Package 'brixtools'

August 13, 2017

Type Package

Title Some statistical methods for non-life insurance

Version 0.1.0

Author Kristoffer Brix

Maintainer Kristoffer Brix <kbrix@protonmail.com>

Description
Implementation of some statistical methods for non-life insurance taking deductibles into account.

Depends R (>= 2.10)

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

Suggests testthat, knitr, rmarkdown

URL https://github.com/kbrix/brixtools

BugReports https://github.com/kbrix/brixtools/issues

VignetteBuilder knitr

Imports truncdist

R topics documented:

augment *Augmentation of simple_brix model*

Description

Augments simple_brix model with unobserved claims (losses smaller than the deductibles) for claims model distribution testing.

Usage

```
augment(object)
```

Arguments

object	object of class "brix_simple", see `brix_simple`.

Details

Using the Poisson distribution for augmentation of unobserved claims. (Might implement a Geometric method).

Value

Returns an object of `class` "augment". Using the print command displays test results from both the Shaprio-Wilks Normality test amd the Kolmogorov-Smirnov one-sample test for the standard normal transformed augmented data. See `shapiro.test` and `ks.test`. Using the plot command displays standard normal Q-Q plots for the standard normal transformed augmented data by default.

Examples

```
b <- brix_simple(optimizer = optim,
                 fn = normal_poisson_log_likelihood,
                 par = c(1, 1, 1),
                 control = list(fnscale = -1, maxit = 1e6),
                 hessian = TRUE,
                 X = normal_poisson_x, N = normal_poisson_n)

a <- augment(object = b)
print(a)
plot(a)
plot(a, augment = FALSE)
```

brix_simple *Simple optimizer*

Description

A simple optimizer for log-likelihoods or other functions.

Usage

```
brix_simple(optimizer, fn, X, N, ...)
```

Arguments

`optimizer`	recommended choice is `optim`.
`fn`	function to be optimized.
`X`	`data.frame` of claims and deductibles.
`N`	`data.frame` of occurrences and deductibles.
`...`	additional arguments passed to the `optimizer`, see `examples`.

Details

The optimizer must always be specified first. The function to be optimized must always be specified second. See e.g. `optim` for the optimizer.

Value

Returns an object of `class` "brix_simple". The function `summary` can also be used to print a summary of the model.

An object of class "brix" is a list containing at the function call and the results from the optimizer. The optimal parameters are named if applicable. The summary contains at least the following components:

`call`	the function call.
`coefficients`	the optimal coefficents.
`value`	the optimal value of the log-likelihood function.
`AIC`	the AIC of the log-likelihood.

Examples

```
# Using optim for optimization
model <- brix_simple(optimizer = optim,
                     # function to be optimized
                     fn = normal_poisson_log_likelihood,
                     # arguments passed to fn
                     X = normal_poisson_x, N = normal_poisson_n,
                     # additional arguments passed to optim
                     par = rep(1, 3),
                     control = list(fnscale = -1),
                     method = "Nelder-Mead",
                     hessian = TRUE)
model
summary(model)
```

normal_poisson_log_likelihood *Normal-Poisson log-likelihood*

Description

Implementation of Normal-Poisson log-likelihood, taking deductibles into account.

Usage

```
normal_poisson_log_likelihood(parameter = NULL, X = NULL, N = NULL,
  check = 1)
```

Arguments

`parameter` numeric, vector of parameters of length 3 consisting of the mean and standard deviation of the Normal component and the mean of the Poisson component. Default is NULL.

`X` data.frame, must have two columns named claim and deductible for the Normal component of the log-likelihood. Default is NULL.

`N` data.frame, must have two columns occurrence and deductible for the Poisson component of the log-likelihood. Default is NULL.

`check` default value is 1 and returns the log-likelihood, else returns the parameter names of the model: mean, sd and lambda.

Details

Note that the row number of X and N need not be the same.

Value

Returns a numeric value or character vector of parameter names depending on whether check = 1 or not.

Examples

```
normal_poisson_log_likelihood(c(1,2,3), X = normal_poisson_x, N = normal_poisson_n, check = 1)
normal_poisson_log_likelihood(c(1,2,3), X = normal_poisson_x, N = normal_poisson_n, check = 0)
normal_poisson_log_likelihood(check = 0)
```

normal_poisson_policy *Normal-Poisson policy data*

Description

Simulated Normal-Poisson policy data for testing and demostration purposes.

Usage

```
normal_poisson_policy

normal_poisson_n

normal_poisson_x
```

Format

The object `normal_poisson_policy` is a `list` with $m = 10000$ policies. Each policy consists of the following:

deductible Sample of standard Normal random variable of length 1.

occurrence Sample of Poisson distributed random varible of length 1 with mean 1.

loss Sample of Normal distributed random variable of length `occurrence`, with mean 1.5 and standard deviation 1.5. Returns `numeric(0)` if `occurrence` is 0.

claim The `loss` given that it is larger than the `deductible`. Returns `numeric(0)` if `occurrence` is zero or if all losses are smaller than the `deductible`.

reported Length of `claim`. Returns zero if `claim` is `numeric(0)`.

The object `normal_poisson_n` is a `data.frame` with 10000 rows (policies) and 3 columns consisting of `occurrence`, `deductible` and `length`.

The object `normal_poisson_x` is a `data.frame` with `sum(normal_poisson_n$length)` rows and 2 columns consisting of `claim` and `deductible`.

Index