# Package 'burgle'

October 1, 2024

**Type** Package

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

**Version** 0.1.2

**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>),
Adam T. Perzynski [aut] (<https://orcid.org/0000-0002-1323-0353>)

**Repository** CRAN

**Date/Publication** 2024-10-01 08:40:07 UTC

## Contents

---

burgle_                               *Burgle*

---

### 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 'cph'
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

```
## 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_cph'
predict(object, ...)

## 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,
  limits = NULL,
  ...
)

## 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 |
| ... | for future methods |
| floor | will set the minimum odds to 0, if negative odds exists |
| seed | a seed to specificy for simulating responses (multinomial only) |
| se | whether or not to include the standard error in the simulations |
| limits | limits (minimum and maximum) for simulated response values. |

## Value

either a matrix or list of new model predictions

# Index