Generic0 model

Parametrization

The Type 0 generic model implements the following precision matrix

$$\mathbf{Q} = \tau \mathbf{C}$$

where $\mathbf{C}$ is the structure matrix.

Hyperparameters

The precision parameters of the generic0 model is represented as

$$\theta = \log(\tau)$$

and prior is assigned to θ

Specification

The generic0 models is specified inside the `f()` function as

```
f(<whatever>,model="generic0",Cmatrix = <Cmat>, hyper = <hyper>)
```

where `<Cmat>` can be given in two different ways:

- a dense matrix or a sparse-matrix defined be `Matrix::sparseMatrix()`.
- the name of a file giving the structure matrix. The file should have the following format

$$i \quad j \quad \mathbf{C}_{ij}$$

  where i and j are the row and column index and $\mathbf{C}_{ij}$ is the corresponding element of the precision matrix. Only the non-zero elements of the precision matrix need to be stored in the file.

See the following example for an application

Hyperparameter spesification and default values

doc `A generic model (type 0)`

hyper

theta

hyperid `18001`
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 `FALSE`

nrow.ncol `FALSE`

augmented `FALSE`

aug.factor `1`

aug.constr

n.div.by

n.required `TRUE`

set.default.values `TRUE`

pdf `generic0`

Example

In the example below we define a RW1 model first using the `generic0` model and this using the `rw1` model.

```
## Simulate data
n=100
z=1:n
y=sin(z/n*2*pi)+rnorm(n,mean=0,sd=0.5)
data=data.frame(y=y,z=z)

Q = toeplitz(c(2,-1, rep(0,n-3),-1))
Q[1,1] = Q[n,n] = 1
Q[n,1] = Q[1,n] = 0

## Q as dense
formula1 = y ~ f(z, model="generic0", Cmatrix = Q,
                 rankdef=1, constr=TRUE, diagonal=1e-05)
result1 = inla(formula1, data=data, family="gaussian")

## Q as sparse
Q.sparse = as(Q, "dgTMatrix")
formula2 = y ~ f(z, model="generic0", Cmatrix = Q.sparse,
                 rankdef=1, constr=TRUE, diagonal=1e-05)
result2 = inla(formula2, data=data, family="gaussian")

## This is the same model defined using the rw1 model
formula3 = y ~ f(z,model="rw1")
result3 = inla(formula3, data=data, family="gaussian")
```

Notes

INLA uses for this model the following normalizing constant

$$\tau^{n/2} \left(\frac{1}{2\pi}\right)^{n/2}$$

where n is the dimension of the C matrix, and NOT the correct one

$$\tau^{n/2} \left(\frac{1}{2\pi}\right)^{n/2} \mid C \mid^{1/2} .$$

Different algorithms are required to compute the determinant depending on the structure and size of C (and it might be singular), and therefore this constant is not computed. However, for near-all use of this `generic0` model, this constant is not of interest.

The missing constant *only matters* for the marginal likelihood value. Say you are comparing two runs with two models, one where this generic-component is present, and one where this generic-component is not present. Since the marginal likelihood does depend on the normalising constant not only the on the "shape", then a comparison between the two models will be wrong using the reported `result$mlik`-values from INLA. You have to add to one of them

$$1/2 \log(|C|)$$

to account for this missing constant.