# LogLogistic

## Parametrisation

The LogLogistic distribution has cumulative distribution function

$$F(y) = \frac{1}{1 + (y/\beta)^{-\alpha}}$$

where

$\alpha > 0$ is the shape parameter.

$\beta > 0$ is the scale parameter.

## Link-function

The parameter $\beta$ is linked to the linear predictor as:

$$\beta = \exp(\eta)$$

## Hyperparameters

The $\alpha$ parameter is represented as

$$\theta = \log \alpha$$

and the prior is defined on $\theta$.

## Specification

- family = `loglogistic`

- Required arguments: $y$ (to be given in a format by using `inla.surv()` function )

**Hyperparameter spesification and default values**

**hyper**

    **theta**

        **name**   log alpha
        **short.name**   alpha
        **initial**   1
        **fixed**   FALSE
        **prior**   loggamma
        **param**   25 25
        **to.theta**   function(x) log(x)
        **from.theta**   function(x) exp(x)

**survival**   TRUE

**discrete**   FALSE

**link**   default log

**pdf**   loglogistic

## Example

In the following example we estimate the parameters in a simulated case

```
rloglogistic = function(n,  beta,  alpha = 1)
{
    p = runif(n)
    return (beta* (((1-p)/p)^(-1/alpha)))
}


n = 1000
alpha = 2
x = runif(n)
eta = 1+x
beta = exp(eta)
y = rloglogistic(n, beta = beta, alpha = alpha)
event = rep(1,n)
data = list(y=y, event=event, x=x)
formula=inla.surv(y,event) ~ x
r=inla(formula, family ="loglogistic", data=data, verbose=T)
```

## Notes

- Loglogistic model can be used for right censored, left censored, interval censored data.

- A general frame work to represent time is given by `inla.surv`

- If the observed times $y$ are large/huge, then this can cause numerical overflow in the likelihood routines giving error messages like

```
        file: smtp-taucs.c  hgid: 891deb69ae0c  date: Tue Nov 09 22:34:28 2010 +0100
        Function: GMRFLib_build_sparse_matrix_TAUCS(), Line: 611, Thread: 0
        Variable evaluates to NAN/INF. This does not make sense. Abort...
```

  If you encounter this problem, try to scale the observatios, `time = time / max(time)` or similar, before running `inla()`.