The Gamma-distribution

Parametrisation

The Gamma-distribution has the following density

$$\pi(y) = \frac{b^a}{\Gamma(a)} y^{a-1} \exp(-by), \qquad a > 0, \quad b > 0, \quad y > 0,$$

where $E(y) = \mu = a/b$ and $\text{Var}(y) = 1/\tau = a/b^2$, where τ is the precision and μ is the mean. We will use the following parameterisation for the precision

$$\tau = (s\phi)/\mu^2$$

where ϕ is the precision parameter (or $1/\phi$ is the dispersion parameter) and $s > 0$ is a fixed scaling (for the regression model), which gives this density

$$\pi(y) = \frac{1}{\Gamma(s\phi)} \left(\frac{(s\phi)}{\mu} \right)^{(s\phi)} y^{(s\phi)-1} \exp\left(-(s\phi)\frac{y}{\mu} \right)$$

Link-function

The linear predictor η is linked to the mean μ using a default log-link

$$\mu = \exp(\eta)$$

Hyperparameter

The hyperparameter is the precision parameter ϕ, which is represented as

$$\phi = \exp(\theta)$$

and the prior is defined on θ.

Specification

- `family="gamma"` for regression models and `family="gamma.surv"` for survival models.

- Required arguments: for `gamma.surv`, y (to be given in a format by using `inla.surv()`), and for `gamma`, y and s (default value 1).

The scalings s is **not** used for `family="gamma.surv"`.

Hyperparameter spesification and default values

doc `The Gamma likelihood`

hyper

    theta

        hyperid `58001`
        name `precision parameter`
        short.name `prec`
        output.name `Precision-parameter for the Gamma observations`
        output.name.intern `Intern precision-parameter for the Gamma observations`

        initial `4.60517018598809`

        fixed `FALSE`

        prior `loggamma`

        param `1 0.01`

        to.theta `function(x) log(x)`

        from.theta `function(x) exp(x)`

survival `FALSE`

discrete `FALSE`

link `default log quantile`

pdf `gamma`

doc `The Gamma likelihood (survival)`

hyper

    theta1

        hyperid `58101`

        name `precision parameter`

        short.name `prec`

        output.name `Precision-parameter for the Gamma surv observations`

        output.name.intern `Intern precision-parameter for the Gamma surv observations`

        initial `0`

        fixed `FALSE`

        prior `loggamma`

        param `1 0.01`

        to.theta `function(x) log(x)`

        from.theta `function(x) exp(x)`

    theta2

        hyperid `58102`

        name `beta1`

        short.name `beta1`

        output.name `beta1 for Gamma-Cure`

        output.name.intern `beta1 for Gamma-Cure`

        initial `-7`

        fixed `FALSE`

        prior `normal`

        param `-4 100`

        to.theta `function(x) x`

        from.theta `function(x) x`

    theta3

        hyperid `58103`

        name `beta2`

        short.name `beta2`

        output.name `beta2 for Gamma-Cure`

output.name.intern beta2 for Gamma-Cure

initial 0

fixed FALSE

prior normal

param 0 100

to.theta function(x) x

from.theta function(x) x

theta4

    hyperid 58104

    name beta3

    short.name beta3

    output.name beta3 for Gamma-Cure

    output.name.intern beta3 for Gamma-Cure

    initial 0

    fixed FALSE

    prior normal

    param 0 100

    to.theta function(x) x

    from.theta function(x) x

theta5

    hyperid 58105

    name beta4

    short.name beta4

    output.name beta4 for Ga mma-Cure

    output.name.intern beta4 for Gamma-Cure

    initial 0

    fixed FALSE

    prior normal

    param 0 100

    to.theta function(x) x

    from.theta function(x) x

theta6

    hyperid 58106

    name beta5

    short.name beta5

    output.name beta5 for Gamma-Cure

    output.name.intern beta5 for Gamma-Cure

    initial 0

    fixed FALSE

    prior normal

    param 0 100

    to.theta function(x) x

    from.theta function(x) x

theta7

hyperid 58107
name beta6
short.name beta6
output.name beta6 for Gamma-Cure
output.name.intern beta6 for Gamma-Cure
initial 0
fixed FALSE
prior normal
param 0 100
to.theta function(x) x
from.theta function(x) x

theta8

hyperid 58108
name beta7
short.name beta7
output.name beta7 for Gamma-Cure
output.name.intern beta7 for Gamma-Cure
initial 0
fixed FALSE
prior normal
param 0 100
to.theta function(x) x
from.theta function(x) x

theta9

hyperid 58109
name beta8
short.name beta8
output.name beta8 for Gamma-Cure
output.name.intern beta8 for Gamma-Cure
initial 0
fixed FALSE
prior normal
param 0 100
to.theta function(x) x
from.theta function(x) x

theta10

hyperid 58110
name beta9
short.name beta9
output.name beta9 for Gamma-Cure
output.name.intern beta9 for Gamma-Cure
initial 0
fixed FALSE
prior normal

> **param** `0 100`
>
> **to.theta** `function(x) x`
>
> **from.theta** `function(x) x`

theta11

> **hyperid** `58111`
>
> **name** `beta10`
>
> **short.name** `beta10`
>
> **output.name** `beta10 for Gamma-Cure`
>
> **output.name.intern** `beta10 for Gamma-Cure`
>
> **initial** `0`
>
> **fixed** `FALSE`
>
> **prior** `normal`
>
> **param** `0 100`
>
> **to.theta** `function(x) x`
>
> **from.theta** `function(x) x`

survival `TRUE`

discrete `FALSE`

link `default log neglog quantile`

pdf `gammasurv`

Example 1

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

```
n = 1000
x = rnorm(n)
eta = 1 + x
mu = exp(eta)
prec.scale = runif(n, min = 0.5, max = 2)
prec.par = 1.2
a = prec.par * prec.scale
b = mu / (prec.par * prec.scale)
y = rgamma(n, shape = a, scale = b)
r = inla(y ~ 1 + x,  data = data.frame(y, x),
        scale = prec.scale, family = "gamma")
```

Example 2

This is an example using the quantile link.

```
n <- 10^4
phi <- 3
a <- phi
## mu <-  phi / rate
alpha <- 0.85
log.mu <- seq(-5, 5, by = 0.01)
```

```
## if exp(eta) is the alpha-quantile,  what is then the mu?
log.q <- log(qgamma(alpha, shape = phi, rate =  phi / exp(log.mu)))
fun <- splinefun(log.q, log.mu)

x <- rnorm(n, sd = 0.3)
eta <- 2 + x
mu <- exp(fun(eta))

## just a check
head(cbind(eta, log(qgamma(alpha, shape = phi, rate = phi / mu))))

y <- rgamma(n, shape = phi, rate = phi / mu)
r <- inla(y ~ 1 + x,
          family = "gamma",
          control.family = list(control.link = list(model = "quantile",
                                                     quantile = alpha)),
          data = data.frame(y, x),
          control.fixed = list(prec.intercept = 1, prec = 1),
          safe = FALSE,
          verbose = TRUE)
summary(r)

cbind(estimate = c(r$summary.fixed$mean, r$summary.hyperpar$mean),
      true = c(2, 1, phi))
```

Notes

None.