Package ‘chainbinomial’

May 8, 2026

Type Package

Title Chain Binomial Models for Analysis of Infectious Disease Data

Version 0.1.5

Maintainer Jonas Christoffer Lindstrøm <`jonaslindstrom88@gmail.com`>

Description Implements the chain binomial model for analysis of infectious disease data. Contains functions for calculating probabilities of the final size of infectious disease outbreaks using the method from D. Ludwig (1975) <doi:10.1016/0025-5564(75)90119-4> and for outbreaks that are not concluded, from Lindstrøm et al. (2024) <doi:10.48550/arXiv.2403.03948>. The package also contains methods for estimation and regression analysis of secondary attack rates.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

Suggests testthat (>= 3.0.0), numDeriv (>= 2016.8-1.1), dplyr (>= 1.1.3), tidyr (>= 1.3.0), tibble, knitr, rmarkdown

Config/testthat/edition 3

Imports generics

Depends R (>= 3.5.0)

VignetteBuilder knitr

NeedsCompilation no

Author Jonas Christoffer Lindstrøm [aut, cre]

Repository CRAN

Date/Publication 2024-05-27 17:40:09 UTC

Contents

cbmod 2
coef.cbmod 4

confint.cbmod 4
confint.sar 5
dchainbinom 5
estimate_sar 6
glance.cbmod 7
heasman_reid_1961 8
predict.cbmod 9
summary.cbmod 10
tidy.cbmod 10
vcov.cbmod 11

Index **12**

`cbmod` *Fitting models for Secondary Attack Rate with Chain Binomial response*

Description

Fitting models for Secondary Attack Rate with Chain Binomial response

Usage

```
cbmod(
  y,
  s0,
  x = NULL,
  i0 = 1,
  generations = Inf,
  link = "identity",
  optim_method = "BFGS"
)
```

Arguments

`y`	numeric, the number of infected cases.
`s0`	numeric, the number of initial susceptibles.
`x`	matrix of predictors (design matrix).
`i0`	numeric, number of initial infected. Default is 1.
`generations`	numeric.
`link`	Link function. Default is 'identity'.
`optim_method`	Optimization method used by optim.

Details

The following link functions are available: `identity`, `log`, `logit`, and `cloglog`.

Value

A list of class `cbmod` with the following components:

- `parameters` The point estimate of the regression coefficients.
- `se` Standard error of the regression coefficient estimates.
- `vcov` Variance-Covariance matrix of the regression coefficient estimates.
- `p_values` P-values of the null hypothesis that the regression regression coefficient estimate is 0.
- `loglikelihood` the log likelihood value at the point estimate.
- `npar` Number of parameters.
- `sar_hat` Vector of fitted secondary attack rates.
- `fitted_values` Vector of expected outbreak size (final attack rate).
- `link` Link function used by the regression model.
- `null_model` = Null model, fitted with `estimate_sar()`. This is equivalent to an intercept only model.
- `warnings` Warning_messages,
- `est_time`: Time used to fit the model.
- `omitted_values` Vector indicating data points that were ignored during estimation because of missing values.

See Also

Methods for `cbmod` objects:

- `summary.cbmod()`
- `predict.cbmod()`
- `coef.cbmod()`
- `confint.cbmod()`
- `vcov.cbmod()`
- `tidy.cbmod()`
- `glance.cbmod()`

Examples

```
set.seed(234)
mydata <- data.frame(infected = rchainbinom(n = 15, s0 = 5, sar = 0.2,
   i0 = 1, generations = Inf),
   s0 = 5, i0 = 1, generations = Inf)
xmat <- model.matrix(~ 1, data = mydata)
res <- cbmod(y = mydata$infected, s0 = mydata$s0, x = xmat, i0 = mydata$i0,
   generations = mydata$generations)
summary(res)
```

`coef.cbmod` *Extract Model Coefficient for cbmod Fits*

Description

Extract Model Coefficient for cbmod Fits

Usage

```
## S3 method for class 'cbmod'
coef(object, ...)
```

Arguments

`object`	a cbmod object.
`...`	other arguments. Ignored.

Value

Coefficients extracted from the cbmod object.

`confint.cbmod` *Confidence intervals for cbmod Object.*

Description

Confidence intervals for cbmod Object.

Usage

```
## S3 method for class 'cbmod'
confint(object, parm = NULL, level = 0.95, ...)
```

Arguments

`object`	a cbmod object.
`parm`	Character or number of which coefficient ot compute confidence intervals for. By default intervals are computed for all coefficients.
`level`	Default is 0.95, for 95% confidence intervals.
`...`	other arguments. Ignored.

Value

A two-column matrix with the lower and upper end of the confidence intervals.

`confint.sar` *Confidence intervals for sar Object.*

Description

Confidence intervals for sar Object.

Usage

```
## S3 method for class 'sar'
confint(object, parm = NULL, level = 0.95, method = "chisq", ...)
```

Arguments

`object`	a cbmod object.
`parm`	Character or number of which coefficient ot compute confidence intervals for. By default intervals are computed for all coefficients.
`level`	Default is 0.95, for 95% confidence intervals.
`method`	Either 'chisq'(default) or 'normal'.
`...`	other arguments. Ignored.

Value

A numeric of length 2 with the lower and upper end of the confidence interval.

`dchainbinom` *The Chain Binomial distribution*

Description

Probability mass function, expected value, and random generation, for the chain binomial distribution, with parameters s0, sar, i0, and number of generations, for the number of infected cases in a population of size s0 after a given number of generations.

Usage

```
dchainbinom(x, s0, sar, i0 = 1, generations = Inf)

rchainbinom(n, s0, sar, i0 = 1, generations = Inf)

echainbinom(s0, sar, i0 = 1, generations = Inf)
```

Arguments

`x`	numeric vector of the number of infected.
`s0`	the number of initial susceptibles.
`sar`	the secondary attack rate, or the per person risk of infection by an infected.
`i0`	the number of primary cases.
`generations`	the number of generations. Default is Inf, which represents the entire epidemic.
`n`	number of observations. If length(n) > 1, the length is taken to be the number required.

Value

`dchainbinom` gives the probability of x infected, given s0, i0, sar and generations. `echainbinom` gives the expected value. `rchainbinom` generates random data.

Examples

```
dchainbinom(x = 0:5, s0 = 5, sar = 0.2, i0 = 1, generations = Inf)
rchainbinom(n = 10, s0 = 5, sar = 0.2, i0 = 1, generations = Inf)
echainbinom(s0 = 5, sar = 0.2, i0 = 1, generations = Inf)
```

`estimate_sar` *Estimate Secondary Attack Rate of the Chain Binomial Model*

Description

Given data on the number of infected after a number of generation, initial number of susceptible, and initial number of infected, estimate the secondary attack rate (SAR) using maximum likelihood.

Usage

```
estimate_sar(infected, s0, i0 = 1, generations = Inf, se = TRUE)
```

Arguments

`infected`	numeric.
`s0`	numeric.
`i0`	numeric.
`generations`	numeric.
`se`	logical. If TRUE (default), the standard error is computed.

Value

A list of class `sar` with the following components:

- `sar_hat` The point estimate of the secondary attack rate.
- `se` Standard error of the estimate (if se = TRUE).
- `loglikelihood` The log likelihood value at the point estimate.
- `data` The input data.

See Also

`confint.sar()` for calculating confidence intervals.

Examples

```
set.seed(234)
mydata <- rchainbinom(n = 15, s0 = 5, sar = 0.2, i0 = 1, generations = Inf)
res <- estimate_sar(infected = mydata, s0 = 5, i0 = 1, generations = Inf)
```

`glance.cbmod` *Glance at a cbmod object*

Description

Glance at a cbmod object

Usage

```
## S3 method for class 'cbmod'
glance(x, ...)
```

Arguments

`x`	A cbmod object.
`...`	Unused.

Value

Returns a tibble with the following columns:

- `logLik` The model’s log-likelihood.
- `npar` Numnber of parameters in the model.

`heasman_reid_1961` *Common Cold Data*

Description

Data presented in Heasman & Reid (1961), originally gathered and analyzed by Brimblecombe et al (1958). The data set describes 664 outbreaks of the common cold in 72 families over two years. All families consists of two parents and three children. The data is available in aggregated form as presented in the paper, with counts of the number of outbreaks that belong to a given classification.

Usage

```
heasman_reid_1961_chains

heasman_reid_1961_crowding

heasman_reid_1961_intro_case_status

heasman_reid_1961_intro_case_status

heasman_reid_1961_crowding
```

Format

An object of class `data.frame` with 24 rows and 2 columns.

An object of class `data.frame` with 5 rows and 5 columns.

An object of class `data.frame` with 5 rows and 4 columns.

Chain Data

Each outbreak was classified to a specific chain suitable for analysis by the Chain Binomial model by Heasman & Reid (1961), Table V.

`heasman_reid_1961_chains`: A data frame with 24 rows and 2 columns:

chain the number of infected in each generation, separated by '-', ie the Chain.

n Number of outbreaks

Crowding

Each outbreak classified according to the degree of domestic overcrowding. Heasman & Reid (1961), Table IV. Overcrowded homes have either one or two rooms, crowded homes have three rooms, while uncrowded homes have more than three rooms.

`heasman_reid_1961_crowding`: A data frame with 5 rows and 4 columns:

further_cases The number of cases in the outbreak, in addition to the primary case.

overcrowding Number of outbreaks that belong to the overcrowded household category.

crowded Number of outbreaks that belong to the crowded household category.

uncrowded Number of outbreaks that belong to the uncrowded household category.

Index case status

Each outbreak classified according to who the the introducing case was. Heasman & Reid (1961), Table II.

`heasman_reid_1961_intro_case_status`: A data frame with 5 rows and 5 columns:

further_cases The number of cases in the outbreak, in addition to the primary case.

father Number of outbreaks with father as the index case.

mother Number of outbreaks with mother as the index case.

school_child Number of outbreaks with a school child as the index case.

pre_school_child Number of outbreaks with a pre-school child as the index case.

References

- Heasman & Reid (1961) Theory And Observation In Family Epidemics Of The Common Cold. Brit. J. prev. soc. Med.
- Brimblecombe et al (1958) Family Studies Of Respiratory Infections. British Medical Journal.

`predict.cbmod` *Predict Method for cbmod Fits*

Description

Predict Method for cbmod Fits

Usage

```
## S3 method for class 'cbmod'
predict(object, x, type = "identity", ...)
```

Arguments

`object`	a fitted object of class inheriting from "cbmod".
`x`	matrix of predictors (design matrix). Must have the same column names and order as the x matrix used to fit the model.
`type`	the type of prediction, either 'link' (default) or 'sar'. The default is on the scale of the linear predictors. 'sar' gives the predicted secondary attack rate, by transforming the linear predictors by the inverse link function used in the model fit.
`...`	additional arguments.

Value

a vector of predictions.

Examples

```
set.seed(234)
mydata <- data.frame(infected = rchainbinom(n = 15, s0 = 5, sar = 0.2,
  i0 = 1, generations = Inf),
  s0 = 5, i0 = 1, generations = Inf)
xmat <- model.matrix(~ 1, data = mydata)
res <- cbmod(y = mydata$infected, s0 = mydata$s0, x = xmat, i0 = mydata$i0,
  generations = mydata$generations, link = 'identity')
summary(res)
predict(res, x = xmat, type = 'sar')
```

`summary.cbmod` *Summary of cbmod Object.*

Description

Summary of cbmod Object.

Usage

```
## S3 method for class 'cbmod'
summary(object, ...)
```

Arguments

`object`	a cbmod object.
`...`	other arguments. Ignored.

Value

Returns nothing, but displays a summary of the model fit.

`tidy.cbmod` *Tidy a cbmod object*

Description

Tidy a cbmod object

Usage

```
## S3 method for class 'cbmod'
tidy(x, conf.int = FALSE, conf.level = 0.95, ...)
```

Arguments

`x`	A `cbmod` object.
`conf.int`	Logical indicating whether or not to include a confidence interval in the tidied output. Defaults to FALSE.
`conf.level`	The confidence level to use for the confidence interval if conf.int = TRUE. Defaults to 0.95, which corresponds to a 95 percent confidence interval.
`...`	Unused.

Value

Returns a tibble with the following columns:

- `term` The coefficients name.
- `estimate` The point estimates of the coefficients.
- `std.error` Standard error of the regression coefficient estimates.
- `p.value` P-values of the null hypothesis that the regression regression coefficient estimate is 0.
- `conf.low` If `conf.int` = `TRUE`, the lower end of the confidence interval.
- `conf.high`If `conf.int` = `TRUE`, the upper end of the confidence interval.

`vcov.cbmod` *Variance-Covariance Matrix of cbmod Object.*

Description

Variance-Covariance Matrix of cbmod Object.

Usage

```
## S3 method for class 'cbmod'
vcov(object, ...)
```

Arguments

`object`	a cbmod object.
`...`	other arguments. Ignored.

Value

A variance-covariance matrix.

Index

∗ **datasets**
    heasman_reid_1961, 8

cbmod, 2
coef.cbmod, 4
coef.cbmod(), *3*
confint.cbmod, 4
confint.cbmod(), *3*
confint.sar, 5
confint.sar(), *7*

dchainbinom, 5

echainbinom *(*dchainbinom*)*, 5
estimate_sar, 6
estimate_sar(), *3*

glance.cbmod, 7
glance.cbmod(), *3*

heasman_reid_1961, 8
heasman_reid_1961_chains
        *(*heasman_reid_1961*)*, 8
heasman_reid_1961_crowding
        *(*heasman_reid_1961*)*, 8
heasman_reid_1961_intro_case_status
        *(*heasman_reid_1961*)*, 8

predict.cbmod, 9
predict.cbmod(), *3*

rchainbinom *(*dchainbinom*)*, 5

summary.cbmod, 10
summary.cbmod(), *3*

tidy.cbmod, 10
tidy.cbmod(), *3*

vcov.cbmod, 11
vcov.cbmod(), *3*