Censored Poisson (version 2)

Parametrisation

The Poisson distribution is

$$\text{Prob}(y) = \frac{\lambda^y}{y!}\exp(-\lambda)$$

for responses $y = 0, 1, 2, \ldots$, where λ is the expected value. The cencored version is that reponse response in the interval $L \leq y \leq H$ are cencored (and reported as $y = L$, say), whereas other values are reported as is. This is often due to privacy issue, for example using $L = 1$ and $H = 5$, for example.

The "cenpoisson" probability distribution is then, for $y = 0, 1, \ldots,$

$$\text{Prob}^*(y) = \begin{cases} \sum_{z=L}^{H} \frac{\lambda^z}{z!}\exp(-\lambda) & L \leq y \leq H \\ \frac{\lambda^y}{y!}\exp(-\lambda) & \text{otherwise} \end{cases}$$

Link-function

The mean-parameter is λ and is linked to the linear predictor η by

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

where $E > 0$ is a known constant (or $\log(E)$ is the offset of η).

Hyperparameters

None.

Specification

- `family="cenpoisson2"`
- The `cenpoisson2` differ from `cenpoisson`, in that L and H are vectors and not scalars, hence different observations can have different censoring.
- Required arguments: y, E, L and H. The vector of the triplet (y_i, L_i, H_i) must be given as a `inla.mdata`-object. L and H are vectors of same length as y hence the cencoring can be different for each observation. L and H must be integer valued or `Inf`.

  $L[i]$ =`Inf` and/or $H[i]$ =`Inf` are allowed, which is equivalent to $L[i] = -1$ and/or $H[i] = -1$. See the example for details.

Example

In the following example we estimate the parameters in a simulated example with Poisson responses.

```
n <- 300
a <- 0
b <- 1
x <- rnorm(n, sd = 0.3)
eta = a + b*x
low = sample(c(0, 1, 4, Inf), n, replace = TRUE)
high <- low + sample(c(0, 1, 2, Inf), n, replace = TRUE)

E = sample(1:10, n, replace=TRUE)
lambda = E*exp(eta)
```

```
y = rpois(n, lambda = lambda)

censored = which(y >= low & y <= high)
y[censored] = low[censored]

r <- inla(inla.mdata(cbind(y, low, high))  ~ 1 + x,
          family = "cenpoisson2",
          data = data.frame(y, low, high, x),
          E=E)
summary(r)
```

Notes

For censored values, then y must be one arbitrary value in the interval; `NA` does not work!!!