

## Autoregressive model of order 1 (AR1)

### Parametrization

The autoregressive model of order 1 (AR1) for the Gaussian vector  $\mathbf{x} = (x_1, \dots, x_n)$  is defined as:

$$\begin{aligned}x_1 &\sim \mathcal{N}(0, (\tau(1 - \rho^2))^{-1}) \\x_i &= \rho x_{i-1} + \epsilon_i; \quad \epsilon_i \sim \mathcal{N}(0, \tau^{-1}) \quad i = 2, \dots, n\end{aligned}$$

where

$$|\rho| < 1$$

### Hyperparameters

The precision parameter  $\kappa$  is represented as

$$\theta_1 = \log(\kappa)$$

where  $\kappa$  is the *marginal* precision,

$$\kappa = \tau(1 - \rho^2).$$

The parameter  $\rho$  is represented as

$$\theta_2 = \log\left(\frac{1 + \rho}{1 - \rho}\right)$$

and the prior is defined on  $\theta = (\theta_1, \theta_2)$ .

### Specification

The AR1 model is specified inside the `f()` function as

```
f(<whatever>, model="ar1", values=<values>, hyper = <hyper>)
```

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 the example for RW1 for an application.

### Hyperparameter specification and default values

**hyper**

**theta1**

```
hyperid 14001
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)
```

**theta2**

```
hyperid 14002
name logit lag one correlation
short.name rho
```

```

    prior normal
    param 0 0.15
    initial 2
    fixed FALSE
    to.theta function(x) log((1+x)/(1-x))
    from.theta function(x) 2*exp(x)/(1+exp(x))-1

constr FALSE

nrow.ncol FALSE

augmented FALSE

aug.factor 1

aug.constr

n.div.by

n.required FALSE

set.default.values FALSE

pdf ar1

```

## Example

In this exaple we implement an ar1 model observed with Poisson counts

```

#simulate data
n = 100
rho = 0.8
prec = 10
## note that the marginal precision would be
marg.prec = prec * (1-rho^2)

E=sample(c(5,4,10,12),size=n,replace=T)
eta = as.vector(arima.sim(list(order = c(1,0,0), ar = rho), n = n,sd=sqrt(1/prec)))
y=rpois(n,E*exp(eta))
data = list(y=y, z=1:n, E=E)

## fit the model
formula = y~f(z,model="ar1")
result = inla(formula,family="poisson", data = data, E=E)

```

## Notes

None