Package 'sure'

October 25, 2026

Type Package

Title Surrogate Residuals for Ordinal and General Regression Models

Description An implementation of the surrogate approach to residuals and diagnostics for ordinal and general regression models; for details, see Liu and Zhang (2018) <doi:10.1080/01621459.2017.1292915> and Greenwell et al. (2018) <doi:10.32614/RJ-2018-004>. These residuals can be used to construct standard residual plots for model diagnostics (e.g., residual-vs-fitted value plots, residual-vs-covariate plots, Q-Q plots, etc.). The package also provides 'plot' methods for producing standard diagnostic plots using lightweight base R graphics (via 'tinyplot'). The package currently supports cumulative link models from packages 'MASS', 'ordinal', 'rms', and 'VGAM', binary and other generalized linear models fit with 'glm', and multinomial models or general classifiers via a matrix of predicted class probabilities.

Version 0.3.1

Depends R (>= 4.0.0)

Imports stats

Suggests goftest, knitr, MASS, ordinal, PResiduals, rmarkdown, rms, tinyplot, tinytest, VGAM

License GPL (>= 2)

URL https://github.com/bgreenwell/sure, https://bgreenwell.github.io/sure/, https://bgreenwell.r-universe.dev/sure

BugReports https://github.com/bgreenwell/sure/issues

Encoding UTF-8

RoxygenNote 7.3.3

VignetteBuilder knitr

NeedsCompilation no

Author Brandon Greenwell [aut, cre], Andrew McCarthy [aut], Bradley Boehmke [aut], Liz Glaser [ctb]

Maintainer Brandon Greenwell <greenwell.brandon@gmail.com>

Repository CRAN

Date/Publication 2026-10-24 23:40:02 UTC

Contents

calc_resid 2
gof 3
plot.resid 4
resids 7
sim_data 9
surrogate 10

Index **13**

`calc_resid` *Calculate surrogate residuals for multinomial/discrete choice models*

Description

Calculate surrogate residuals for multinomial logit and general discrete choice models using a fast, exact conditional Gumbel simulation method.

Usage

```
calc_resid(y, proba)
```

Arguments

`y`	Vector of observed classes (either a factor or a 1-based integer vector).
`proba`	Matrix of predicted probabilities, where rows correspond to observations and columns correspond to classes.

Value

A matrix of surrogate residuals of the same dimensions as `proba`.

gof *Goodness-of-Fit Simulation*

Description

Simulate p-values from a goodness-of-fit test.

Usage

```
gof(object, nsim = 10, test = c("ks", "ad", "cvm"), ...)

## Default S3 method:
gof(object, nsim = 10, test = c("ks", "ad", "cvm"), ...)

## S3 method for class 'gof'
plot(x, ...)
```

Arguments

object	An object of class ordinal::clm(), stats::glm(), rms::lrm(), rms::orm(), MASS::polr(), or VGAM::vglm().
nsim	Integer specifying the number of bootstrap replicates to use.
test	Character string specifying which goodness-of-fit test to use. Current options include: "ks" for the Kolmogorov-Smirnov test, "ad" for the Anderson-Darling test, and "cvm" for the Cramer-Von Mises test. Default is "ks".
...	Additional optional arguments. (Currently ignored.)
x	An object of class "gof".

Details

Under the null hypothesis, the distribution of the p-values should appear uniformly distributed on the interval [0, 1]. This can be visually investigated using the plot method. A 45 degree line is indicative of a "good" fit.

Value

A numeric vector of class "gof", "numeric" containing the simulated p-values.

Examples

```
# See ?resids for an example
?resids
```

plot.resid *Plot Residual Diagnostics*

Description

Residual-based diagnostic plots for cumulative link and general regression models using `tinyplot::tinyplot()` graphics.

Usage

```
## S3 method for class 'resid'
plot(
  x,
  y = NULL,
  what = c("qq", "fitted", "covariate"),
  covariate = NULL,
  fit = NULL,
  distribution = NULL,
  ncol = NULL,
  alpha = 1,
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  smooth = TRUE,
  ...
)

## S3 method for class 'clm'
plot(
  x,
  y = NULL,
  what = c("qq", "fitted", "covariate"),
  covariate = NULL,
  fit = NULL,
  distribution = NULL,
  ncol = NULL,
  alpha = 1,
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  smooth = TRUE,
  ...
)

## S3 method for class 'lrm'
plot(
  x,
```

```
  y = NULL,
  what = c("qq", "fitted", "covariate"),
  covariate = NULL,
  fit = NULL,
  distribution = NULL,
  ncol = NULL,
  alpha = 1,
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  smooth = TRUE,
  ...
)

## S3 method for class 'orm'
plot(
  x,
  y = NULL,
  what = c("qq", "fitted", "covariate"),
  covariate = NULL,
  fit = NULL,
  distribution = NULL,
  ncol = NULL,
  alpha = 1,
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  smooth = TRUE,
  ...
)

## S3 method for class 'polr'
plot(
  x,
  y = NULL,
  what = c("qq", "fitted", "covariate"),
  covariate = NULL,
  fit = NULL,
  distribution = NULL,
  ncol = NULL,
  alpha = 1,
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  smooth = TRUE,
  ...
)
```

```
## S3 method for class 'vglm'
plot(
  x,
  y = NULL,
  what = c("qq", "fitted", "covariate"),
  covariate = NULL,
  fit = NULL,
  distribution = NULL,
  ncol = NULL,
  alpha = 1,
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  smooth = TRUE,
  ...
)
```

Arguments

x	An object of class "resid" (returned by resids()), or a fitted model object of class ordinal::clm(), rms::lrm(), rms::orm(), MASS::polr(), or VGAM::vglm(). For stats::glm() fits, use plot(resids(fit), fit = fit).
y	Ignored.
what	Character string specifying what to plot. Default is "qq" which produces a quantile-quantile plot of the residuals.
covariate	A vector giving the covariate values to use for residual-by- covariate plots (i.e., when what = "covariate").
fit	The fitted model from which the residuals were extracted. Required if what = "fitted" and x inherits from class "resid"; when supplied, it also determines the default reference distribution for the quantile-quantile plot.
distribution	Function that computes the quantiles for the reference distribution to use in the quantile-quantile plot. Default is NULL, which uses U(-0.5, 0.5) for residuals obtained by jittering on the probability scale, the distribution implied by the link function of fit when it is supplied, and qnorm otherwise.
ncol	Integer specifying the number of columns to use for the plot layout (if requesting multiple plots). Default is NULL.
alpha	A single value in the interval [0, 1] controlling the opacity alpha of the plotted points. Only used when nsim > 1.
xlab	Character string giving the text to use for the x-axis label. Default is NULL.
ylab	Character string giving the text to use for the y-axis label. Default is NULL.
main	Character string giving the plot title. Default is NULL.
smooth	Logical indicating whether or not to add a nonparametric smooth to certain plots. Default is TRUE.
...	Additional optional arguments to be passed onto tinyplot::tinyplot() (e.g., col, pch, cex, etc.) or resids() (if calling the plot method directly on a fitted model object).

Value

No return value, called for plotting.

Note

For `vglm` (and `vgam`) objects, the **VGAM** package defines its own S4 `plot` method which will override this S3 method if **VGAM** is loaded. In such cases, we recommend extracting the residuals first using `resids()` and then plotting the resulting `resid` object (e.g. `plot(resids(fit), fit = fit)`), which will correctly dispatch to `plot.resid()`.

There is no `plot()` method for `glm` objects, since it would replace the standard diagnostic plots from `stats::plot.lm()`. Use `plot(resids(fit), fit = fit)` instead.

Examples

```
set.seed(101)
df <- sim_data(n = 500, type = "quadratic")
fit <- glm(y ~ x, data = df, family = binomial(link = "probit"))
if (requireNamespace("tinyplot", quietly = TRUE)) {
  plot(resids(fit), fit = fit, what = c("qq", "covariate"),
       covariate = df$x)
}
```

---

resids *Surrogate residuals*

Description

Simulate surrogate response values for cumulative link regression models using the latent method described in Liu and Zhang (2017).

Usage

```
resids(object, ...)

## Default S3 method:
resids(
  object,
  nsim = 1L,
  method = c("latent", "jitter"),
  jitter_scale = c("response", "probability"),
  ...
)

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

## S3 method for class 'matrix'
resids(object, y, ...)
```

Arguments

object	An object of class ordinal::clm(), stats::glm(), rms::lrm(), rms::orm(), MASS::polr(), or VGAM::vglm(). Or a matrix of fitted probabilities (e.g., from a multinomial model).
...	Additional optional arguments. (Currently ignored.)
nsim	Integer specifying the number of bootstrap replicates to use. Default is 1L meaning no bootstrap samples.
method	Character string specifying which method to use to generate the surrogate response values. Current options are "latent" and "jitter". Default is "latent".
jitter_scale	Character string specifyint the scale on which to perform the jittering whenever method = "jitter". Current options are "response" and "probability". Default is "response".
y	A vector of the response variable (only used when object is a matrix of probabilities).

Value

A numeric vector of class c("numeric", "surrogate") containing the simulated surrogate response values. Additionally, if nsim > 1, then the result will contain the attributes:

- boot_reps: A matrix with nsim columns, one for each bootstrap replicate of the surrogate values. Note, these are random and do not correspond to the original ordering of the data.
- boot_id: A matrix with nsim columns. Each column contains the observation number each surrogate value corresponds to in boot_reps. (This is used for plotting purposes.)

Note

Surrogate response values require sampling from a continuous distribution; consequently, the result will be different with every call to surrogate. The internal functions used for sampling from truncated distributions are based on modified versions of truncdist::rtrunc() and truncdist::qtrunc().

For "glm" objects, only the binomial() family is supported.

References

Liu, Dungang and Zhang, Heping. Residuals and Diagnostics for Ordinal Regression Models: A Surrogate Approach. *Journal of the American Statistical Association*, 113(522), 845-854, 2018. doi:10.1080/01621459.2017.1292915

Nadarajah, Saralees and Kotz, Samuel. R Programs for Truncated Distributions. *Journal of Statistical Software, Code Snippets*, 16(2), 1-8, 2006. doi:10.18637/jss.v016.c02

Examples

```
# Generate data from a quadratic probit model
set.seed(101)
n <- 2000
x <- runif(n, min = -3, max = 6)
z <- 10 + 3 * x - 1 * x^2 + rnorm(n)
```

```
y <- ifelse(z <= 0, yes = 0, no = 1)

# Scatterplot matrix
pairs(~ x + y + z)

# Setup for side-by-side plots
par(mfrow = c(1, 2))

# Misspecified mean structure
fm1 <- glm(y ~ x, family = binomial(link = "probit"))
scatter.smooth(x, y = resids(fm1),
               main = "Misspecified model",
               ylab = "Surrogate residual",
               lpars = list(lwd = 3, col = "red2"))
abline(h = 0, lty = 2, col = "blue2")

# Correctly specified mean structure
fm2 <- glm(y ~ x + I(x ^ 2), family = binomial(link = "probit"))
scatter.smooth(x, y = resids(fm2),
               main = "Correctly specified model",
               ylab = "Surrogate residual",
               lpars = list(lwd = 3, col = "red2"))
abline(h = 0, lty = 2, col = "blue2")
```

`sim_data` *Simulate Data for Diagnostic Plots*

Description

Generate simulated data sets designed to illustrate various model misspecifications and diagnostics for ordinal regression models, as described in Liu and Zhang (2017).

Usage

```
sim_data(
  n = 2000,
 type = c("quadratic", "heteroscedastic", "gumbel", "proportionality", "interaction"),
  ...
)
```

Arguments

`n`	Integer specifying the number of observations to simulate. Default is `2000`.
`type`	Character string specifying the type of data/model to simulate. Default is `"quadratic"` (linear relation with a quadratic trend). Other options include: `"heteroscedastic"` (non-constant variance), `"gumbel"` (Gumbel error distribution), `"proportionality"` (non-proportional hazards), and `"interaction"` (interaction effect).
`...`	Additional optional arguments (currently ignored).

Value

A data frame containing the simulated predictor(s) and the ordered factor response variable `y`.

References

Liu, Dungang and Zhang, Heping. Residuals and Diagnostics for Ordinal Regression Models: A Surrogate Approach. *Journal of the American Statistical Association* (accepted).

Examples

```
# Simulate quadratic data
set.seed(101)
df <- sim_data(n = 500, type = "quadratic")
head(df)
```

`surrogate` *Surrogate response*

Description

Simulate surrogate response values for cumulative link regression models using the latent method described in Liu and Zhang (2017).

Usage

```
surrogate(
  object,
  nsim = 1L,
  method = c("latent", "jitter"),
  jitter_scale = c("response", "probability"),
  ...
)
```

Arguments

`object`	An object of class `ordinal::clm()`, `stats::glm()` `rms::lrm()`, `rms::orm()`, `MASS::polr()`, or `VGAM::vglm()`.
`nsim`	Integer specifying the number of bootstrap replicates to use. Default is `1L` meaning no bootstrap samples.
`method`	Character string specifying which method to use to generate the surrogate response values. Current options are `"latent"` and `"jitter"`. Default is `"latent"`.
`jitter_scale`	Character string specifyint the scale on which to perform the jittering whenever `method = "jitter"`. Current options are `"response"` and `"probability"`. Default is `"response"`.
`...`	Additional optional arguments. (Currently ignored.)

Value

A numeric vector of class `c("numeric", "surrogate")` containing the simulated surrogate response values. Additionally, if `nsim` > 1, then the result will contain the attributes:

- `boot_reps`: A matrix with `nsim` columns, one for each bootstrap replicate of the surrogate values. Note, these are random and do not correspond to the original ordering of the data.
- `boot_id`: A matrix with `nsim` columns. Each column contains the observation number each surrogate value corresponds to in `boot_reps`. (This is used for plotting purposes.)

Note

Surrogate response values require sampling from a continuous distribution; consequently, the result will be different with every call to `surrogate`. The internal functions used for sampling from truncated distributions are based on modified versions of `truncdist::rtrunc()` and `truncdist::qtrunc()`.

For `"glm"` objects, only the `binomial()` family is supported.

References

Liu, Dungang and Zhang, Heping. Residuals and Diagnostics for Ordinal Regression Models: A Surrogate Approach. *Journal of the American Statistical Association*, 113(522), 845-854, 2018. doi:10.1080/01621459.2017.1292915

Nadarajah, Saralees and Kotz, Samuel. R Programs for Truncated Distributions. *Journal of Statistical Software, Code Snippets*, 16(2), 1-8, 2006. doi:10.18637/jss.v016.c02

Examples

```
# Generate data from a quadratic probit model
set.seed(101)
n <- 2000
x <- runif(n, min = -3, max = 6)
z <- 10 + 3*x - 1*x^2 + rnorm(n)
y <- ifelse(z <= 0, yes = 0, no = 1)

# Scatterplot matrix
pairs(~ x + y + z)

# Setup for side-by-side plots
par(mfrow = c(1, 2))

# Misspecified mean structure
fm1 <- glm(y ~ x, family = binomial(link = "probit"))
s1 <- surrogate(fm1)
scatter.smooth(x, s1 - fm1$linear.predictors,
               main = "Misspecified model",
               ylab = "Surrogate residual",
               lpars = list(lwd = 3, col = "red2"))
abline(h = 0, lty = 2, col = "blue2")

# Correctly specified mean structure
fm2 <- glm(y ~ x + I(x ^ 2), family = binomial(link = "probit"))
```

```
s2 <- surrogate(fm2)
scatter.smooth(x, s2 - fm2$linear.predictors,
               main = "Correctly specified model",
               ylab = "Surrogate residual",
               lpars = list(lwd = 3, col = "red2"))
abline(h = 0, lty = 2, col = "blue2")
```

Index

calc_resid, 2

gof, 3

MASS::polr(), *3*, *6*, *8*, *10*

ordinal::clm(), *3*, *6*, *8*, *10*

plot.clm *(*plot.resid*)*, 4
plot.gof *(*gof*)*, 3
plot.lrm *(*plot.resid*)*, 4
plot.orm *(*plot.resid*)*, 4
plot.polr *(*plot.resid*)*, 4
plot.resid, 4
plot.vglm *(*plot.resid*)*, 4

resids, 7
resids(), *6*
rms::lrm(), *3*, *6*, *8*, *10*
rms::orm(), *3*, *6*, *8*, *10*

sim_data, 9
stats::glm(), *3*, *6*, *8*, *10*
stats::plot.lm(), *7*
surrogate, 10

tinyplot::tinyplot(), *4*, *6*

VGAM::vglm(), *3*, *6*, *8*, *10*