

Occupancy likelihood

Parametrisation

This is a specialized likelihood to for occupancy models.

Details

An observation is an vector $y = (y_1, \dots, y_m)$ of binary observations, each depending on specific covariates, with additional zero-inflation. If there fewer than m observations, like $m' < m$, then “observations not observed” must be set to **NA**. The likelihood for one observation(-vector) is

$$f(y) = \phi \prod_{i=1}^m p_i^{y_i} (1 - p_i)^{1-y_i} + (1 - \phi) 1_{[y_i = 0, \forall i]}$$

with the convension that if $y_i = \text{NA}$, the contribution from y_i is ignored. Further,

$$\text{logit}(p_i) = x_i^T \beta$$

and $x_i = (x_{i1}, \dots, x_{ik})$ are the $k > 0$ covariate associated to y_i , and $\beta = (\beta_1, \dots, \beta_k)$ are the regression coefficients. The linear predictor from the formula η , goes into ϕ , as

$$\text{logit}(\phi) = \eta$$

Link-function

The link-function for the p_i -model is given by argument `link.simple` in the `control.family-` argument. The link-function for the ϕ -model is given as `normal`. Both defaults to the logit-link.

Hyperparameters

The regression coefficients β are treated as hyperparameters, and k is maximum 10. An intercept must be defined manually by adding a constant covariate vector.

Specification

- `family="occupancy"`
- Required arguments: A matrix Y with the observations and a matrix X with the covariates.

The matrix Y is $n \times m$, where m is then the *maximum* number of observations over all locations. If there fewer than m observations in one location, then **NA** must be added to reach m .

The matrix X stored the covariates. Since there are k covariate for each observation, then the dimension of X is $n \times mk$. For the i th observation(-vector), then the i th row of X is $(x_{i1}, x_{i2}, \dots, x_{im})$, where x_{ij} is the covariate vector for j th observation at location i . In a single observation is **NA**, the corresponding covariate(-vectors) is not used (but it still needs to be given).

Here is a simple example with $n = 5$ locations, maximum $m = 3$ observations, and $k = 2$ covariates.

```
> Y
      [,1] [,2] [,3]
[1,]    1    1  NA
[2,]    0    1    0
[3,]    0    0    0
[4,]    0    1    0
[5,]   NA    0    0
> round(dig=3,X)
      [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    1 -0.129    1 -0.248  NA   NA
[2,]    1 -0.030    1  0.151    1  0.100
[3,]    1 -0.148    1 -0.061    1  0.023
[4,]    1 -0.055    1  0.081    1 -0.112
[5,]   NA     NA    1  0.164    1  0.027
```

For $Y[1,1]$ the covariates are $(1, -0.129)$, for $Y[1,2]$ the covariates are $(1, -0.248)$, for $Y[1,3]$ is NA hence not used, for $Y[2,1]$ the covariates are $(1, -0.030)$, etc. We pass both Y and X in the `inla.mdata()` in the formula, as

```
inla.mdata(Y,X) ~ ...
```

Hyperparameter spesification and default values

`doc` Occupancy likelihood

`hyper`

`theta1`

```
hyperid 56601
name beta1
short.name beta1
output.name beta1 for occupancy observations
output.name.intern beta1 for occupancy observations
initial -2
fixed FALSE
prior normal
param -2 10
to.theta function(x) x
from.theta function(x) x
```

`theta2`

```
hyperid 56602
name beta2
short.name beta2
output.name beta2 for occupancy observations
output.name.intern beta2 for occupancy observations
initial 0
fixed FALSE
prior normal
param 0 10
to.theta function(x) x
```

```

    from.theta function(x) x
theta3
    hyperid 56603
    name beta3
    short.name beta3
    output.name beta3 for occupancy observations
    output.name.intern beta3 for occupancy observations
    initial 0
    fixed FALSE
    prior normal
    param 0 10
    to.theta function(x) x
    from.theta function(x) x
theta4
    hyperid 56604
    name beta4
    short.name beta4
    output.name beta4 for occupancy observations
    output.name.intern beta4 for occupancy observations
    initial 0
    fixed FALSE
    prior normal
    param 0 10
    to.theta function(x) x
    from.theta function(x) x
theta5
    hyperid 56605
    name beta5
    short.name beta5
    output.name beta5 for occupancy observations
    output.name.intern beta5 for occupancy observations
    initial 0
    fixed FALSE
    prior normal
    param 0 10
    to.theta function(x) x
    from.theta function(x) x
theta6
    hyperid 56606
    name beta6
    short.name beta6
    output.name beta6 for occupancy observations
    output.name.intern beta6 for occupancy observations
    initial 0
    fixed FALSE
    prior normal
    param 0 10
    to.theta function(x) x
    from.theta function(x) x

```

theta7

hyperid 56607
name beta7
short.name beta7
output.name beta7 for occupancy observations
output.name.intern beta7 for occupancy observations
initial 0
fixed FALSE
prior normal
param 0 10
to.theta function(x) x
from.theta function(x) x

theta8

hyperid 56608
name beta8
short.name beta8
output.name beta8 for occupancy observations
output.name.intern beta8 for occupancy observations
initial 0
fixed FALSE
prior normal
param 0 10
to.theta function(x) x
from.theta function(x) x

theta9

hyperid 56609
name beta9
short.name beta9
output.name beta9 for occupancy observations
output.name.intern beta9 for occupancy observations
initial 0
fixed FALSE
prior normal
param 0 10
to.theta function(x) x
from.theta function(x) x

theta10

hyperid 56610
name beta10
short.name beta10
output.name beta10 for occupancy observations
output.name.intern beta10 for occupancy observations
initial 0
fixed FALSE
prior normal
param 0 10
to.theta function(x) x
from.theta function(x) x

survival FALSE

discrete TRUE

link default logit

link.simple default logit

pdf occupancy

Example

```
n <- 1000
m <- 3
nc <- 2
beta <- c(-1, rnorm(nc-1, sd = 0.2))
Y <- matrix(NA, n, m)
X <- matrix(NA, n, m*nc)
z <- rnorm(n, mean = 0, sd = 0.3)
eta <- 3 + z
p.obs <- 1/(1 + exp(-eta))
for (i in 1:n) {
  is.zero <- rbinom(1, size = 1, prob = 1 - p.obs[i])
  nyy <- sample(2:m, 1)
  for(j in 1:m) {
    off <- (j-1) * nc
    if (j <= nyy) {
      X[i, off + 1:nc] <- c(1, rnorm(nc-1, sd = 0.1))
      eeta <- sum(X[i, off + 1:nc] * beta)
      p <- 1/(1 + exp(-eeta))
      if (is.zero) {
        Y[i, j] <- 0
      } else {
        Y[i, j] <- rbinom(1, size = 1, prob = p)
      }
    } else {
      X[i, off + 1:nc] <- rep(NA, nc)
      Y[i, j] <- NA
    }
  }
}

r <- inla(inla.mdata(Y, X) ~ 1 + z,
  family = "occupancy",
  data = list(Y = Y, X = X, z = z),
  control.fixed = list(prec.intercept = 1),
  control.inla = list(cmin = 0.0))
summary(r)
cbind(beta, r$summary.hyperpar[, "mean"])
```