mixed effects model: Epileptic-example from OpenBUGS (I)

Observations are the number of seizures during the two weeks before each of the four clinic visits

$$y_{ij} \sim \text{Poisson}(\exp(\eta_{ij}))$$

Linear predictor ($i = 1, \ldots, 59$ and $j = 1, \ldots, 4$)

$$\begin{aligned} \eta_{ij} &= \beta_0 + \beta_{\text{Base}} \log(\text{Baseline}_j/4) \\ &+ \beta_{\text{Trt}} \text{Trt}_j + \beta_{\text{Trt} \times \text{Base}} \text{Trt}_j \times \log(\text{Baseline}_j/4) \\ &+ \beta_{\text{Age}} \text{Age}_j + \beta_{\text{V4}} \text{V4}_j + \epsilon_i + \nu_{ij} \end{aligned}$$

Unstructured terms

$$\{\epsilon_i\} \overset{\text{iid}}{\sim} \mathcal{N}(0, 1/\tau_\epsilon) \qquad \{\nu_{ij}\} \overset{\text{iid}}{\sim} \mathcal{N}(0, 1/\tau_\nu) \qquad \boldsymbol{\theta} = (\tau_\epsilon, \tau_\nu)$$

Longitudinal mixed effects model: Epileptic-example from OpenBUGS (I)

Observations are the number of seizures during the two weeks before each of the four clinic visits

$$y_{ij} \sim \text{Poisson}(\exp(\eta_{ij}))$$

Linear predictor ($i = 1, \ldots, 59$ and $j = 1, \ldots, 4$)

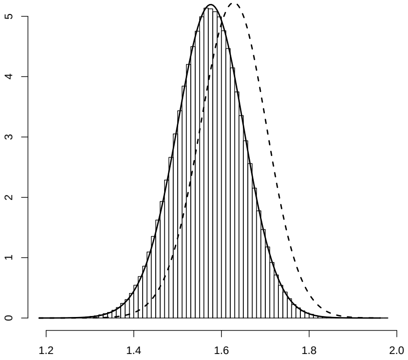
$$\begin{aligned}\eta_{ij} &= \beta_0 + \beta_{\text{Base}} \log(\text{Baseline}_j/4) \\ &+ \beta_{\text{Trt}} \text{Trt}_j + \beta_{\text{Trt}\times\text{Base}} \text{Trt}_j \times \log(\text{Baseline}_j/4) \\ &+ \beta_{\text{Age}} \text{Age}_j + \beta_{\text{V4}} \text{V4}_j + \epsilon_i + \nu_{ij}\end{aligned}$$

Unstructured terms

$$\{\epsilon_i\} \overset{\text{iid}}{\sim} \mathcal{N}(0, 1/\tau_\epsilon) \qquad \{\nu_{ij}\} \overset{\text{iid}}{\sim} \mathcal{N}(0, 1/\tau_\nu) \qquad \boldsymbol{\theta} = (\tau_\epsilon, \tau_\nu)$$

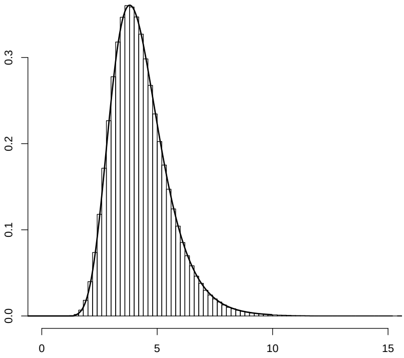
Longitudinal mixed effects model: Epileptic-example from OpenBUGS (II)

Posterior marginal for β_0



Longitudinal mixed effects model: Epileptic-example from OpenBUGS (II)

Posterior marginal for τ_ϵ



Longitudinal mixed effects model: Epileptic-example from OpenBUGS (III)

R-interface:

```
formula <- y ~ lBase4 + Trt + BT+ lAge + V4
            + f( Ind,  model="iid" )
            + f( rand, model="iid" )

fit = inla( formula, family="poisson" )

summary( fit )
plot( fit )
```

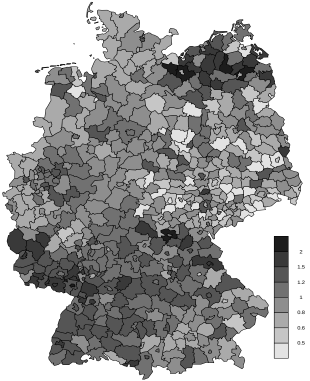
Running time on my laptop: 0.55s.

Disease mapping: The BYM-model

- Data $y_i \sim \text{Poisson}(E_i exp(\eta_i))$
- Log-relative risk $\boldsymbol{\eta} = \mathbf{u} + \mathbf{v} + \mathbf{Z}\boldsymbol{\beta}$
- Structured component $\mathbf{u}$

$$u_i \mid \mathbf{u}_{-i} \sim \mathcal{N}\left(\frac{1}{n_i}\sum_{j\sim i} u_j, \frac{1}{n_i \kappa_u}\right)$$

- Unstructured component $\mathbf{v}$
- $\boldsymbol{\theta} = (\kappa_u, \kappa_v)$



Survival models [with R.Akerkar and S.Martino]

$$\text{log(hazard)} = \text{log(baseline hazard)} + \text{covariates}$$

We are able to do

- Right or left censoring
- Piecewise constant baseline hazard (and parametric survival time)
- Log-Normal frailty
- +standard stuff

Survival models [with R.Akerkar and S.Martino]

$$\text{log(hazard)} = \text{log(baseline hazard)} + \text{covariates}$$

We are able to do

- Right or left censoring
- Piecewise constant baseline hazard (and parametric survival time)
- Log-Normal frailty
- +standard stuff

The (trival) trick

Hazard (no covariates)

$$\log \lambda(t)$$

Log-Likelihood from one observation

$$\delta_i \log \lambda(t_i) - \int_0^{t_i} \lambda(s) ds$$

where $\delta_i = 1$ if failed or 0 if (right-)cencored.

The (trival) trick

Hazard (no covariates)

$$\log \lambda(t)$$

Log-Likelihood from one observation

$$\delta_i \log \lambda(t_i) - \int_0^{t_i} \lambda(s) ds$$

where $\delta_i = 1$ if failed or 0 if (right-)cencored.

The (trival) trick

Pice-wise constant $\log \lambda(t)$ on $[0,1] + [1,t]$ gives

$$\underbrace{-\lambda^{(1)}}_{\text{log Poisson}(y=0 \mid \text{mean}=\lambda^{(1)})} + \underbrace{\delta_i \log \lambda^{(2)} - (t-1)\lambda^{(2)}}_{\text{log Poisson}(y=\delta_i \mid \text{mean}=(t-1)\lambda^{(2)})}$$

Result: Augment the model with fictive-observations

$$(\delta_i, t) \rightarrow (0, 0, 0, 0, \ldots, (\delta_i, t))$$

and we have a LGM with Poisson observations!

The (trival) trick

Pice-wise constant $\log \lambda(t)$ on $[0, 1] + [1, t]$ gives

$$\underbrace{-\lambda^{(1)}}_{\log \text{Poisson}(y=0 \mid \text{mean}=\lambda^{(1)})} + \underbrace{\delta_i \log \lambda^{(2)} - (t-1)\lambda^{(2)}}_{\log \text{Poisson}(y=\delta_i \mid \text{mean}=(t-1)\lambda^{(2)})}$$

Result: Augment the model with fictive-observations

$$(\delta_i, t) \rightarrow (0, 0, 0, 0, \ldots, (\delta_i, t))$$

and we have a LGM with Poisson observations!

Spatial survival: example

Leukaemia survival data (Henderson et al, 2002, 1043 cases.

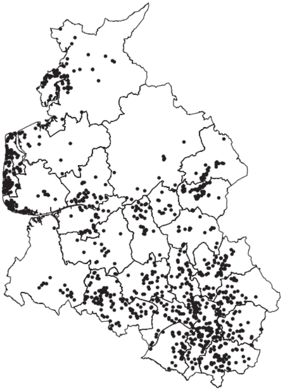


Fig. 1. Leukaemia survival data: districts of Northwest England and locations of the observations.

Spatial survival: example

$$\log(\text{hazard}) = \log(\text{baseline})$$
$$+f(\text{age})$$
$$+f(\text{white blood cell count})$$
$$+f(\text{deprivation index})$$
$$+\text{sex}$$

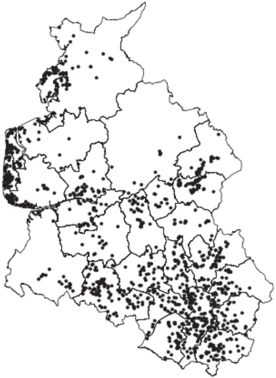


Fig. 1. Leukaemia survival data: districts of Northwest England and locations of the obse

Some internal statistics

In this example:

- Factorise $\mathbf{Q}$ (dim $= 3144$): 370 times
- Solve $\mathbf{Qx} = \mathbf{b}$: 12 915 times
- This is 50% of total CPU time

Use information about the locations!

Leukaemia survival data (Henderson et al, 2002, JASA). 1043 cases.

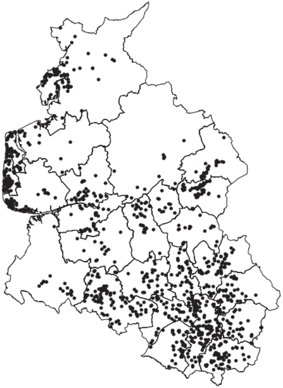


Fig. 1. Leukaemia survival data: districts of Northwest England and locations of the observations.

Gaussian fields

- Covariance function γ(distance; parameters)
- Matérn family
- Set of locations: n points
- Multivariate Normal distribution
    - Dense covariance matrix
    - Dense precision matrix
- Dense matrix calculations

$$\mathcal{O}(n^3)$$

- Often referred to as the "big n" problem

Gaussian fields

- Covariance function $\gamma(\text{distance; parameters})$
- Matérn family
- Set of locations: n points
- Multivariate Normal distribution
  - Dense covariance matrix
  - Dense precision matrix
- Dense matrix calculations

$$\mathcal{O}(n^3)$$

- Often referred to as the "big n" problem

Gaussian fields

- Covariance function γ(distance; parameters)
- Matérn family
- Set of locations: n points
- Multivariate Normal distribution
    - Dense covariance matrix
    - Dense precision matrix
- Dense matrix calculations

$$\mathcal{O}(n^3)$$

- Often referred to as the "big n" problem

Gaussian fields

- Covariance function $\gamma(\text{distance; parameters})$
- Matérn family
- Set of locations: n points
- Multivariate Normal distribution
  - Dense covariance matrix
  - Dense precision matrix
- Dense matrix calculations

$$\mathcal{O}(n^3)$$

- Often referred to as the "big n" problem

Gaussian fields

- Covariance function $\gamma(\text{distance; parameters})$
- Matérn family
- Set of locations: n points
- Multivariate Normal distribution
    - Dense covariance matrix
    - Dense precision matrix
- Dense matrix calculations

$$\mathcal{O}(n^3)$$

- Often referred to as the "big n" problem

Gaussian fields

- Covariance function γ(distance; parameters)
- Matérn family
- Set of locations: n points
- Multivariate Normal distribution
    - Dense covariance matrix
    - Dense precision matrix
- Dense matrix calculations

$$\mathcal{O}(n^3)$$

- Often referred to as the "big n" problem

Gaussian fields

- Covariance function $\gamma(\text{distance}; \text{parameters})$
- Matérn family
- Set of locations: n points
- Multivariate Normal distribution
  - Dense covariance matrix
  - Dense precision matrix
- Dense matrix calculations

$$\mathcal{O}(n^3)$$

- Often referred to as the “big n” problem

Gaussian fields

- Covariance function $\gamma(\text{distance; parameters})$
- Matérn family
- Set of locations: n points
- Multivariate Normal distribution
  - Dense covariance matrix
  - Dense precision matrix
- Dense matrix calculations

$$\mathcal{O}(n^3)$$

- Often referred to as the "big n" problem

Gaussian Markov Random fields

- $x_i | \mathbf{x}_{-i}$ only depends on the (few nearest) neighbours

$$
\mathrm{E}(x_{ij} \mid \mathbf{x}_{-ij}) = \frac{1}{20}\left(8 \begin{array}{ccccc} \circ&\circ&\circ&\circ&\circ \\ \circ&\circ&\bullet&\circ&\circ \\ \circ&\bullet&\circ&\bullet&\circ \\ \circ&\circ&\bullet&\circ&\circ \\ \circ&\circ&\circ&\circ&\circ \end{array} - 2 \begin{array}{ccccc} \circ&\circ&\circ&\circ&\circ \\ \circ&\bullet&\circ&\bullet&\circ \\ \circ&\circ&\circ&\circ&\circ \\ \circ&\bullet&\circ&\bullet&\circ \\ \circ&\circ&\circ&\circ&\circ \end{array} - 1 \begin{array}{ccccc} \circ&\circ&\bullet&\circ&\circ \\ \circ&\circ&\circ&\circ&\circ \\ \bullet&\circ&\circ&\circ&\bullet \\ \circ&\circ&\circ&\circ&\circ \\ \circ&\circ&\bullet&\circ&\circ \end{array} \right)
$$

$$
\mathrm{Prec}(x_{ij} \mid \mathbf{x}_{-ij}) = 20\kappa.
$$

- Simple conditional interpretation
- Small memory footprint
- Fast computations $\mathcal{O}(n^{3/2})$ in $\mathbb{R}^2$

Gaussian Markov Random fields

- $x_i | \mathbf{x}_{-i}$ only depends on the (few nearest) neighbours

$$
\begin{aligned}
\mathrm{E}(x_{ij} \mid \mathbf{x}_{-ij}) &= \frac{1}{20}\left(8 \begin{smallmatrix} \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \bullet & \circ & \circ \\ \circ & \bullet & \circ & \bullet & \circ \\ \circ & \circ & \bullet & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \end{smallmatrix} - 2 \begin{smallmatrix} \circ & \circ & \circ & \circ & \circ \\ \circ & \bullet & \circ & \bullet & \circ \\ \circ & \circ & \circ & \circ & \circ \\ \circ & \bullet & \circ & \bullet & \circ \\ \circ & \circ & \circ & \circ & \circ \end{smallmatrix} - 1 \begin{smallmatrix} \circ & \circ & \bullet & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \\ \bullet & \circ & \circ & \circ & \bullet \\ \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \bullet & \circ & \circ \end{smallmatrix} \right) \\
\mathrm{Prec}(x_{ij} \mid \mathbf{x}_{-ij}) &= 20\kappa.
\end{aligned}
$$

- Simple conditional interpretation
- Small memory footprint
- Fast computations $\mathcal{O}(n^{3/2})$ in $\mathbb{R}^2$

Gaussian Markov Random fields

- $x_i | \mathbf{x}_{-i}$ only depends on the (few nearest) neighbours

$$
\begin{aligned}
\mathrm{E}(x_{ij} \mid \mathbf{x}_{-ij}) &= \frac{1}{20}\left(8 \begin{array}{ccccc} \circ&\circ&\circ&\circ&\circ \\ \circ&\circ&\bullet&\circ&\circ \\ \circ&\bullet&\circ&\bullet&\circ \\ \circ&\circ&\bullet&\circ&\circ \\ \circ&\circ&\circ&\circ&\circ \end{array} - 2 \begin{array}{ccccc} \circ&\circ&\circ&\circ&\circ \\ \circ&\bullet&\circ&\bullet&\circ \\ \circ&\circ&\circ&\circ&\circ \\ \circ&\bullet&\circ&\bullet&\circ \\ \circ&\circ&\circ&\circ&\circ \end{array} - 1 \begin{array}{ccccc} \circ&\circ&\bullet&\circ&\circ \\ \circ&\circ&\circ&\circ&\circ \\ \bullet&\circ&\circ&\circ&\bullet \\ \circ&\circ&\circ&\circ&\circ \\ \circ&\circ&\bullet&\circ&\circ \end{array} \right) \\
\mathrm{Prec}(x_{ij} \mid \mathbf{x}_{-ij}) &= 20\kappa.
\end{aligned}
$$

- Simple conditional interpretation
- Small memory footprint
- Fast computations $\mathcal{O}(n^{3/2})$ in $\mathbb{R}^2$

Gaussian Markov Random fields

- $x_i | \mathbf{x}_{-i}$ only depends on the (few nearest) neighbours

$$
\begin{aligned}
\mathrm{E}(x_{ij} \mid \mathbf{x}_{-ij}) &= \frac{1}{20}\left(8 \begin{smallmatrix} \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \bullet & \circ & \circ \\ \circ & \bullet & \circ & \bullet & \circ \\ \circ & \circ & \bullet & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \end{smallmatrix} - 2 \begin{smallmatrix} \circ & \circ & \circ & \circ & \circ \\ \circ & \bullet & \circ & \bullet & \circ \\ \circ & \circ & \circ & \circ & \circ \\ \circ & \bullet & \circ & \bullet & \circ \\ \circ & \circ & \circ & \circ & \circ \end{smallmatrix} - 1 \begin{smallmatrix} \circ & \circ & \bullet & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \\ \bullet & \circ & \circ & \circ & \bullet \\ \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \bullet & \circ & \circ \end{smallmatrix}\right) \\
\mathrm{Prec}(x_{ij} \mid \mathbf{x}_{-ij}) &= 20\kappa.
\end{aligned}
$$

- Simple conditional interpretation
- Small memory footprint
- Fast computations $\mathcal{O}(n^{3/2})$ in $\mathbb{R}^2$

Can we use GMRFs as proxies?

Is there a GMRF with local neighbourhood that can approximate a Gaussian field with given covariance function?

Matérn fields [with F.Lindgren and J.Lindström]

The solution of

$$(\kappa^2 - \Delta)^{\alpha/2}\mathbf{x}(\mathbf{s}) = \epsilon(\mathbf{s})$$

in $\mathbb{R}^2$, is a Matérn field:

$$\mathrm{Cov}(\mathbf{x}(\mathbf{s}), \mathbf{x}(\mathbf{s}+\boldsymbol{\tau})) \propto (\kappa \, \|\boldsymbol{\tau}\|)^{\nu} \;\; K_{\nu}(\kappa \, \|\boldsymbol{\tau}\|), \qquad \alpha = \nu + \mathrm{dim}/2$$

and K_ν is the modified Bessel function.

Matérn fields on manifolds (definition)

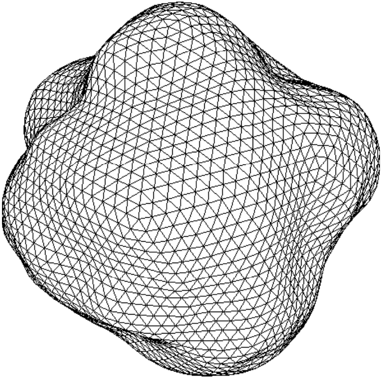
- The solution of

$$(\kappa^2 - \Delta)^{\alpha/2}\mathbf{x}(\mathbf{s}) = \epsilon(\mathbf{s})$$

  on the manifold $\mathcal{S}$

- Driven by Gaussian "white noise" on $\mathcal{S}$

$$\mathrm{Cov}(\epsilon(A_i), \epsilon(A_j)) = \int_{A_i \cap A_j} d\mathcal{S}(\mathbf{s})$$



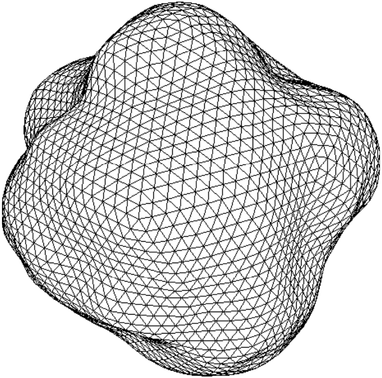
Matérn fields on manifolds (definition)

- The solution of

$$(\kappa^2 - \Delta)^{\alpha/2}\mathbf{x}(\mathbf{s}) = \epsilon(\mathbf{s})$$

  on the manifold $\mathcal{S}$
- Driven by Gaussian "white noise" on $\mathcal{S}$

$$\mathrm{Cov}(\epsilon(A_i), \epsilon(A_j)) = \int_{A_i \cap A_j} d\mathcal{S}(\mathbf{s})$$



Result

Can "solve"

$$(\kappa^2 - \Delta)^{\alpha/2}\mathbf{x}(\mathbf{s}) = \epsilon(\mathbf{s}), \quad \alpha = 1, 2, 3, \ldots, \quad \alpha = \nu + \mathrm{dim}/2$$

for

- any κ
- any triangulation
- on any ("regular") manifold

"solve" means here: write down, explicitly, the corresponding precision matrix of the (local) GMRF

Result

Can "solve"

$$(\kappa^2 - \Delta)^{\alpha/2}\mathbf{x}(\mathbf{s}) = \epsilon(\mathbf{s}), \quad \alpha = 1, 2, 3, \ldots, \quad \alpha = \nu + \mathsf{dim}/2$$

for

- any κ
- any triangulation
- on any ("regular") manifold

"solve" means here: write down, explicitly, the corresponding precision matrix of the (local) GMRF

Result

Can "solve"

$$(\kappa^2 - \Delta)^{\alpha/2}\mathbf{x}(\mathbf{s}) = \boldsymbol{\epsilon}(\mathbf{s}), \quad \alpha = 1, 2, 3, \ldots, \quad \alpha = \nu + \mathrm{dim}/2$$

for

- any κ
- any triangulation
- on any ("regular") manifold

"solve" means here: write down, explicitly, the corresponding precision matrix of the (local) GMRF

Result

Can "solve"

$$(\kappa^2 - \Delta)^{\alpha/2} \mathbf{x}(\mathbf{s}) = \boldsymbol{\epsilon}(\mathbf{s}), \quad \alpha = 1, 2, 3, \ldots, \quad \alpha = \nu + \text{dim}/2$$

for

- any κ
- any triangulation
- on any ("regular") manifold

"solve" means here: write down, explicitly, the corresponding precision matrix of the (local) GMRF

Example: Explicit results for a regular lattice

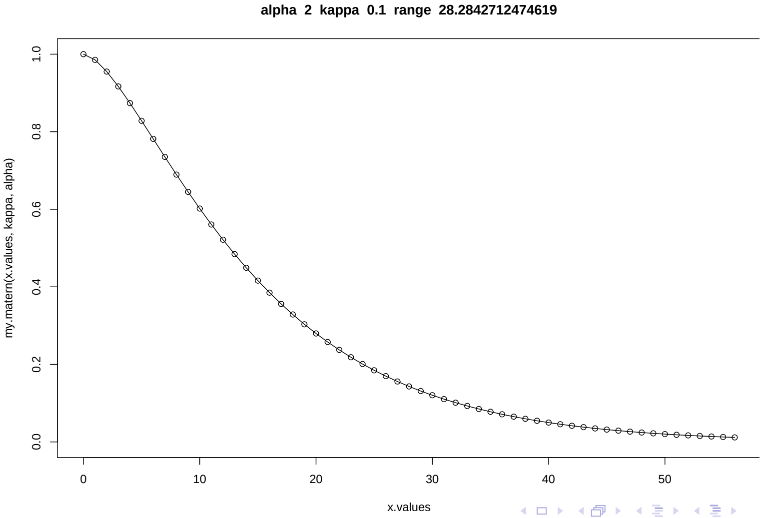
For $\alpha = 2$ ($\nu = 1$)

$$\mathbf{Q} = \frac{\kappa^{-2}}{4\pi} \times \begin{bmatrix} & & 1 & & \\ & 2 & -2(4+\kappa^2) & 2 & \\ 1 & -2(4+\kappa^2) & \color{red}{4+(4+\kappa^2)^2} & -2(4+\kappa^2) & 1 \\ & 2 & -2(4+\kappa^2) & 2 & \\ & & 1 & & \end{bmatrix}$$

Range $\approx \sqrt{8}/\kappa$

Can compute boundary corrections

Example: the fit



Same example with anisotropy

For $\alpha = 2$ $(\nu = 1)$

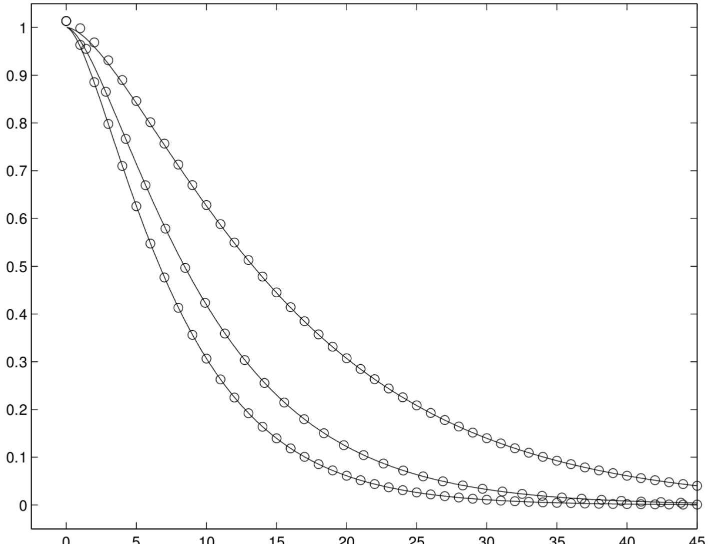
$$a = \kappa_1^{-2} \qquad b = \kappa_2^{-2} \qquad c = 1 + 2(a+b)$$

$$\mathbf{Q} = \frac{1}{4\pi\sqrt{ab}} \begin{bmatrix} & & a^2 & & \\ & 2ab & -2ca & 2ab & \\ b^2 & -2cb & c^2 + 2(a^2+b^2) & -2cb & b^2 \\ & 2ab & -2ca & 2ab & \\ & & a^2 & & \end{bmatrix}$$

Range $\approx \sqrt{8}/\kappa_1$ and $\sqrt{8}/\kappa_2$

Can compute boundary corrections

Example: Anisotropy on a regular grid



Survival example: irregular grid

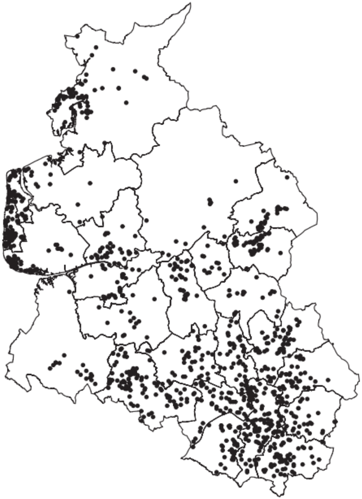
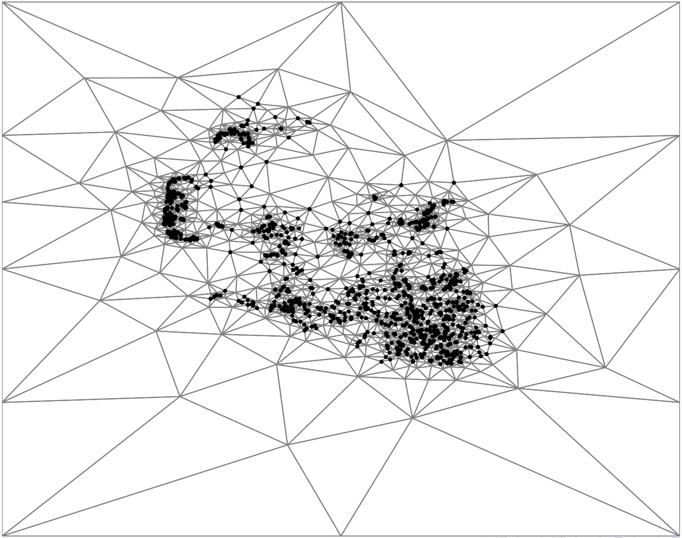
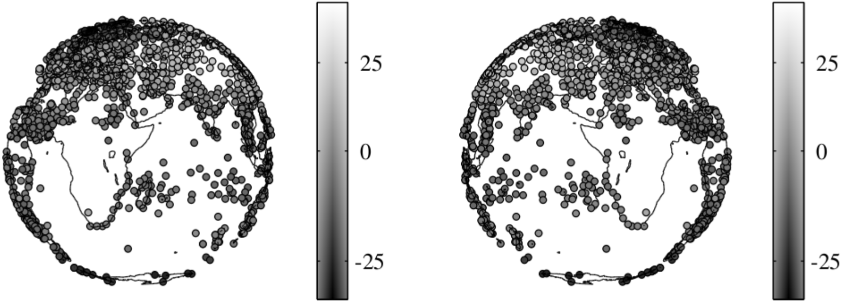


Fig. 1. Leukaemia survival data: districts of Northwest England and locations of the observations.

Survival example: irregular grid



Spatial-temporal example



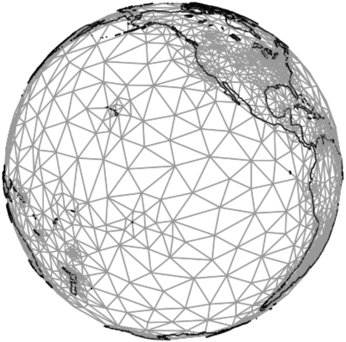
Spatio-temporal example

- Monthly (January) average sea-level pressure at around 1 000 stations
- About 250 stations withheld for validation purposes
- 5 years of data

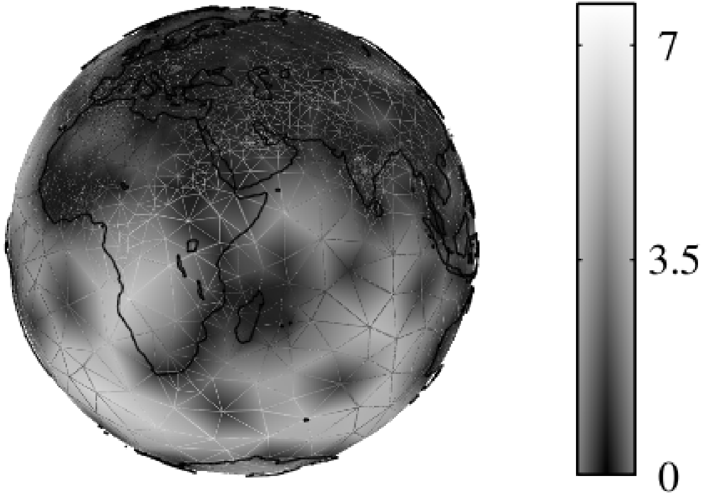
Simple model:

- Model for the mean
- Treat the data as (conditionally) independent
- Matérn ($\alpha = 2$, $\nu = 1$) covariance function, or oscillating field.

Spatial-temporal example



Spatial-temporal example



Binary MRF

- Defined on a regular lattice size n
- Lattice points x_i take values $\{-1, 1\}$
- Full conditional $\pi(x_i|\mathbf{x}_{-i}, \beta) = \pi(x_i|\text{neighbours of } i, \beta)$.

Ising model

$$\pi(\mathbf{x} \mid \beta) = \frac{q(\mathbf{x} \mid \beta)}{z(\beta)} = \frac{\exp\left(\beta_1 \sum_i x_i + \beta_2 \sum_{i \sim j} x_i x_j\right)}{z(\beta)}$$

The normalising constant is

$$z(\beta) = \sum_{\mathbf{x}} q(\mathbf{x} \mid \beta)$$

Binary MRF

- Defined on a regular lattice size n
- Lattice points x_i take values $\{-1, 1\}$
- Full conditional $\pi(x_i|\mathbf{x}_{-i}, \beta) = \pi(x_i|\text{neighbours of } i, \beta)$.

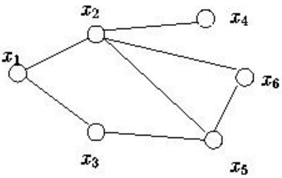
Ising model

$$\pi(\mathbf{x} \mid \beta) = \frac{q(\mathbf{x} \mid \beta)}{z(\beta)} = \frac{\exp\left(\beta_1 \sum_i x_i + \beta_2 \sum_{i \sim j} x_i x_j\right)}{z(\beta)}$$

The normalising constant is

$$z(\beta) = \sum_{\mathbf{x}} q(\mathbf{x} \mid \beta)$$

Undirected Graphs and joint distributions

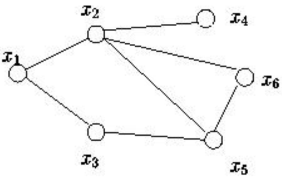


$$\pi(\mathbf{x}) = \frac{1}{z}\psi(x_1, x_2)\psi(x_1, x_3)\psi(x_2, x_4)\psi(x_3, x_5)\psi(x_2, x_5, x_6)$$

$$z = \sum_{x_1} \cdots \sum_{x_6} \psi(x_1, x_2)\psi(x_1, x_3)\psi(x_2, x_4)\psi(x_3, x_5)\psi(x_2, x_5, x_6)$$

Computing z costs $\mathcal{O}(2^6)$

Undirected Graphs and joint distributions

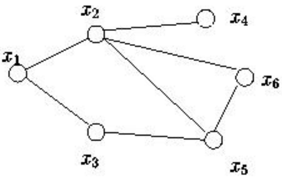


$$\pi(\mathbf{x}) = \frac{1}{z}\psi(x_1,x_2)\psi(x_1,x_3)\psi(x_2,x_4)\psi(x_3,x_5)\psi(x_2,x_5,x_6)$$

$$z = \sum_{x_1}\cdots\sum_{x_6}\psi(x_1,x_2)\psi(x_1,x_3)\psi(x_2,x_4)\psi(x_3,x_5)\psi(x_2,x_5,x_6)$$

Computing z costs $\mathcal{O}(2^6)$

Undirected Graphs and joint distributions



$$\pi(\mathbf{x}) = \frac{1}{z}\psi(x_1, x_2)\psi(x_1, x_3)\psi(x_2, x_4)\psi(x_3, x_5)\psi(x_2, x_5, x_6)$$

$$z = \sum_{x_1} \cdots \sum_{x_6} \psi(x_1, x_2)\psi(x_1, x_3)\psi(x_2, x_4)\psi(x_3, x_5)\psi(x_2, x_5, x_6)$$

Computing z costs $\mathcal{O}(2^6)$

Undirected Graphs and joint distributions

The "trick"

$$
\begin{aligned}
z &= \sum_{x_1} \sum_{x_2} \psi(x_1, x_2) \sum_{x_3} \psi(x_1, x_3) \sum_{x_4} \psi(x_2, x_4) \\
&\quad \sum_{x_5} \psi(x_3, x_5) \sum_{x_6} \psi(x_2, x_5, x_6)
\end{aligned}
$$

No more than 3 terms appear in any summand. Computational complexity is decreased!

Recursive schemes: Bartolucci & Besag (2002) and Reeves & Pettitt (2004).

Undirected Graphs and joint distributions

The "trick"

$$
\begin{aligned}
z = & \sum_{x_1} \sum_{x_2} \psi(x_1, x_2) \sum_{x_3} \psi(x_1, x_3) \sum_{x_4} \psi(x_2, x_4) \\
& \sum_{x_5} \psi(x_3, x_5) \sum_{x_6} \psi(x_2, x_5, x_6)
\end{aligned}
$$

No more than 3 terms appear in any summand. Computational complexity is decreased!

Recursive schemes: Bartolucci & Besag (2002) and Reeves & Pettitt (2004).

Undirected Graphs and joint distributions

The "trick"

$$
\begin{aligned}
z &= \sum_{x_1}\sum_{x_2}\psi(x_1,x_2)\sum_{x_3}\psi(x_1,x_3)\sum_{x_4}\psi(x_2,x_4) \\
&\quad \sum_{x_5}\psi(x_3,x_5)\sum_{x_6}\psi(x_2,x_5,x_6)
\end{aligned}
$$

No more than 3 terms appear in any summand. Computational complexity is decreased!

Recursive schemes: Bartolucci & Besag (2002) and Reeves & Pettitt (2004).

Results for binary MRFs

Nial and Rue (2007) "completed" the algorithm by Reeves & Pettitt (2004), to allow for

- compute (all) normalising constants,
- compute marginals $\pi(x_i|\beta)$, $\pi(x_i, x_j|\beta)$, ...
- sample exact,
- compute modal configuration,
- compute marginal likelihood.

For lattices where **smallest** dimension m does not exceed 19

Results for binary MRFs

Nial and Rue (2007) "completed" the algorithm by Reeves & Pettitt (2004), to allow for

- compute (all) normalising constants,
- compute marginals $\pi(x_i|\beta)$, $\pi(x_i, x_j|\beta)$, ...
- sample exact,
- compute modal configuration,
- compute marginal likelihood.

For lattices where **smallest** dimension m does not exceed 19

Results for binary MRFs

Nial and Rue (2007) "completed" the algorithm by Reeves & Pettitt (2004), to allow for

- compute (all) normalising constants,
- compute marginals $\pi(x_i|\beta)$, $\pi(x_i, x_j|\beta)$, ...
- sample exact,
- compute modal configuration,
- compute marginal likelihood.

For lattices where **smallest** dimension m does not exceed 19

Results for binary MRFs

Nial and Rue (2007) “completed” the algorithm by Reeves & Pettitt (2004), to allow for

- compute (all) normalising constants,
- compute marginals $\pi(x_i|\beta)$, $\pi(x_i, x_j|\beta)$, ...
- sample exact,
- compute modal configuration,
- compute marginal likelihood.

For lattices where **smallest** dimension m does not exceed 19

Results for binary MRFs

Nial and Rue (2007) "completed" the algorithm by Reeves & Pettitt (2004), to allow for

- compute (all) normalising constants,
- compute marginals $\pi(x_i|\beta)$, $\pi(x_i, x_j|\beta)$, ...
- sample exact,
- compute modal configuration,
- compute marginal likelihood.

For lattices where **smallest** dimension m does not exceed 19

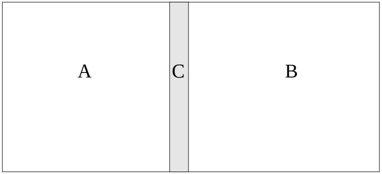
Results for binary MRFs

Nial and Rue (2007) "completed" the algorithm by Reeves & Pettitt (2004), to allow for

- compute (all) normalising constants,
- compute marginals $\pi(x_i|\beta)$, $\pi(x_i, x_j|\beta)$, ...
- sample exact,
- compute modal configuration,
- compute marginal likelihood.

For lattices where **smallest** dimension m does not exceed 19

Approximate inference for large lattices

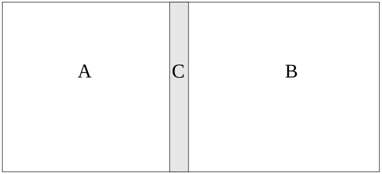


Consider a large lattice partitioned into sub-lattices A and B, separated by a column of lattice points C

$$\pi(\mathbf{x} \mid \beta) = \pi(\mathbf{x}_A \mid \beta, \mathbf{x}_C)\, \pi(\mathbf{x}_C \mid \beta)\pi(\mathbf{x}_B \mid \beta, \mathbf{x}_C).$$

If we can compute the marginal of $\mathbf{x}_C$ then we have two independent smaller problems.

Approximate inference for large lattices

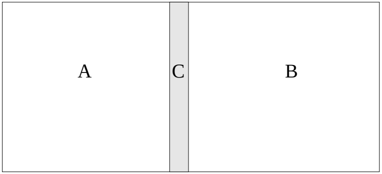


Consider a large lattice partitioned into sub-lattices A and B, separated by a column of lattice points C

$$\pi(\mathbf{x} \mid \beta) = \pi(\mathbf{x}_A \mid \beta, \mathbf{x}_C)\, \pi(\mathbf{x}_C \mid \beta)\pi(\mathbf{x}_B \mid \beta, \mathbf{x}_C).$$

If we can compute the marginal of $\mathbf{x}_C$ then we have two independent smaller problems.

Approximate inference for large lattices

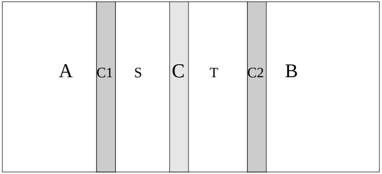


Consider a large lattice partitioned into sub-lattices A and B, separated by a column of lattice points C

$$\pi(\mathbf{x} \mid \beta) = \pi(\mathbf{x}_A \mid \beta, \mathbf{x}_C)\, \pi(\mathbf{x}_C \mid \beta)\pi(\mathbf{x}_B \mid \beta, \mathbf{x}_C).$$

If we can compute the marginal of $\mathbf{x}_C$ then we have two independent smaller problems.

Approximating the marginal of $\mathbf{x}_C$



$$\begin{aligned}
\pi(\mathbf{x}_C \mid \beta) &= \frac{\pi(\mathbf{x} \mid \beta)}{\pi(\mathbf{x}_A \mid \mathbf{x}_C, \beta)\pi(\mathbf{x}_B \mid \mathbf{x}_C, \beta)} \\
&\approx \frac{\pi(\mathbf{x}_{S,C,T} \mid \mathbf{x}_{C1}, \mathbf{x}_{C2}, \beta)}{\pi(\mathbf{x}_S \mid \mathbf{x}_C, \mathbf{x}_{C1}, \beta)\;\pi(\mathbf{x}_T \mid \mathbf{x}_C, \mathbf{x}_{C2}, \beta)}.
\end{aligned}$$

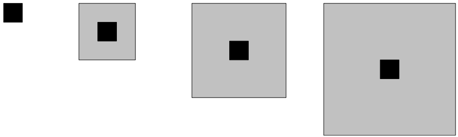
Approximations

Same basic ideas as in INLA can be applied:

$$\pi(x_i \mid \mathbf{y}) = \sum_j \pi(x_i \mid \mathbf{y}, \boldsymbol{\theta}_j)\, \pi(\theta_j \mid \mathbf{y})\, \Delta_j$$

etc...

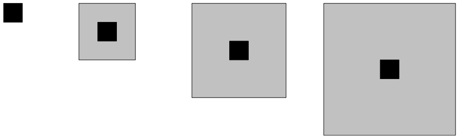
Approximations...



Improved approximations for $\pi(x_i|\mathbf{y}, \boldsymbol{\theta}_j)$ for increasing size of window centred at site i.

- More difficult MRF models: talk by H.Tjelmeland and H.Austad
- Approximating normalising constants: previously announced talk by N.Friel

Approximations...



Improved approximations for $\pi(x_i|\mathbf{y}, \boldsymbol{\theta}_j)$ for increasing size of window centred at site i.

- More difficult MRF models: talk by H.Tjelmeland and H.Austad
- Approximating normalising constants: previously announced talk by N.Friel

Change-point models

Stat Comput (2006) 16: 203–213
DOI 10.1007/s11222-006-8450-8

Exact and efficient Bayesian inference for multiple changepoint problems

Paul Fearnhead

based on work by Yo (1984, AoS), Barry and Hartigan (1992 AoS, 1993 JASA).

Model

- Data $y_1, \ldots, y_n$
- Conditionally independent given $\boldsymbol{\theta}$
- $\boldsymbol{\theta}$ is piece-wise constant $(\theta_1, \ldots, \theta_m)$ in m-segments
- Independent θ_j's over segments

Recursions

- Forward-backward based recursions
- Require the marginal likelihood in each segment
- Fixed number of change-points: compute posterior marginal for each change-point and corresponding θ_j.
- Random number of change-points: compute the posterior marginal for the number of change-points.

Coal mining disaster data

718 PETER J. GREEN

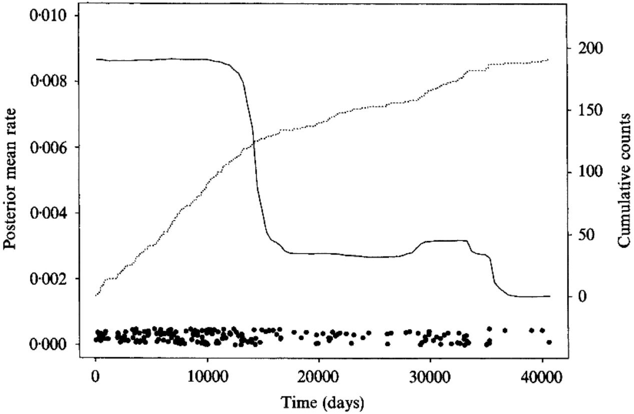


Fig. 1. Coal mining disaster data, 1851–1962: dates of disasters, cumulative counting process (dotted curve) and posterior mean rate of occurrence (solid curve).

Coal mining disaster data: number of change-points

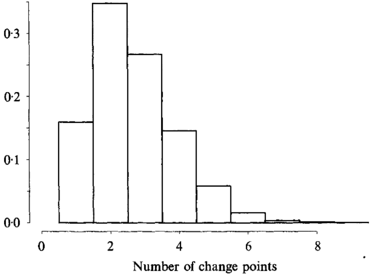
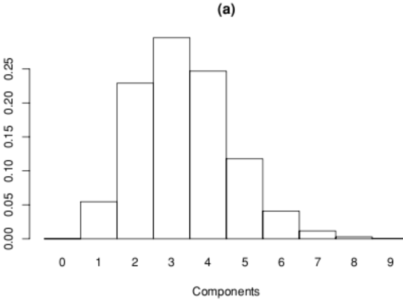


Fig. 2. Coal mining disaster data: posterior distribution of k, the number of change-points.

RJMCMC



Exact: discrete time

Coal mining disaster data

of the algorithm. For example, in the analysis of the coal-mining disaster data in Green (1995), the reversible jump MCMC algorithm had not converged. The reanalysis of the data in Green (2003), using a reversible jump MCMC algorithm run for 25 times as long, does fully explore the posterior distribution. The exact simulation method we describe here avoids any problems of needing to diagnose convergence of an MCMC algorithm.

Change-point models

- Recursions can be "tuned" to derive near instant algorithms
- Extentions are possible
- Need marginal likelihood

Parallel computing

- (Near) every new computer is dual-core/quad-core
- Programs has to take advantage of this in order to gain speedup.

OpenMP (www.openmp.org)

Dual/Quad-core computers has *shared memory*; makes (my) life easier.
OpenMP defines a nice set of tools for creating parallel programs (multi-threading), in C/C++/Fortran.
OpenMP was included in gcc/gfortran-4.2.

Example: Loops

Standard C-code

```
        for(i=0; i<N; i++)
            x[i] = GetNewSample(...)   // Independent loop
```

with OpenMP directives

```
    #pragma omp parallel for private(i)
        for(i=0; i<N; i++)
            x[i] = GetNewSample(...)
```

Example: Parallel regions

Standard C-code

```
DoTaskA(); // Independent tasks
DoTaskB();
```

with OpenMP directives

```
#pragma omp parallel sections
{
#pragma omp section
    DoTaskA();

#pragma omp section
    DoTaskB();
}
```

Examples

Spatial-survival example:

Number of threads	CPU seconds
1	15.7
2	11.2

- optimisation is less parallel
- integration is parallel

Examples

Spatial-survival example:

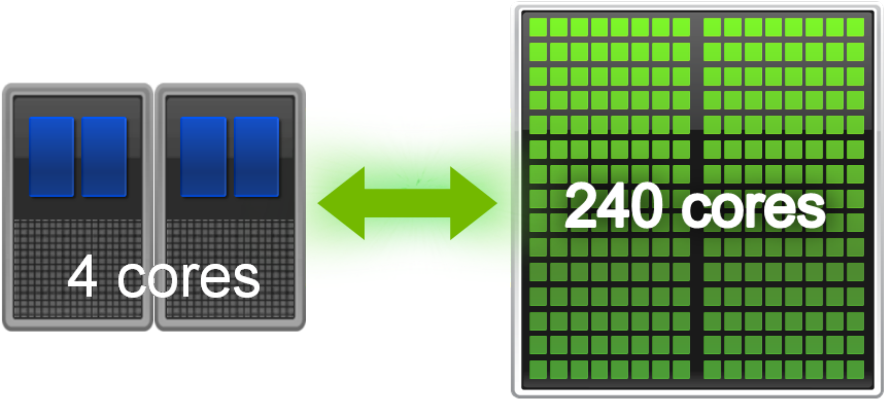
Number of threads	CPU seconds
1	15.7
2	11.2

- optimisation is less parallel
- integration is parallel

(GP)GPU Computing

- GPU computing is the use of a GPU (graphics processing unit) to do general purpose scientific and engineering computing.
- The model for GPU computing is to use a CPU and GPU together in a heterogeneous computing model.
- The sequential part of the application runs on the CPU and the computationally-intensive part runs on the GPU.

Model



The future

"GPUs have evolved to the point where many real-world applications are easily implemented on them and run significantly faster than on multi-core systems. Future computing architectures will be hybrid systems with parallel-core GPUs working in tandem with multi-core CPUs."

Prof. Jack Dongarra
Director of the Innovative Computing Laboratory
The University of Tennessee

Challenge

More parallel

- implementation
- case studies
- algorithm development

Summary (I)

- Good and fast approximations for *posterior marginals*++ do exists for certain model classes
- LGM are particular important and successful
  - HUGE class
  - fast and generic algorithm and implementation
  - makes these models usable on routinely basis
- Spatial Latent Skew-Normals: talk by J.Eidsvik
- More on approximations in the Machine-Learning literature...

Software: `www.math.ntnu.no/~hrue/GMRFLib`

Summary (I)

- Good and fast approximations for *posterior marginals*++ do exists for certain model classes
- LGM are particular important and successful
  - HUGE class
  - fast and generic algorithm and implementation
  - makes these models usable on routinely basis
- Spatial Latent Skew-Normals: talk by J.Eidsvik
- More on approximations in the Machine-Learning literature...

Software: `www.math.ntnu.no/~hrue/GMRFLib`

Summary (I)

- Good and fast approximations for *posterior marginals*++ do exists for certain model classes
- LGM are particular important and successful
  - HUGE class
  - fast and generic algorithm and implementation
  - makes these models usable on routinely basis
- Spatial Latent Skew-Normals: talk by J.Eidsvik
- More on approximations in the Machine-Learning literature...

Software: `www.math.ntnu.no/~hrue/GMRFLib`

Summary (II)

- Analytical intractable models do not automatic require MCMC!
- Computing (simple) posterior marginals is (much) simpler!
- The literature is partly miss-leading on these issues

Summary (III)

Combine $\{$MC,S,$\}$MC and Laplace approximations?

Laplace Expansions in Markov Chain Monte Carlo Algorithms

Chantal Guihenneuc-Jouyaux and Judith Rousseau

Journal of Computational and Graphical Statistics, Volume 14, Number 1, Pages 75–94
DOI: 10.1198/106186005X25727