New 0inflated models: Poisson & Binomial

Parametrisation

This is a new implementation (Nov'22) of zero-inflated Poisson and Binomial likelihood, where we will allow for a linear predictor in both the zero-inflattion and in the mean, but one of them needs to consists of fixed effects only. This means the setup will be somewhat different that for other likelihood models.

Details

The zero-inflated likelihood $f_0(y|\ldots)$ is defined as

$$f_0(y|\eta_1, \eta_2) = p(\eta_1)1_{[y=0]} + (1 - p(\eta_1))f(y|\eta_2)$$

where $f(y|\ldots)$ is either Poisson or Binomial. We allow for two linear predictors in the model, but one needs to be "simple" (i.e. only consists of fixed effects). The other is general and defined via the formula. Normally, the zero-inflation probability is simpler (`family="0..."`)

$$\eta_1 = \text{simple} \qquad \eta_2 = \text{formula}$$

but they can also be swapped (`family="0...S"`)

$$\eta_1 = \text{formula} \qquad \eta_2 = \text{simple}$$

Link-function

This is similar to Poisson and Binomial.

The link-function for the 'simple'-model must be given by argument `link.simple` in the `control.family`-argument. Only link-models without covariates/parameters are currently available. The examples later on shows how this is done.

Hyperparameters

All parameters in the simple model are treated as hyperparameters. The j'th element of η_1 is

$$(\eta_1)_j = \sum_{i=1} m\beta_i x_{ij}$$

for covariates $x_1,\ldots,$, where m is maximum 10. An intercept in this model have to be defined manually by adding a constant covariate vector.

Specification

- `family="0poisson"`

- `family="0poissonS"`

- `family="0binomial"`

- `family="0binomialS"`

- Required arguments: As for the Poisson and Binomial (but how these arguments are given, will differ). Optional argument `link.simple`.

Hyperparameter spesification and default values

0poisson

doc `New 0-inflated Poisson`

hyper

> **theta1**
>> **hyperid** `56201`
>> **name** `beta1`
>> **short.name** `beta1`
>> **output.name** `beta1 for 0poisson observations`
>> **output.name.intern** `beta1 for 0poisson observations`
>> **initial** `-4`
>> **fixed** `FALSE`
>> **prior** `normal`
>> **param** `-4 10`
>> **to.theta** `function(x) x`
>> **from.theta** `function(x) x`

> **theta2**
>> **hyperid** `56202`
>> **name** `beta2`
>> **short.name** `beta2`
>> **output.name** `beta2 for 0poisson observations`
>> **output.name.intern** `beta2 for 0poisson observations`
>> **initial** `0`
>> **fixed** `FALSE`
>> **prior** `normal`
>> **param** `0 10`
>> **to.theta** `function(x) x`
>> **from.theta** `function(x) x`

> **theta3**
>> **hyperid** `56203`
>> **name** `beta3`
>> **short.name** `beta3`
>> **output.name** `beta3 for 0poisson observations`
>> **output.name.intern** `beta3 for 0poisson observations`
>> **initial** `0`
>> **fixed** `FALSE`
>> **prior** `normal`
>> **param** `0 10`
>> **to.theta** `function(x) x`
>> **from.theta** `function(x) x`

> **theta4**
>> **hyperid** `56204`
>> **name** `beta4`
>> **short.name** `beta4`
>> **output.name** `beta4 for 0poisson observations`
>> **output.name.intern** `beta4 for 0poisson observations`
>> **initial** `0`
>> **fixed** `FALSE`

prior normal

param 0 10

to.theta function(x) x

from.theta function(x) x

theta5

hyperid 56205

name beta5

short.name beta5

output.name beta5 for 0poisson observations

output.name.intern beta5 for 0poisson observations

initial 0

fixed FALSE

prior normal

param 0 10

to.theta function(x) x

from.theta function(x) x

theta6

hyperid 56206

name beta6

short.name beta6

output.name beta6 for 0poisson observations

output.name.intern beta6 for 0poisson observations

initial 0

fixed FALSE

prior normal

param 0 10

to.theta function(x) x

from.theta function(x) x

theta7

hyperid 56207

name beta7

short.name beta7

output.name beta7 for 0poisson observations

output.name.intern beta7 for 0poisson observations

initial 0

fixed FALSE

prior normal

param 0 10

to.theta function(x) x

from.theta function(x) x

theta8

hyperid 56208

name beta8

short.name beta8

output.name beta8 for 0poisson observations

output.name.intern beta8 for 0poisson observations

initial 0

fixed FALSE

prior normal

param `0 10`
to.theta `function(x) x`
from.theta `function(x) x`

theta9

    hyperid `56209`
    name `beta9`
    short.name `beta9`
    output.name `beta9 for 0poisson observations`
    output.name.intern `beta9 for 0poisson observations`
    initial `0`
    fixed `FALSE`
    prior `normal`
    param `0 10`
    to.theta `function(x) x`
    from.theta `function(x) x`

theta10

    hyperid `56210`
    name `beta10`
    short.name `beta10`
    output.name `beta10 for 0poisson observations`
    output.name.intern `beta10 for 0poisson observations`
    initial `0`
    fixed `FALSE`
    prior `normal`
    param `0 10`
    to.theta `function(x) x`
    from.theta `function(x) x`

status `experimental`

survival `FALSE`

discrete `TRUE`

link `default log quantile`

link.simple `default logit cauchit probit cloglog ccloglog`

pdf `0inflated`

0poissonS

doc `New 0-inflated Poisson Swap`

hyper

    theta1

        hyperid `56301`
        name `beta1`
        short.name `beta1`
        output.name `beta1 for 0poissonS observations`
        output.name.intern `beta1 for 0poissonS observations`
        initial `-4`
        fixed `FALSE`
        prior `normal`
        param `-4 10`

to.theta `function(x) x`

from.theta `function(x) x`

theta2

    hyperid `56302`

    name `beta2`

    short.name `beta2`

    output.name `beta2 for 0poissonS observations`

    output.name.intern `beta2 for 0poissonS observations`

    initial `0`

    fixed `FALSE`

    prior `normal`

    param `0 10`

    to.theta `function(x) x`

    from.theta `function(x) x`

theta3

    hyperid `56303`

    name `beta3`

    short.name `beta3`

    output.name `beta3 for 0poissonS observations`

    output.name.intern `beta3 for 0poissonS observations`

    initial `0`

    fixed `FALSE`

    prior `normal`

    param `0 10`

    to.theta `function(x) x`

    from.theta `function(x) x`

theta4

    hyperid `56304`

    name `beta4`

    short.name `beta4`

    output.name `beta4 for 0poissonS observations`

    output.name.intern `beta4 for 0poissonS observations`

    initial `0`

    fixed `FALSE`

    prior `normal`

    param `0 10`

    to.theta `function(x) x`

    from.theta `function(x) x`

theta5

    hyperid `56305`

    name `beta5`

    short.name `beta5`

    output.name `beta5 for 0poissonS observations`

    output.name.intern `beta5 for 0poissonS observations`

    initial `0`

    fixed `FALSE`

    prior `normal`

    param `0 10`

    to.theta `function(x) x`

from.theta `function(x) x`

theta6

    hyperid `56306`

    name `beta6`

    short.name `beta6`

    output.name `beta6 for 0poissonS observations`

    output.name.intern `beta6 for 0poissonS observations`

    initial `0`

    fixed `FALSE`

    prior `normal`

    param `0 10`

    to.theta `function(x) x`

    from.theta `function(x) x`

theta7

    hyperid `56307`

    name `beta7`

    short.name `beta7`

    output.name `beta7 for 0poissonS observations`

    output.name.intern `beta7 for 0poissonS observations`

    initial `0`

    fixed `FALSE`

    prior `normal`

    param `0 10`

    to.theta `function(x) x`

    from.theta `function(x) x`

theta8

    hyperid `56308`

    name `beta8`

    short.name `beta8`

    output.name `beta8 for 0poissonS observations`

    output.name.intern `beta8 for 0poissonS observations`

    initial `0`

    fixed `FALSE`

    prior `normal`

    param `0 10`

    to.theta `function(x) x`

    from.theta `function(x) x`

theta9

    hyperid `56309`

    name `beta9`

    short.name `beta9`

    output.name `beta9 for 0poissonS observations`

    output.name.intern `beta9 for 0poissonS observations`

    initial `0`

    fixed `FALSE`

    prior `normal`

    param `0 10`

    to.theta `function(x) x`

    from.theta `function(x) x`

theta10

    hyperid 56310

    name beta10

    short.name beta10

    output.name beta10 for 0poissonS observations

    output.name.intern beta10 for 0poissonS observations

    initial 0

    fixed FALSE

    prior normal

    param 0 10

    to.theta function(x) x

    from.theta function(x) x

status experimental

survival FALSE

discrete TRUE

link default logit loga cauchit probit cloglog ccloglog loglog log sslogit logitoffset quantile pquanti

link.simple default log

pdf 0inflated

0binomial

doc New 0-inflated Binomial

hyper

    theta1

        hyperid 56401

        name beta1

        short.name beta1

        output.name beta1 for 0binomial observations

        output.name.intern beta1 for 0binomial observations

        initial -4

        fixed FALSE

        prior normal

        param -4 10

        to.theta function(x) x

        from.theta function(x) x

    theta2

        hyperid 56402

        name beta2

        short.name beta2

        output.name beta2 for 0binomial observations

        output.name.intern beta2 for 0binomial observations

        initial 0

        fixed FALSE

        prior normal

        param 0 10

        to.theta function(x) x

        from.theta function(x) x

    theta3

hyperid 56403

name beta3

short.name beta3

output.name beta3 for 0binomial observations

output.name.intern beta3 for 0binomial observations

initial 0

fixed FALSE

prior normal

param 0 10

to.theta function(x) x

from.theta function(x) x

theta4

hyperid 56404

name beta4

short.name beta4

output.name beta4 for 0binomial observations

output.name.intern beta4 for 0binomial observations

initial 0

fixed FALSE

prior normal

param 0 10

to.theta function(x) x

from.theta function(x) x

theta5

hyperid 56405

name beta5

short.name beta5

output.name beta5 for 0binomial observations

output.name.intern beta5 for 0binomial observations

initial 0

fixed FALSE

prior normal

param 0 10

to.theta function(x) x

from.theta function(x) x

theta6

hyperid 56406

name beta6

short.name beta6

output.name beta6 for 0binomial observations

output.name.intern beta6 for 0binomial observations

initial 0

fixed FALSE

prior normal

param 0 10

to.theta function(x) x

from.theta function(x) x

theta7

hyperid 56407

name `beta7`

short.name `beta7`

output.name `beta7 for 0binomial observations`

output.name.intern `beta7 for 0binomial observations`

initial `0`

fixed `FALSE`

prior `normal`

param `0 10`

to.theta `function(x) x`

from.theta `function(x) x`

theta8

hyperid `56408`

name `beta8`

short.name `beta8`

output.name `beta8 for 0binomial observations`

output.name.intern `beta8 for 0binomial observations`

initial `0`

fixed `FALSE`

prior `normal`

param `0 10`

to.theta `function(x) x`

from.theta `function(x) x`

theta9

hyperid `56409`

name `beta9`

short.name `beta9`

output.name `beta9 for 0binomial observations`

output.name.intern `beta9 for 0binomial observations`

initial `0`

fixed `FALSE`

prior `normal`

param `0 10`

to.theta `function(x) x`

from.theta `function(x) x`

theta10

hyperid `56410`

name `beta10`

short.name `beta10`

output.name `beta10 for 0binomial observations`

output.name.intern `beta10 for 0binomial observations`

initial `0`

fixed `FALSE`

prior `normal`

param `0 10`

to.theta `function(x) x`

from.theta `function(x) x`

status `experimental`

survival `FALSE`

discrete `TRUE`

link `default logit loga cauchit probit cloglog ccloglog loglog log`

link.simple `default logit cauchit probit cloglog ccloglog`

pdf `0inflated`

0poisson

doc `New 0-inflated Binomial Swap`

hyper

    theta1

        hyperid `56501`
        name `beta1`
        short.name `beta1`
        output.name `beta1 for 0binomialS observations`
        output.name.intern `beta1 for 0binomialS observations`
        initial `-4`
        fixed `FALSE`
        prior `normal`
        param `-4 10`
        to.theta `function(x) x`
        from.theta `function(x) x`

    theta2

        hyperid `56502`
        name `beta2`
        short.name `beta2`
        output.name `beta2 for 0binomialS observations`
        output.name.intern `beta2 for 0binomialS observations`
        initial `0`
        fixed `FALSE`
        prior `normal`
        param `0 10`
        to.theta `function(x) x`
        from.theta `function(x) x`

    theta3

        hyperid `56503`
        name `beta3`
        short.name `beta3`
        output.name `beta3 for 0binomialS observations`
        output.name.intern `beta3 for 0binomialS observations`
        initial `0`
        fixed `FALSE`
        prior `normal`
        param `0 10`
        to.theta `function(x) x`
        from.theta `function(x) x`

    theta4

        hyperid `56504`
        name `beta4`

short.name `beta4`

output.name `beta4 for 0binomialS observations`

output.name.intern `beta4 for 0binomialS observations`

initial `0`

fixed `FALSE`

prior `normal`

param `0 10`

to.theta `function(x) x`

from.theta `function(x) x`

theta5

hyperid `56505`

name `beta5`

short.name `beta5`

output.name `beta5 for 0binomialS observations`

output.name.intern `beta5 for 0binomialS observations`

initial `0`

fixed `FALSE`

prior `normal`

param `0 10`

to.theta `function(x) x`

from.theta `function(x) x`

theta6

hyperid `56506`

name `beta6`

short.name `beta6`

output.name `beta6 for 0binomialS observations`

output.name.intern `beta6 for 0binomialS observations`

initial `0`

fixed `FALSE`

prior `normal`

param `0 10`

to.theta `function(x) x`

from.theta `function(x) x`

theta7

hyperid `56507`

name `beta7`

short.name `beta7`

output.name `beta7 for 0binomialS observations`

output.name.intern `beta7 for 0binomialS observations`

initial `0`

fixed `FALSE`

prior `normal`

param `0 10`

to.theta `function(x) x`

from.theta `function(x) x`

theta8

hyperid `56508`

name `beta8`

short.name `beta8`

output.name `beta8 for 0binomialS observations`

output.name.intern `beta8 for 0binomialS observations`

initial `0`

fixed `FALSE`

prior `normal`

param `0 10`

to.theta `function(x) x`

from.theta `function(x) x`

theta9

hyperid `56509`

name `beta9`

short.name `beta9`

output.name `beta9 for 0binomialS observations`

output.name.intern `beta9 for 0binomialS observations`

initial `0`

fixed `FALSE`

prior `normal`

param `0 10`

to.theta `function(x) x`

from.theta `function(x) x`

theta10

hyperid `56510`

name `beta10`

short.name `beta10`

output.name `beta10 for 0binomialS observations`

output.name.intern `beta10 for 0binomialS observations`

initial `0`

fixed `FALSE`

prior `normal`

param `0 10`

to.theta `function(x) x`

from.theta `function(x) x`

status `experimental`

survival `FALSE`

discrete `TRUE`

link `default logit loga cauchit probit cloglog ccloglog loglog log`

link.simple `default logit cauchit probit cloglog ccloglog`

pdf `0inflated`

Example: Poisson

```
sim.poisson <- function(prob, m)
{
    stopifnot(length(prob) == length(m) && length(prob) > 0)
    n <- length(m)
    y <- numeric(n)
    event <- (runif(n) < prob)
    idx.zero <- which(event)
    idx.non.zero <- which(!event)
    y[idx.zero] <- 0
    y[idx.non.zero] <- rpois(length(idx.non.zero), lambda = m[idx.non.zero])
    return (y)
}


## chose link-function to use for the zero-inflation probability
link.simple <- "logit"
inv.link <- inla.link.invlogit
## link.simple <- "probit"
## inv.link <- inla.link.invprobit
## link.simple <- "cloglog"
## inv.link <- inla.link.invcloglog


n <- 1000
z <- rnorm(n, sd = 0.3)
x <- rnorm(n, sd = 0.2)
xx <- rnorm(n, sd = 0.3)
zz <- rnorm(n, sd = 0.2)
E <- runif(n, min = 0.8, max = 1/0.8)

beta <- c(1, 1.1, 2.1, 0, -2, 1.2, 2.2, 0)
eta2 <- beta[1] + beta[2] * xx + beta[3] * zz + beta[4] * xx * zz
eta1 <- beta[5] + beta[6] * x + beta[7] * z + beta[8] * x * z
prob <- inv.link(eta1)
m <- E*exp(eta2)

ok <- FALSE
while(!ok) {
    y <- sim.poisson(prob, m)
    ok <- !all(y == 0)
}

## head(data.frame(y, E, x, z, xx, zz))

r <- inla(
    inla.mdata(cbind(y, E), cbind(1, x, z, x*z)) ~ 1 + xx + zz + xx*zz,
    family = "0poisson",
    data = data.frame(y, E, x, z, xx, zz),
    control.fixed = list(prec = 1, prec.intercept = 1),
    control.compute = list(cpo = TRUE),
    control.family = list(link.simple = link.simple,
                          hyper = list(beta1 = list(param = c(0, 1)),
                                       beta2 = list(param = c(0, 1)),
                                       beta3 = list(param = c(0, 1)),
                                       beta4 = list(param = c(0, 1)),
                                       beta5 = list(param = c(0, 1)))))


rr <- inla(
```

```
        inla.mdata(cbind(y, E), cbind(1, xx, zz, xx*zz)) ~ 1 + x + z + x*z,
        family = "0poissonS",
        data = data.frame(y, E, x, z, xx, zz),
        control.fixed = list(prec = 1, prec.intercept = 1),
        control.compute = list(cpo = TRUE),
        ## in this case we need to define link.simple as the main link
        control.family = list(control.link = list(model = link.simple),
                              hyper = list(beta1 = list(param = c(0, 1)),
                                           beta2 = list(param = c(0, 1)),
                                           beta3 = list(param = c(0, 1)),
                                           beta4 = list(param = c(0, 1)),
                                           beta5 = list(param = c(0, 1)))))
summary(r)
summary(rr)

res <- cbind("beta" = beta,
             "0poisson" = c(r$summary.fixed$mean, r$summary.hyperpar$mean),
             "0poissonS" = c(rr$summary.hyperpar$mean, rr$summary.fixed$mean))
res <- cbind(res,
             diff = (res[, 2]-beta),
             diffS = (res[, 3]-beta),
             "diff/sd" = (res[, 2]-beta) / c(r$summary.fixed$sd, r$summary.hyperpar$sd),
             "diffS/sd" = (res[, 3]-beta) / c(rr$summary.hyperpar$sd, rr$summary.fixed$sd))
mm <- nrow(res) %/% 2
rownames(res) <- c(paste0("beta", 1:mm, ".poisson"), paste0("beta", 1:mm, ".prob"))
print(round(dig = 2, res))
```

Example: Binomial

```
sim.binomial <- function(prob, p, size)
{
    ## - prob=zero-inflation-prob
    ## - binomial(size, p)
    stopifnot(length(prob) == length(p) && length(prob) == length(size)
              && length(prob) > 0)
    n <- length(prob)
    y <- numeric(n)
    event <- (runif(n) < prob)
    idx.zero <- which(event)
    idx.non.zero <- which(!event)
    y[idx.zero] <- 0
    y[idx.non.zero] <- rbinom(length(idx.non.zero),
                              size = size[idx.non.zero],
                              prob = p[idx.non.zero])
    return (y)
}


n <- 1000
z <- rnorm(n, sd = 0.3)
x <- rnorm(n, sd = 0.2)
xx <- rnorm(n, sd = 0.3)
zz <- rnorm(n, sd = 0.2)
Ntrials <- sample(1:10, n, replace = TRUE)

## chose link-function to use for the zero-inflation probability
link.simple <- "logit"
inv.link <- inla.link.invlogit
## link.simple <- "probit"
## inv.link <- inla.link.invprobit
## link.simple <- "cloglog"
## inv.link <- inla.link.invcloglog

beta <- c(1, 1.1, 2.1, 0, -2, 1.2, 2.2, 0)
eta2 <- beta[1] + beta[2] * xx + beta[3] * zz + beta[4] * xx * zz
eta1 <- beta[5] + beta[6] * x + beta[7] * z + beta[8] * x * z
prob <- inv.link(eta1)
p <- 1/(1 + exp(-eta2))

ok <- FALSE
while(!ok) {
    y <- sim.binomial(prob, p, Ntrials)
    ok <- !all(y == 0)
}

## head(data.frame(y, Ntrials, x, z, xx, zz))

r <- inla(
    inla.mdata(cbind(y, Ntrials), cbind(1, x, z, x*z)) ~ 1 + xx + zz + xx*zz,
    family = "0binomial",
    data = data.frame(y, Ntrials, x, z, xx, zz),
    control.fixed = list(prec = 1, prec.intercept = 1),
    control.compute = list(cpo = TRUE),
    control.family = list(link.simple = link.simple,
                          hyper = list(beta1 = list(param = c(0, 1)),
                                       beta2 = list(param = c(0, 1)),
```

```
                                           beta3 = list(param = c(0, 1)),
                                           beta4 = list(param = c(0, 1)),
                                           beta5 = list(param = c(0, 1))))))

rr <- inla(
    inla.mdata(cbind(y, Ntrials), cbind(1, xx, zz, xx*zz)) ~ 1 + x + z + x*z,
    family = "0binomialS",
    data = data.frame(y, Ntrials, x, z, xx, zz),
    control.fixed = list(prec = 1, prec.intercept = 1),
    control.compute = list(cpo = TRUE),
    ## in this case we need to define link.simple as the main link
    control.family = list(control.link = list(model = link.simple),
                          hyper = list(beta1 = list(param = c(0, 1)),
                                       beta2 = list(param = c(0, 1)),
                                       beta3 = list(param = c(0, 1)),
                                       beta4 = list(param = c(0, 1)),
                                       beta5 = list(param = c(0, 1))))))

summary(r)
summary(rr)

res <- cbind("beta" = beta,
             "0binomial" = c(r$summary.fixed$mean, r$summary.hyperpar$mean),
             "0binomialS" = c(rr$summary.hyperpar$mean, rr$summary.fixed$mean))
res <- cbind(res,
             diff = (res[, 2]-beta),
             diffS = (res[, 3]-beta),
             "diff/sd" = (res[, 2]-beta) / c(r$summary.fixed$sd, r$summary.hyperpar$sd),
             "diffS/sd" = (res[, 3]-beta) / c(rr$summary.hyperpar$sd, rr$summary.fixed$sd))
mm <- nrow(res) %/% 2
rownames(res) <- c(paste0("beta", 1:mm, ".binomial"), paste0("beta", 1:mm, ".prob"))
print(round(dig = 2, res))
```