Plan for today

We talk about:

- Markov property, conditional independence, precision matrix
- Gaussian Markov random fields
- Integrated nested Laplace approximations

Numerical algorithms for sparse matrices: scaling properties

Cholesky factorization of "sparse" $\mathbf{Q}$

- Time: $\mathcal{O}(n)$

This is to be compared with general $\mathcal{O}(n^3)$ algorithm for the Cholesky factorization of a dense matrix.

Numerical algorithms for sparse matrices: scaling properties

Cholesky factorization of "sparse" $\mathbf{Q}$

- Time: $\mathcal{O}(n)$
- Space: $\mathcal{O}(n^{3/2})$

This is to be compared with general $\mathcal{O}(n^3)$ algorithm for the Cholesky factorization of a dense matrix.

Numerical algorithms for sparse matrices: scaling properties

Cholesky factorization of "sparse" $\mathbf{Q}$

- Time: $\mathcal{O}(n)$
- Space: $\mathcal{O}(n^{3/2})$
- Space-time: $\mathcal{O}(n^2)$

This is to be compared with general $\mathcal{O}(n^3)$ algorithm for the Cholesky factorization of a dense matrix.

Gaussian Markov random field (GMRF)

A GMRF is a simple construct

- A normal distributed random vector

$$\boldsymbol{x} = (x_1, \ldots, x_n)^T$$

- Additional Markov properties:

$$x_i \perp x_j \mid \boldsymbol{x}_{-ij}$$

x_i and x_j are conditional independent (CI).

Conditional independence

If $x_i \perp x_j \mid \boldsymbol{x}_{-ij}$ for a set of $\{i, j\}$, then we need to constrain the parametrisation of the GMRF.

Conditional independence

If $x_i \perp x_j \mid \boldsymbol{x}_{-ij}$ for a set of $\{i, j\}$, then we need to constrain the parametrisation of the GMRF.

- Covariance matrix: difficult
- Precision matrix: easy

Conditional independence and the precision matrix

The density of a zero mean Gaussian

$$\pi(\boldsymbol{x}) \propto |\boldsymbol{Q}|^{1/2} \exp\left(-\frac{1}{2}\boldsymbol{x}^T \boldsymbol{Q}\boldsymbol{x}\right)$$

Conditional independence and the precision matrix

The density of a zero mean Gaussian

$$\pi(\boldsymbol{x}) \propto |\boldsymbol{Q}|^{1/2} \exp\left(-\frac{1}{2}\boldsymbol{x}^T \boldsymbol{Q}\boldsymbol{x}\right)$$

Constraining the parametrisation to obey CI properties

Theorem

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

Use a (undirected) graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to represent the CI properties,

$\mathcal{V}$ Vertices: $1, 2, \ldots, n$.

$\mathcal{E}$ Edges $\{i, j\}$

Use a (undirected) graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to represent the CI properties,

$\mathcal{V}$ Vertices: $1, 2, \ldots, n$.

$\mathcal{E}$ Edges $\{i, j\}$

- No edge between i and j if $x_i \perp x_j \mid \boldsymbol{x}_{-ij}$.
- An edge between i and j if $x_i \not\perp x_j \mid \boldsymbol{x}_{-ij}$.

Definition of a GMRF

A random vector $\boldsymbol{x} = (x_1, \ldots, x_n)^T$ is called a GMRF wrt the graph $\mathcal{G} = (\mathcal{V} = \{1, \ldots, n\}, \mathcal{E})$ with mean $\boldsymbol{\mu}$ and precision matrix $\boldsymbol{Q} > 0$, iff its density has the form

$$\pi(\boldsymbol{x}) = (2\pi)^{-n/2} |\boldsymbol{Q}|^{1/2} \exp\left(-\frac{1}{2} (\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{Q} (\boldsymbol{x} - \boldsymbol{\mu}) \right)$$

and

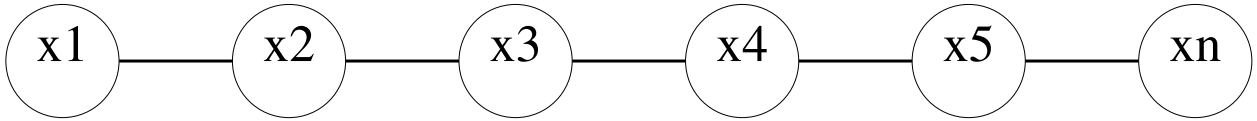
$$Q_{ij} \neq 0 \quad \Longleftrightarrow \quad \{i, j\} \in \mathcal{E} \quad \text{for all} \quad i \neq j.$$

Simple example of a GMRF

Auto-regressive process of order 1

$$x_t \mid x_{t-1}, \ldots, x_1 \sim \mathcal{N}(\phi x_{t-1}, 1), \quad t = 2, \ldots, n$$

and $x_1 \sim \mathcal{N}(0, (1-\phi^2)^{-1})$.



Tridiagonal precision matrix

$$\boldsymbol{Q} = \begin{pmatrix} 1 & -\phi & & & & \\ -\phi & 1+\phi^2 & -\phi & & & \\ & \ddots & \ddots & \ddots & & \\ & & -\phi & 1+\phi^2 & -\phi & \\ & & & -\phi & 1 & \end{pmatrix}$$

Interpretation of elements of $\boldsymbol{Q}$

Let $\boldsymbol{x}$ be a GMRF wrt $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with mean $\boldsymbol{\mu}$ and precision matrix $\boldsymbol{Q} > 0$, then

$$\begin{aligned}
\mathsf{E}(x_i \mid \boldsymbol{x}_{-i}) &= \mu_i - \frac{1}{Q_{ii}} \sum_{j:j\sim i} Q_{ij}(x_j - \mu_j), \\
\mathsf{Var}(x_i \mid \boldsymbol{x}_{-i}) &= 1/Q_{ii} \quad \text{and} \\
\mathsf{Corr}(x_i, x_j \mid \boldsymbol{x}_{-ij}) &= -\frac{Q_{ij}}{\sqrt{Q_{ii}Q_{jj}}}, \quad i \neq j.
\end{aligned}$$

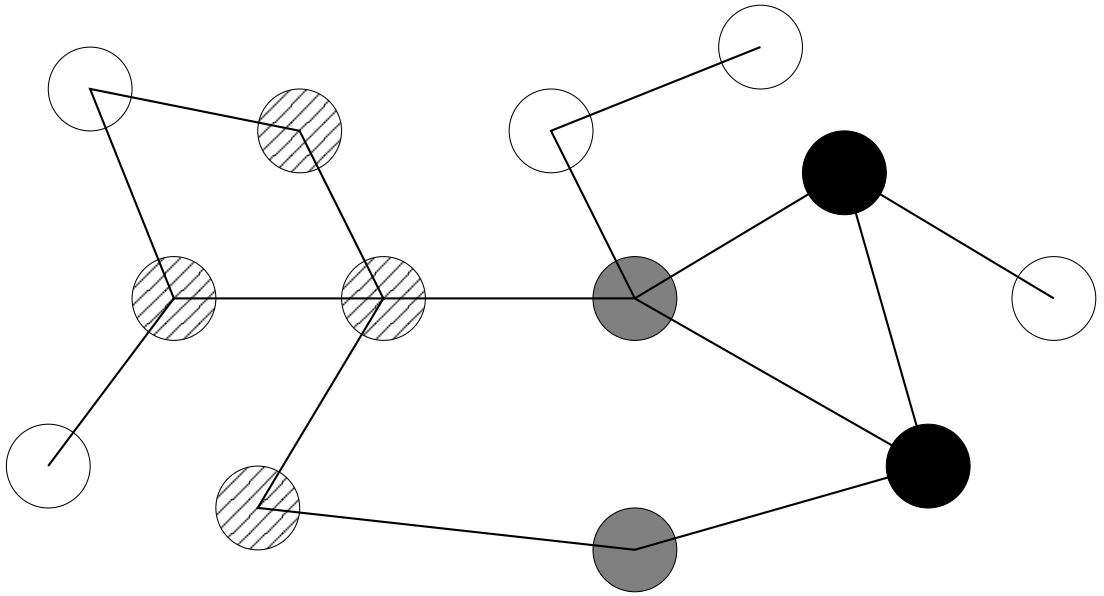
Markov properties

Let $\boldsymbol{x}$ be a GMRF wrt $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Then the following are equivalent.

The global Markov property:

$$\boldsymbol{x}_A \perp \boldsymbol{x}_B \mid \boldsymbol{x}_C$$

for all disjoint sets A, B and C where C separates A and B, and A and B are non-empty.



Hierarchical Bayesian models

Hierarchical models are an extremely useful tool in Bayesian model building.

Three parts:

- Observation model $\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta}$: Encodes information about observed data.
- The latent model $\boldsymbol{x}|\boldsymbol{\theta}$: The unobserved process.
- Hyperpriors for $\boldsymbol{\theta}$: Models for all of the parameters in the observation and latent processes.

Bayesian computing

The posterior distribution is given by

$$\mathrm{p}(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) \propto \mathrm{p}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\, \mathrm{p}(\boldsymbol{x}|\boldsymbol{\theta})\, \mathrm{p}(\boldsymbol{\theta})$$

We are interested in the posterior marginal quantities like $\mathrm{p}(x_i|\boldsymbol{y})$ and $\mathrm{p}(\theta_j|\boldsymbol{y})$. We have discussed how to do Bayesian inference using MCMC, for example Stan, and discussed pros and cons.

Approximate inference

Except for the (trivial) cases when everything can be computed exactly (maybe up to very small integration error), we can never do exact inference.

- Integrated nested Laplace approximations (INLA) are a promising alternative to inference via MCMC in latent Gaussian models (Rue et al, 2009, JRSS-B).
- The methodology is particularly attractive if the latent Gaussian model is a Gaussian Markov random field (GMRF) (Rue and Held, 2005).

Model framework

Latent Gaussian models

$$\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\theta} \sim \prod_i \pi(y_i|\eta_i,\boldsymbol{\theta})$$

$$\boldsymbol{x}|\boldsymbol{\theta} \sim \pi(\boldsymbol{x}|\boldsymbol{\theta}) \sim \mathcal{N}(\mathbf{0},\mathbf{Q}(\boldsymbol{\theta})^{-1}) \qquad \text{Gaussian!}$$

$$\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta}) \qquad \text{Not Gaussian}$$

where the precision matrix $\mathbf{Q}(\boldsymbol{\theta})$ is sparse, coming from a Gaussian Markov random fields (GMRFs).

The sparseness can be exploited for very quick computations for the Gaussian part of the model through numerical algorithms for sparse matrices.

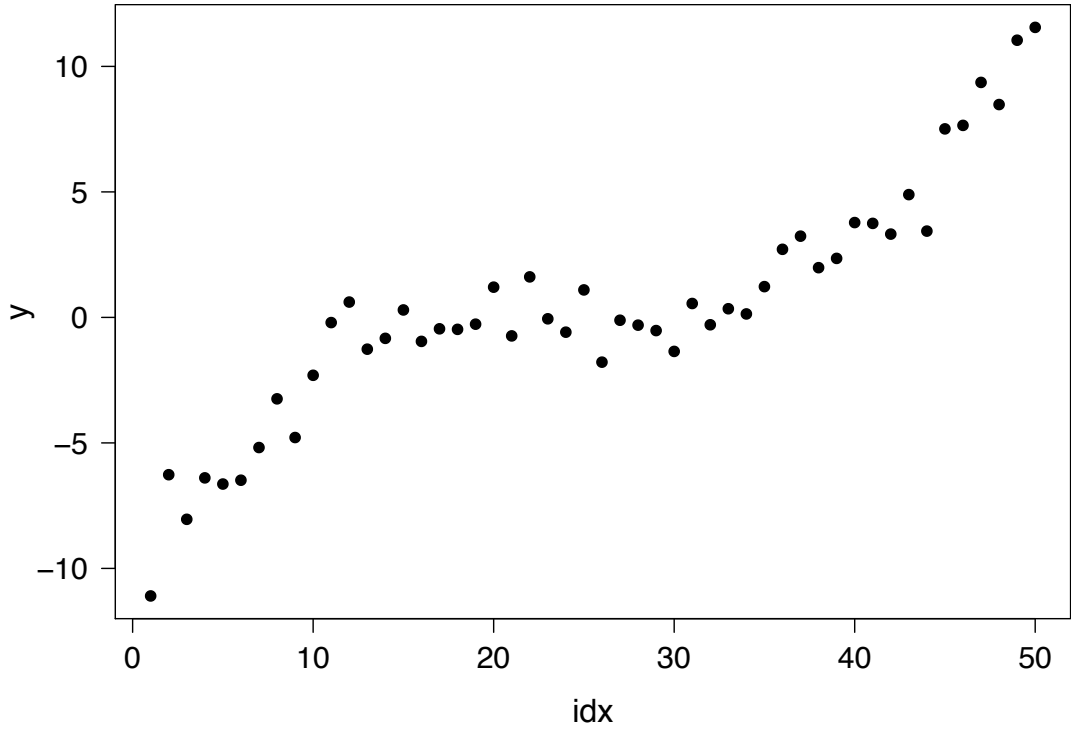
How does INLA work?

Observations

$$y_i = m(i) + \epsilon_i, \qquad i = 1, \ldots, n$$

Here, we assume that $m(i)$ is a smooth function wrt i and $\epsilon_i \overset{\text{iid}}{\sim} \mathcal{N}(0, \tau_0)$ with *known* precision τ_0.

```
1 n = 50
2 idx = 1:n
3 # generate something smooth
      representing m
4 fun = 100*((idx-n/2)/n)^3
5 # add some noise
6 y = fun + rnorm(n, mean=0,
      sd=1)
7 plot(idx, y)
```



Assumed hierarchical model

1. Data: Gaussian observations with known precision

$$y_i \mid x_i, \theta \sim \mathcal{N}(x_i, \tau_0)$$

2. Latent model: A Gaussian model for the smooth function[6]

$$\pi(\boldsymbol{x} \mid \theta) \propto \theta^{(n-2)/2} \exp\left(-\frac{\theta}{2}\sum_{i=3}^{n}(x_i - 2x_{i-1} + x_{i-2})^2\right)$$

3. Hyperparameter: The smoothing parameter θ which we assign a $\Gamma(a, b)$ prior

$$\pi(\theta) \propto \theta^{a-1} \exp(-b\theta), \quad \theta > 0$$

[6] `model="rw2"`

Derivation of posterior marginals (I)

Since

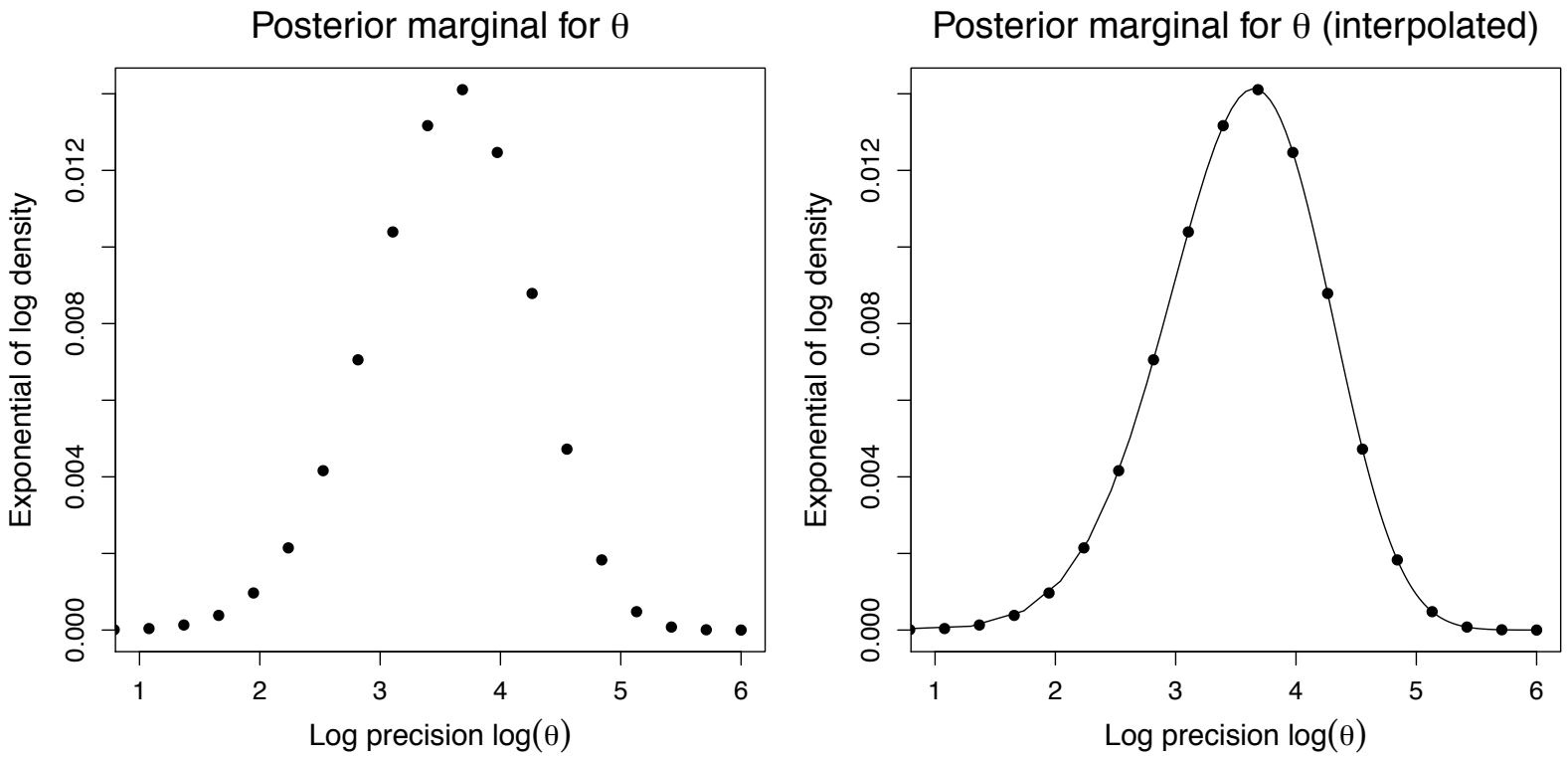
$$\boldsymbol{x}, \boldsymbol{y} \mid \theta \ \sim \ \mathcal{N}(\cdot, \cdot)$$

(derived using $\pi(\boldsymbol{x}, \boldsymbol{y} \mid \theta) \propto \pi(\boldsymbol{y} \mid \boldsymbol{x}, \theta)\, \pi(\boldsymbol{x} \mid \theta)$),
we can compute (numerically) all marginals, using that

$$\pi(\theta \mid \boldsymbol{y}) \propto \frac{\overbrace{\pi(\boldsymbol{x}, \boldsymbol{y} \mid \theta)}^{\text{Gaussian}}\ \pi(\theta)}{\underbrace{\pi(\boldsymbol{x} \mid \boldsymbol{y}, \theta)}_{\text{Gaussian}}}$$

Posterior marginal for hyperparameter

Select a grid of points to represent the density $\theta \mid \boldsymbol{y}$. (Here, the points are chosen to be equi-distant).



Derivation of posterior marginals (II)

From

$$\boldsymbol{x} \mid \boldsymbol{y}, \theta \ \sim \ \mathcal{N}(\cdot, \cdot)$$

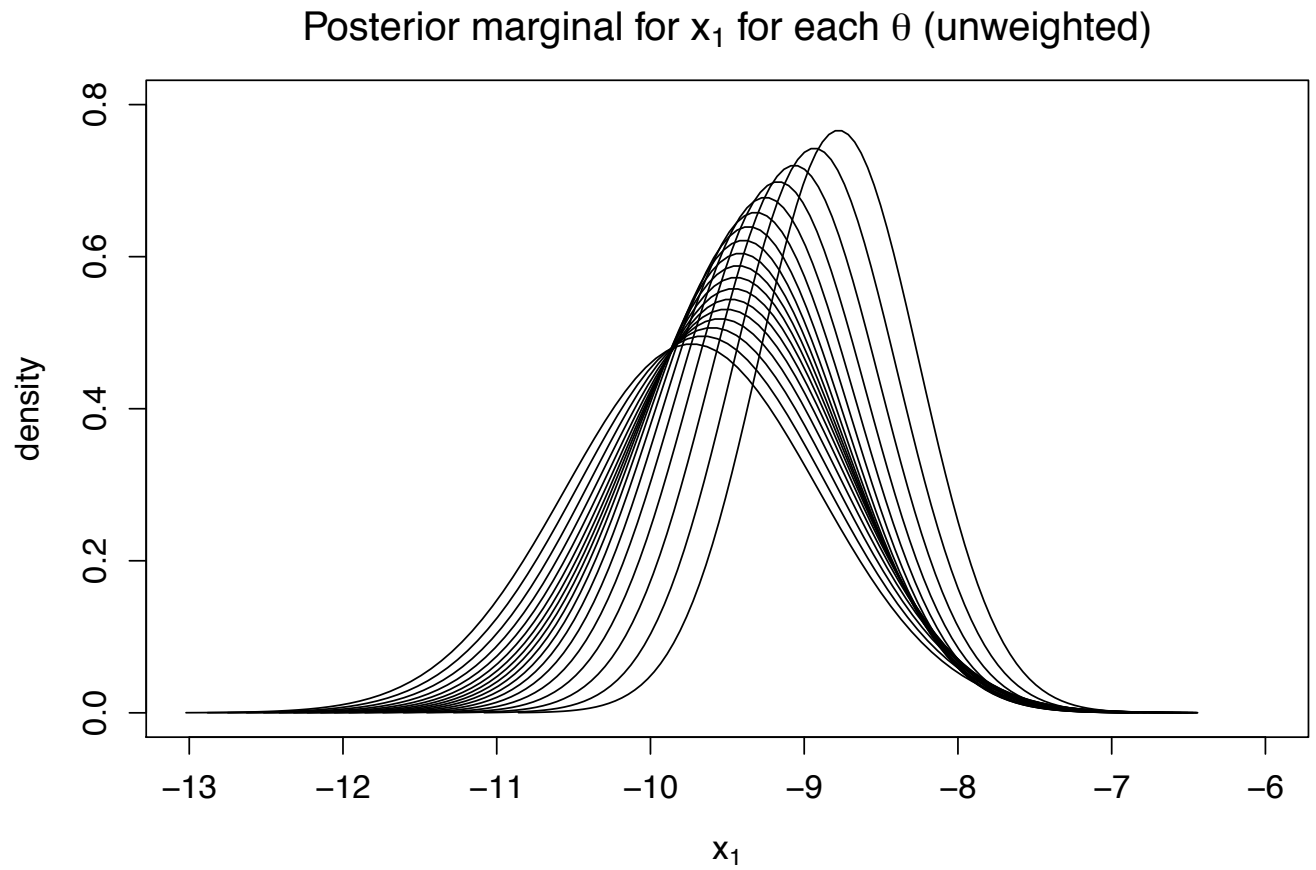
we can compute

$$\begin{aligned} \pi(x_i \mid \boldsymbol{y}) &= \int \underbrace{\pi(x_i \mid \theta, \boldsymbol{y})}_{\text{Gaussian}} \ \pi(\theta \mid \boldsymbol{y}) \ d\theta \\ &\approx \sum_k \pi(x_i \mid \theta_k, \boldsymbol{y}) \pi(\theta_k \mid \boldsymbol{y}) \Delta_k \end{aligned}$$

where θ_k, $k = 1, \ldots, K$, correspond to representative points of $\theta \mid \boldsymbol{y}$ and Δ_k are the corresponding weights.

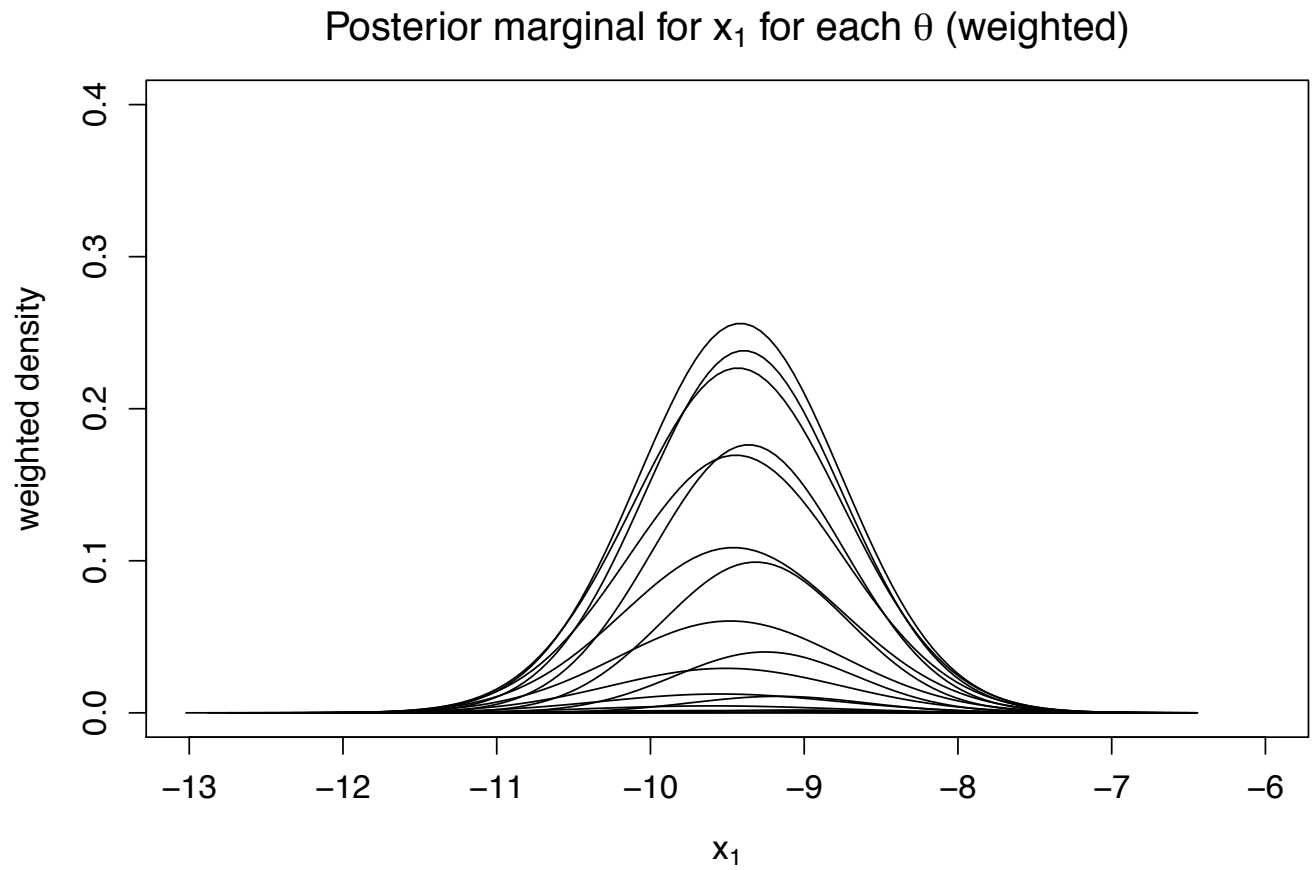
Posterior marginal for latent parameters

Compute the conditional marginal posterior for each x_i given θ_k. Here, shown for x_1.



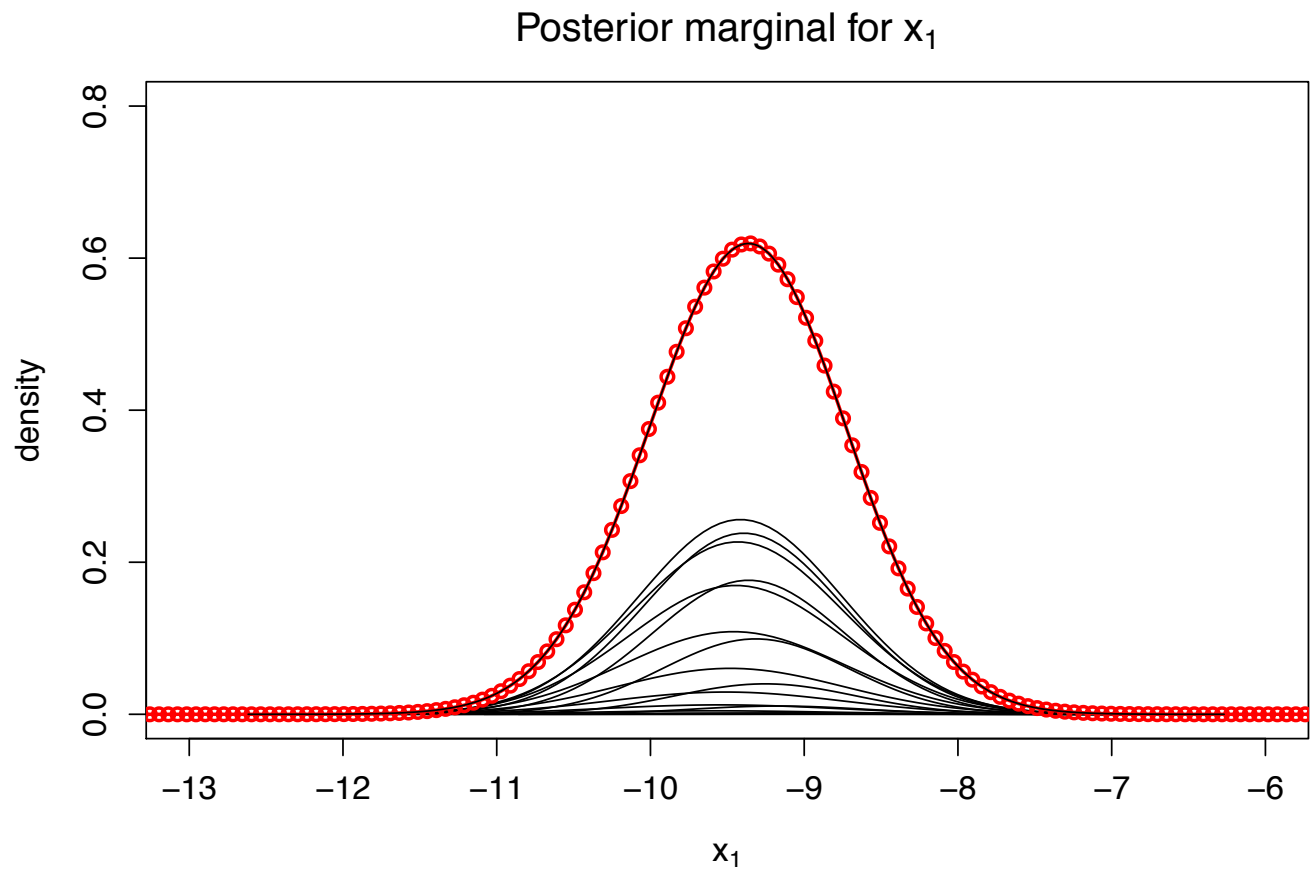
Posterior marginal for latent parameters

Weigh the resulting (conditional) marginal posterior by the density associated with each θ_k.



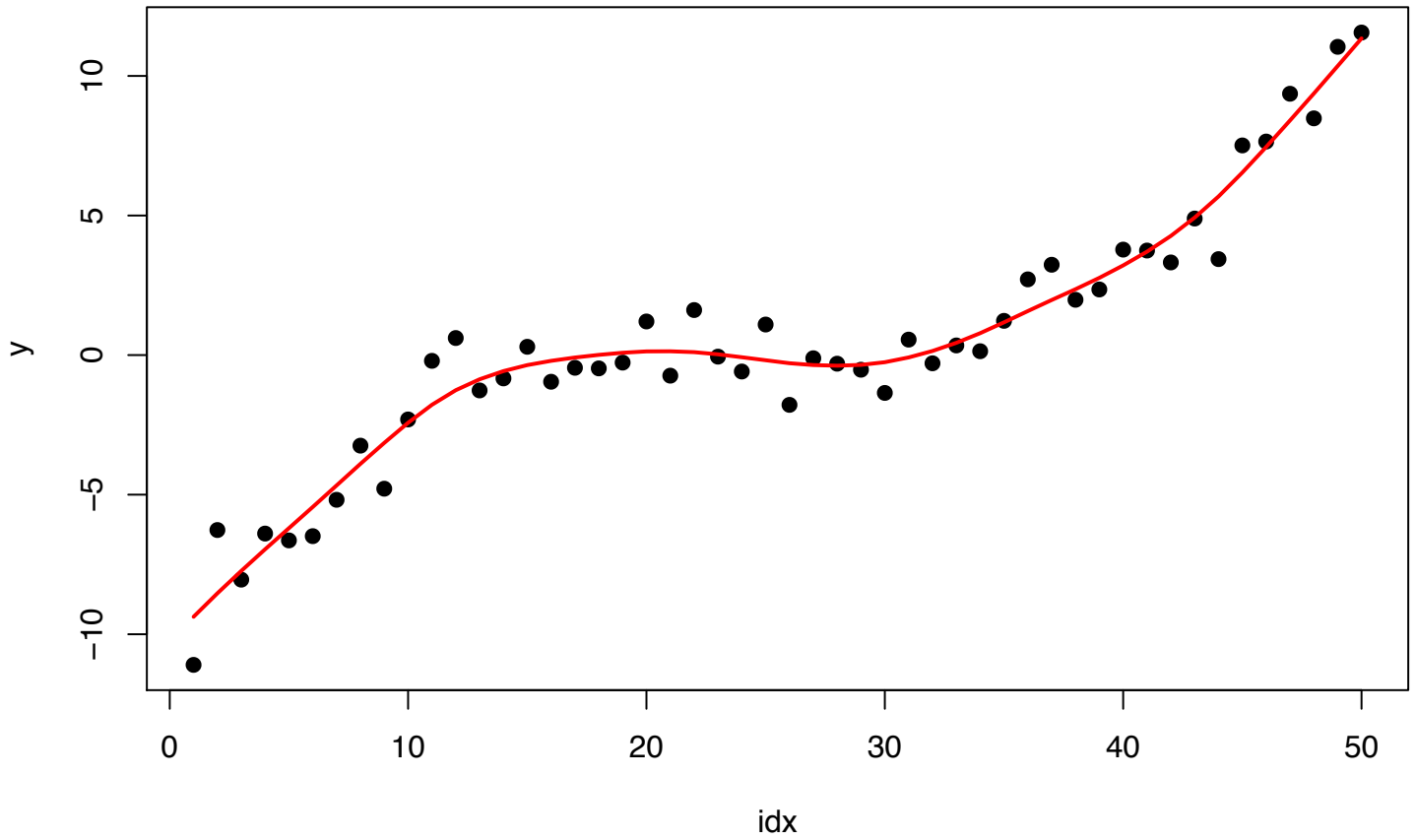
Posterior marginal for latent parameters

Numerically sum over all conditional densities to obtain the posterior marginal for each x_i.



Fitted spline

The posterior marginals are used to calculate summary statistics, like means, variances and credible intervals:



Extensions

This is the basic idea behind INLA. It is quite simple.

However, we need to extend this basic idea so we can deal with

- More than one hyperparameter
- Non-Gaussian observations

How, do things change?

$$\pi(\boldsymbol{\theta} \mid \boldsymbol{y}) \propto \frac{\overbrace{\pi(\boldsymbol{x}, \boldsymbol{y} \mid \boldsymbol{\theta})}^{\text{Non-Gaussian, BUT KNOWN}} \quad \pi(\boldsymbol{\theta})}{\underbrace{\pi(\boldsymbol{x} \mid \boldsymbol{y}, \boldsymbol{\theta})}_{\text{Non-Gaussian and UNKNOWN}}}$$

Extensions

This is the basic idea behind INLA. It is quite simple.

However, we need to extend this basic idea so we can deal with

- More than one hyperparameter
- Non-Gaussian observations

How, do things change?

$$\pi(\boldsymbol{\theta} \mid \boldsymbol{y}) \propto \frac{\overbrace{\pi(\boldsymbol{x}, \boldsymbol{y} \mid \boldsymbol{\theta})}^{\text{Non-Gaussian, BUT KNOWN}} \quad \pi(\boldsymbol{\theta})}{\underbrace{\pi(\boldsymbol{x} \mid \boldsymbol{y}, \boldsymbol{\theta})}_{\text{Non-Gaussian and UNKNOWN}}}$$

Complications... Mostly practical

The non-Gaussian part of the model

- In many cases $\pi(\boldsymbol{x} \mid \boldsymbol{y}, \boldsymbol{\theta})$ is very close to a Gaussian distribution, and can be replaced with a Laplace approximation
- This means that all the really hard, high-dimensional integrals with respect to the latent field are easy, and only the integrals with respect to the hyperparameters remain
- If the number of hyperparameters is low, these integrals can be done efficiently numerically

The GMRF (Laplace) approximation

Let $\boldsymbol{x}$ denote a GMRF with precision matrix $\boldsymbol{Q}$ and mean $\boldsymbol{\mu}$. Approximate

$$\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y}) \propto \exp\left(-\frac{1}{2}\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} + \sum_{i=1}^{n} \log \pi(y_i|x_i)\right)$$

by using a second-order Taylor expansion of $\log \pi(y_i|x_i)$ around $\boldsymbol{\mu}_0$, say.

Recall

$$f(x) \approx f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2}f''(x_0)(x - x_0)^2 = a + bx - \frac{1}{2}cx^2$$

with $b = f'(x_0) - f''(x_0)x_0$ and $c = -f''(x_0)$.

The GMRF approximation (II)

Thus,

$$\tilde{\pi}(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}) \propto \exp\left(-\frac{1}{2}\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} + \sum_{i=1}^{n}(a_i + b_i x_i - 0.5 c_i x_i^2)\right)$$

$$\propto \exp\left(-\frac{1}{2}\boldsymbol{x}^T(\boldsymbol{Q} + \text{diag}(\boldsymbol{c}))\boldsymbol{x} + \boldsymbol{b}^T\boldsymbol{x}\right)$$

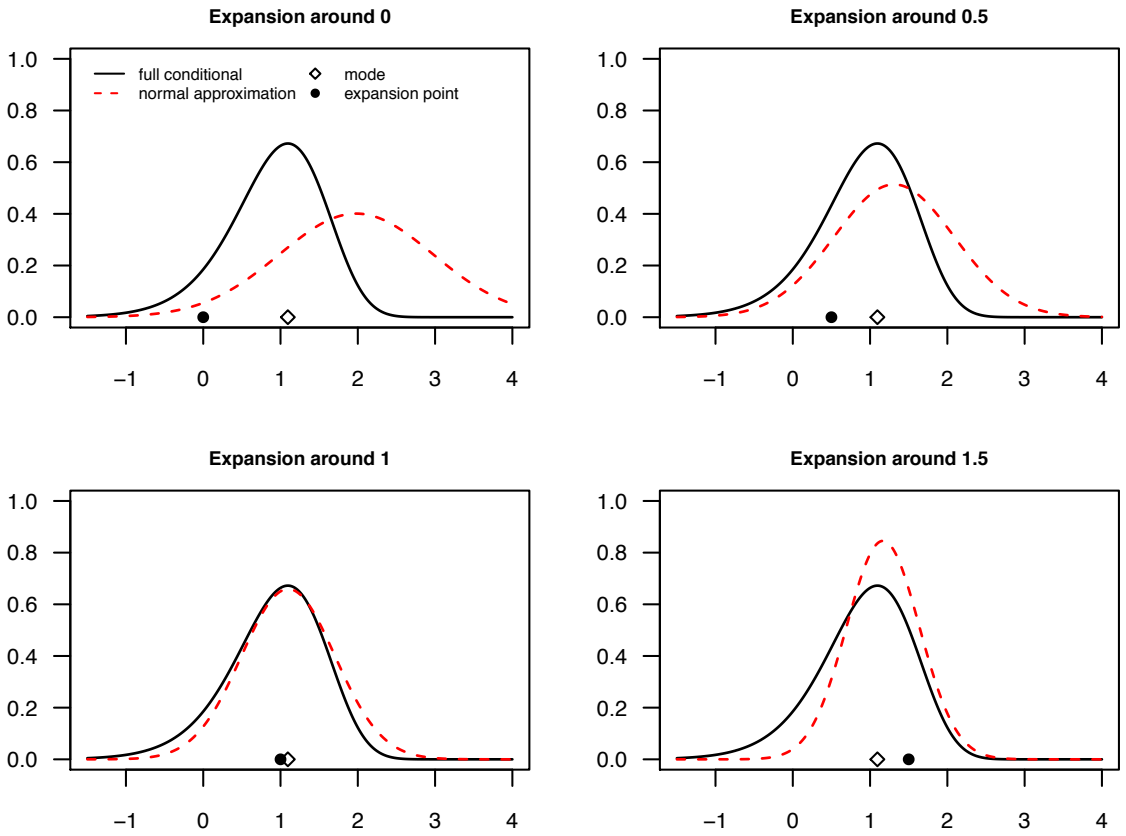
to get a Gaussian approximation with precision matrix $\boldsymbol{Q} + \text{diag}(\boldsymbol{c})$ and mean given by the solution of $(\boldsymbol{Q} + \text{diag}(\boldsymbol{c}))\boldsymbol{\mu} = \boldsymbol{b}$. The canonical parameterisation is

$$\mathcal{N}_C(\mathbf{b}, \boldsymbol{Q} + \text{diag}(\boldsymbol{c}))$$

which corresponds to

$$\mathcal{N}((\boldsymbol{Q} + \text{diag}(\boldsymbol{c}))^{-1}\mathbf{b}, (\boldsymbol{Q} + \text{diag}(\boldsymbol{c}))^{-1}).$$

Illustration



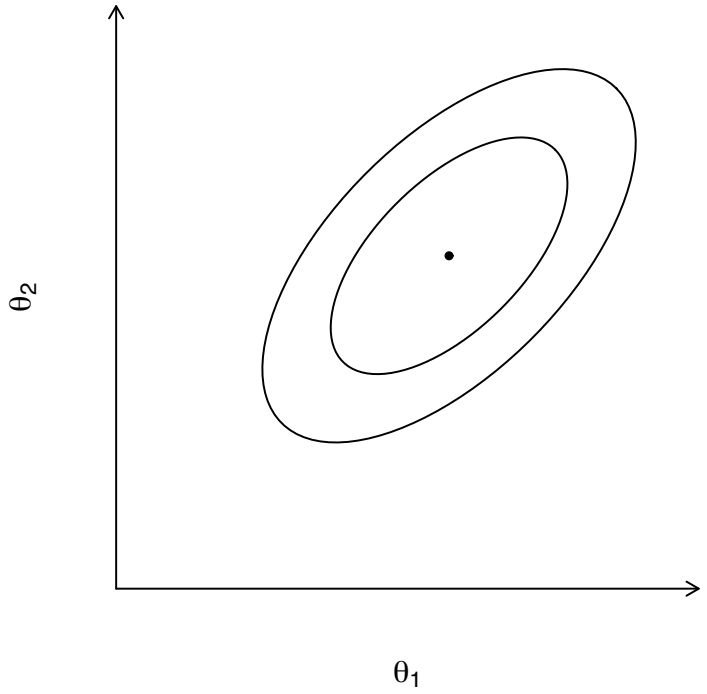
If $\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta}$ is Gaussian "the approximation" is exact!

Exploring $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$

$\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is "numerically explored" to find suitable support points $\boldsymbol{\theta}_k$.

Main use: Select good evaluation points $\boldsymbol{\theta}_k$ for the numerical integration when approximating $\tilde{\pi}(x_i|\boldsymbol{y})$

- Locate the mode

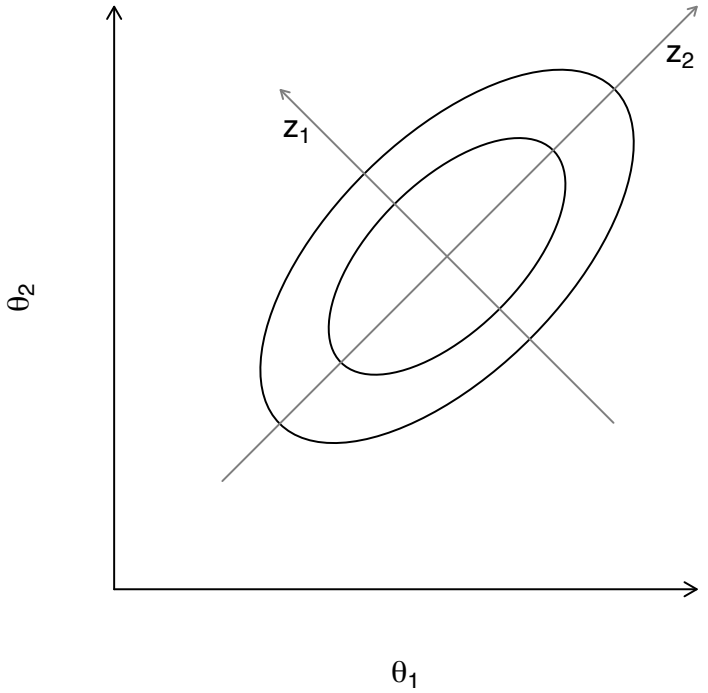


Exploring $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$

$\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is "numerically explored" to find suitable support points $\boldsymbol{\theta}_k$.

Main use: Select good evaluation points $\boldsymbol{\theta}_k$ for the numerical integration when approximating $\tilde{\pi}(x_i|\boldsymbol{y})$

- Locate the mode
- Compute the Hessian to construct principal components

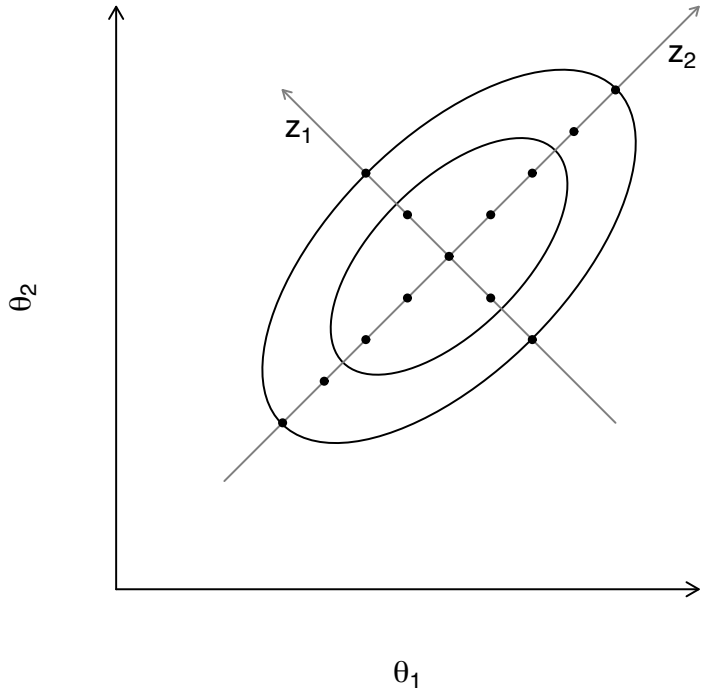


Exploring $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$

$\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is "numerically explored" to find suitable support points $\boldsymbol{\theta}_k$.

Main use: Select good evaluation points $\boldsymbol{\theta}_k$ for the numerical integration when approximating $\tilde{\pi}(x_i|\boldsymbol{y})$

- Locate the mode
- Compute the Hessian to construct principal components
- Grid-search to locate bulk of the probability mass

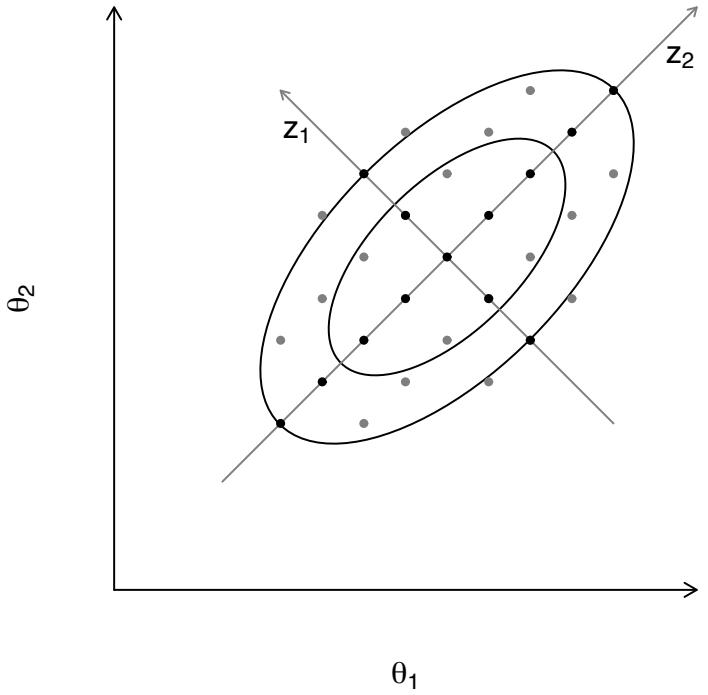


Exploring $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$

$\widetilde{\pi}(\boldsymbol{\theta}|\boldsymbol{y})$ is "numerically explored" to find suitable support points $\boldsymbol{\theta}_k$.

Main use: Select good evaluation points $\boldsymbol{\theta}_k$ for the numerical integration when approximating $\tilde{\pi}(x_i|\boldsymbol{y})$

- Locate the mode
- Compute the Hessian to construct principal components
- Grid-search to locate bulk of the probability mass



All points found have equal area weight Δ_k.

Approximating $\pi(x_i|\boldsymbol{\theta}, \boldsymbol{y})$

For approximating the first component $\pi(x_i|\boldsymbol{\theta}, \boldsymbol{y})$ we can use

- a Gaussian approximation, easily extractable from $\widetilde{\pi}_G(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$. However, errors in location and/or lack of skewness possible.
- a Laplace approximation

$$\tilde{\pi}_{\mathrm{LA}}(x_i|\boldsymbol{\theta}, \boldsymbol{y}) \propto \left. \frac{\pi(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{y})}{\tilde{\pi}_{\mathrm{GG}}(\boldsymbol{x}_{-i}|x_i, \boldsymbol{\theta}, \boldsymbol{y})} \right|_{\boldsymbol{x}_{-i}=\boldsymbol{x}^{\star}_{-i}(x_i,\boldsymbol{\theta})}.$$

  The approximation is very accurate but very expensive.
- a simplified Laplace approximation based on fitting a skew-normal distribution to a series expansion of $\tilde{\pi}_{LA}$.

INLA: Overview

Step I Approximate $\pi(\boldsymbol{\theta}|y)$ using the Laplace approximation and select good evaluation points $\boldsymbol{\theta}_k$.

Step II For each $\boldsymbol{\theta}_k$ and i approximate $\pi(x_i|\boldsymbol{\theta}_k, \boldsymbol{y})$ using the Laplace or simplified Laplace approximation for selected values of x_i .

Step III For each i, sum out $\boldsymbol{\theta}_k$

$$\tilde{\pi}(x_i|\boldsymbol{y}) = \sum_k \tilde{\pi}(x_i|\boldsymbol{\theta}_k, \boldsymbol{y}) \times \tilde{\pi}(\boldsymbol{\theta}_k|\boldsymbol{y}) \times \Delta_k.$$

Build a log spline corrected Gaussian to represent $\tilde{\pi}(x_i|\boldsymbol{y})$.

Limitations

- The dimension of the latent field $\boldsymbol{x}$ can be large (10^2–10^6)
- But the dimension of the hyperparameters $\boldsymbol{\theta}$ must be small (≤ 9)

In other words, each random effect can be big, but there cannot be too many random effects unless they share parameters.