Package ‘burgle’

July 9, 2024

Type Package

Title 'Burgle': Stealing the Necessary Parts of Model Objects

Version 0.1.1

Maintainer Paul R. Gunsalus <gunsalp@ccf.org>

Description Provides a way to reduce model objects to necessary parts, making them easier to work with, store, share and simulate multiple values for new responses while allowing for parameter uncertainty.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

Imports stats, MASS, survival, riskRegression

Suggests flexsurv, nnet

Depends R (>= 4.0.0)

NeedsCompilation no

Author Paul R. Gunsalus [aut, cre] (<https://orcid.org/0000-0001-6976-9094>), Jarrod E. Dalton [aut] (<https://orcid.org/0000-0001-8336-1212>)

Repository CRAN

Date/Publication 2024-07-09 14:40:02 UTC

Contents

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>burgle_</td>
<td></td>
</tr>
<tr>
<td>predict_burgle</td>
<td></td>
</tr>
</tbody>
</table>

Index

1
Description

Burgling what is necessary from different objects

Usage

burgle(object, ...)

## S3 method for class 'lm'
burgle(object, ...)

## S3 method for class 'glm'
burgle(object, ...)

## S3 method for class 'CauseSpecificCox'
burgle(object, ...)

## S3 method for class 'flexsurvreg'
burgle(object, ...)

## S3 method for class 'multinom'
burgle(object, ...)

## S3 method for class 'coxph'
burgle(object, ...)

Arguments

object the model object to burgle
...

must be left empty for now

Value

a burgle_. object

Examples

fit <- lm(Sepal.Length ~ Sepal.Width + Petal.Length, data = iris)
bfit <- burgle(fit)
object.size(fit)
object.size(bfit)
**predict_burgle**

**Predict for burgle methods**

### Description

Predict for burgle methods

### Usage

```r
## S3 method for class 'burgle_CauseSpecificCox'
predict(
  object,
  newdata = NULL,
  type = "lp",
  cause = 1,
  original = TRUE,
  draws = 1,
  sims = 1,
  times = NULL,
  ...
)

## S3 method for class 'burgle_flexsurvreg'
predict(
  object,
  newdata = NA,
  original = TRUE,
  draws = 1,
  sims = 1,
  type = "lp",
  times = NULL,
  ...
)

## S3 method for class 'burgle_multinom'
predict(
  object,
  newdata = NA,
  original = TRUE,
  draws = 1,
  Sims = 1,
  type = "lp",
  floor = FALSE,
  seed = NULL,
  ...
)
```
## S3 method for class 'burgle_coxph'
predict(
  object,
  newdata = NA,
  original = TRUE,
  draws = 1,
  sims = 1,
  type = "lp",
  times = NULL,
  ...
)

## S3 method for class 'burgle_lm'
predict(
  object,
  newdata,
  original = TRUE,
  draws = 1,
  sims = 1,
  type = "lp",
  se = FALSE,
  ...
)

## S3 method for class 'burgle_glm'
predict(
  object,
  newdata,
  original = TRUE,
  draws = 1,
  sims = 1,
  type = "lp",
  se = FALSE,
  ...
)

**Arguments**

- **object**: the results of burgle_* object
- **newdata**: new data of class data.frame
- **type**: either 'lp', 'response', 'link' for glm or 'risk' if time dependent
- **cause**: which cause do you want to predict
- **original**: whether or not to predict using the original model
- **draws**: how many different models to simulate
- **sims**: how many simulated response to draw
- **times**: if type = "risk" time for which to predict risk, if times and sims is multiple the return will be lists within lists
predict_burgle

... for future methods
floor will set the minimum odds to 0, if negative odds exists
seed a seed to specify for simulating responses (multinomial only)
se whether or not to include the standard error in the simulations

**Value**

either a matrix of array of new model predictions
Index

burgle (burgle_), 2
burgle_., 2

predict.burgle_CauseSpecificCox
  (predict_burgle), 3
predict.burgle_coxph (predict_burgle), 3
predict.burgle_flexsurvreg
  (predict_burgle), 3
predict.burgle_glm (predict_burgle), 3
predict.burgle_lm (predict_burgle), 3
predict.burgle_multinom
  (predict_burgle), 3
predict_burgle, 3