Censored Negative Binomial (version2)

Parametrisation

The negative Binomial distribution is

$$Prob(y) = \frac{\Gamma(y+n)}{\Gamma(n)\Gamma(y+1)} p^n (1-p)^y$$

for responses $y = 0, 1, 2, \ldots$, where

n: number of successful trials (size), or dispersion parameter. Must be strictly positive, need not be integer.

p: probability of success in each trial.

The cencoring is that response values in the interval $L \leq y \leq H$ are cencored (and reported as a value in the interval, like y = L, say), whereas other values are reported as is.

This function is the nbinomial version of cenpoisson2.

Link-function

The mean and variance of (the uncensored) y are given as

$$\mu = n \frac{1-p}{p}$$
 and $\sigma^2 = \mu(1 + \frac{\mu}{n})$

and the mean is linked to the linear predictor by

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

where the hyperparameter n (size) plays the role of an dispersion parameter. E represents knows constant and $\log(E)$ is the offset of η .

Hyperparameters

The default parameterization (variant=0) is that the dispersion parameter n (size) is represented as

$$n = \exp(\theta)$$

and the prior is defined on θ .

If variant=1, then dispersion parameter n (size) is represented as

$$n = E \exp(\theta)$$

and the prior is defined on θ .

If variant=2, then dispersion parameter n (size) is represented as

$$n = S \exp(\theta)$$

where S is scale, and the prior is defined on θ .

Specification

- family="nbinomial"
- Required arguments: (y, L, H) [see below] and E (default E = 1) and scale (default scale=1)
- Chose variant with either control.family = list(variant=0) (default) or control.family = list(variant=1) or control.family = list(variant=2).
- Data and cencoring information is given by the triplet: y, L and H, and must be defined as a inla.mdata-object. L and H are vectors of same length as y giving the cencoring information for each observation. L and H must be integer valued or Inf.

```
If y[i] is not in the interval, then the observation is not cencored.
```

```
L[i] = Inf and/or H[i] = Inf is equivalent to L[i] = -1 and/or H[i] = -1.
```

L[i] = Inf implies no interval censoring. H[i] = Inf and $0 \le L[i] < \infty$ implies right censoring.

Hyperparameter spesification and default values

from.theta function(x) exp(x)

```
{
m doc} The CenNegBinomial2 likelihood (similar to cenpoisson2)
```

hyper

```
theta
```

hyperid 63101 name size

```
short.name size
output.name size for the cennbinomial2 observations (1/overdispersion)
output.name.intern log size for the cennbinomial2 observations (1/overdispersion)
initial 2.30258509299405
fixed FALSE
prior pc.mgamma
param 7
to.theta function(x) log(x)
```

survival FALSE

discrete TRUE

link default log logoffset quantile

pdf cennbinomial2

Example

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

```
n = 300
x = rnorm(n, sd = 0.2)
eta = 1 + x
E = runif(n, min = 0.5, max=2)
S = runif(n, min = 0.5, max=2)
## variant 0
```

```
mu = E * exp(eta)
size = 1
prob = size/(size + mu)
y = rnbinom(n, size, mu=mu)
y.low <- sample(c(1:4, Inf), n, replace = TRUE)</pre>
y.high <- y.low + sample(c(1:5, Inf), n, replace = TRUE)
Y <- inla.mdata(cbind(y, y.low, y.high))
r = inla(Y ~1 + x,
         data = list(Y = Y, x = x, E = E),
         family = "cennbinomial2",
         control.family = list(variant = 0),
         E=E, scale = S)
summary(r)
censored = which(y >= y.low & y <= y.high)</pre>
print(length(censored)/ n)
## variant 1
mu = E * exp(eta)
size = 1 * E
prob = size/(size + mu)
y = rnbinom(n, size, mu=mu)
y.low <- sample(c(1:4, Inf), n, replace = TRUE)
y.high <- y.low + sample(c(1:5, Inf), n, replace = TRUE)
Y <- inla.mdata(cbind(y, y.low, y.high))
r = inla(Y ~1 + x,
         data = list(Y = Y, x = x, E = E),
         family = "cennbinomial2",
         control.family = list(variant = 1),
         E=E)
summary(r)
censored = which(y >= y.low & y <= y.high)</pre>
print(length(censored)/ n)
## variant 2
mu = E * exp(eta)
size = 1 * S
prob = size/(size + mu)
y = rnbinom(n, size, mu=mu)
y.low <- sample(c(1:4, Inf), n, replace = TRUE)</pre>
y.high \leftarrow y.low + sample(c(1:5, Inf), n, replace = TRUE)
Y <- inla.mdata(cbind(y, y.low, y.high))
r = inla(Y ~1 + x,
         data = list(Y = Y, x = x, E = E, S = S),
         family = "cennbinomial2",
         control.family = list(variant = 2),
         E=E, scale = S)
summary(r)
censored = which(y >= y.low & y <= y.high)</pre>
print(length(censored)/ n)
```

Notes