

Random walk model of order 1 (RW1)

Parametrization

The random walk model of order 1 (RW1) for the Gaussian vector $\mathbf{x} = (x_1, \dots, x_n)$ is constructed assuming independent increments:

$$\Delta x_i = x_i - x_{i+1} \sim \mathcal{N}(0, \tau^{-1})$$

The density for \mathbf{x} is derived from its $n - 1$ increments as

$$\begin{aligned}\pi(\mathbf{x}|\tau) &\propto \tau^{(n-1)/2} \exp \left\{ -\frac{\tau}{2} \sum (\Delta x_i)^2 \right\} \\ &= \tau^{(n-1)/2} \exp \left\{ -\frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} \right\}\end{aligned}$$

where $\mathbf{Q} = \tau \mathbf{R}$ and \mathbf{R} is the structure matrix reflecting the neighbourhood structure of the model.

It is also possible to define a *cyclic* version of the RW1 model, in this case the graph is modified so that last node x_n is neighbour of x_{n-1} and x_1 .

Hyperparameters

The precision parameter τ is represented as

$$\theta = \log \tau$$

and the prior is defined on θ .

Specification

The RW1 model is specified inside the `f()` function as

```
f(<whatever>, model="rw1", values=<values>, cyclic=<TRUE|FALSE>,  
  hyper = <hyper>, scale.model = FALSE)
```

The (optional) argument `values` is a numeric or factor vector giving the values assumed by the covariate for which we want the effect to be estimated. See next example for an application.

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

hyper

theta

hyperid 4001

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 FALSE
set.default.values FALSE
pdf rw1

```

Example

```

n=100
z=seq(0,6,length.out=n)
y=sin(z)+rnorm(n,mean=0,sd=0.5)
data=data.frame(y=y,z=z)

formula=y~f(z,model="rw1",
            hyper = list(prec = list(prior="loggamma",param=c(1,0.01))))
result=inla(formula,data=data,family="gaussian")

#here we estimate the effect only for some of the values in z
formula1=y~f(z,model="rw1",
            hyper = list(prec = list(prior="loggamma",param=c(1,0.01))))
result1=inla(formula1,data=data,family="gaussian")

```

Notes

- The RW1 is intrinsic with rank deficiency 1.
- The RW1 model for irregular locations are supported although not described here.
- 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.