Besag model for spatial effects

Parametrization

The besag model for random vector $\mathbf{x} = (x_1, \ldots, x_n)$ is defined as

$$x_i | x_j, i \neq j, \tau \sim \mathcal{N}(\frac{1}{n_i} \sum_{i \sim j} x_j, \frac{1}{n_i \tau}) \tag{1}$$

where n_i is the number of neighbours of node i, $i \sim j$ indicates that the two nodes i and j are neighbours.

Hyperparameters

The precision parameter τ is represented as

$$\theta_1 = \log \tau$$

and the prior is defined on θ_1.

Specification

The besag model is specified inside the `f()` function as

```
f(<whatever>,model="besag",graph=<graph>,
  hyper=<hyper>, adjust.for.con.comp = TRUE,
  scale.model = FALSE)
```

The neighbourhood structure of **x** is passed to the program through the `graph` argument.

If the option `adjust.for.con.comp=TRUE` then the model is adjusted if the graph has more than one connected compoment. This adjustment can be disabled setting this option to `FALSE`. If `adjust.for.con.comp=TRUE` then `constr=TRUE` is interpreted as a sum-to-zero constraint on *each* connected component in the graph and the `rankdef` parameter is set to the number of connected components.

The logical option `scale.model` determine if the model should be scaled to have an average variance (the diagonal of the generalized inverse) equal to 1. This makes prior spesification much easier. Default is `FALSE` so that the model is not scaled.

Hyperparameter spesification and default values

doc `The Besag area model (CAR-model)`

hyper

theta

hyperid `8001`
name `log precision`
short.name `prec`
prior `loggamma`
param `1 5e-05`
initial `4`
fixed `FALSE`
to.theta `function(x) log(x)`
from.theta `function(x) exp(x)`

constr `TRUE`

nrow.ncol `FALSE`

augmented `FALSE`

aug.factor `1`

aug.constr

n.div.by

n.required `TRUE`

set.default.values `TRUE`

pdf `besag`

Example

For examples of application of this model see the `Bym`, `Munich`, `Zambia` or `Scotland` examples in Volume I.

Details on the implementation

This gives some details of the implementation, which depends on the following variables

nc1 Number of connected components in the graph with size 1. These nodes, *singletons*, have no neighbours.

nc2 Number of connected components in the graph with size ≥ 2.

scale.model The value of the logical flag, if the model should be scaled or not. (Default FALSE)

adjust.for.con.comp The value of the logical flag if the `constr=TRUE` option should be reinterpreted.

The case `(scale.model==FALSE && adjust.for.con.comp == FALSE)`

The option `constr=TRUE` is interpreted as a sum-to-zero constraint over the whole graph. Singletons are given a uniform distribution on $(-\infty, \infty)$ before the constraint.

The case `(scale.model==TRUE && adjust.for.con.comp == FALSE)`

The option `constr=TRUE` is interpreted as a sum-to-zero constraint over the whole graph. Let $Q = \tau R$ be the standard precision matrix from the `besag`-model with precision parameter τ. Then R, except the singletons, are scaled so that the geometric mean of the marginal variances is 1, and R is modified so that singletons have a standard Normal distribution.

The case `(scale.model==FALSE && adjust.for.con.comp == TRUE)`

The option `constr=TRUE` is interpreted as one sum-to-zero constraint over each of the `nc2` connected components of size ≥ 2. Singletons are given a uniform distribution on $(-\infty, \infty)$.

The case `(scale.model==TRUE && adjust.for.con.comp == TRUE)`

The option `constr=TRUE` is interpreted as `nc2` sum-to-zero constraints for each of the connected components of size ≥ 2. Let $Q = \tau R$ be the standard precision matrix from the `besag`-model with precision parameter τ. Then R, are scaled so that the geometric mean of the marginal variances in each connected component of size ≥ 2 is 1, and modified so that singletons have a standard Normal distribution.

Notes

The term $\frac{1}{2}\log(|R|^*)$ of the normalisation constant is not computed, hence you need to add this part to the log marginal likelihood estimate, if you need it. Here R is the precision matrix with a unit precision parameter.