

# Package ‘prLogistic’

June 19, 2026

**Type** Package

**Title** Estimation of Prevalence Ratios via Logistic Regression Models

**Version** 2.0.2

**Date** 2026-06-12

**Description** Estimates adjusted prevalence ratios (PR) and their confidence intervals from logistic regression models, addressing the well-known limitation of odds ratios (OR) as approximations to PR in cross-sectional studies with common outcomes. Supports independent observations (`glm()`), clustered/multilevel data (`glmer()` from 'lme4'), longitudinal data via Generalised Estimating Equations (`geeglm()` from 'geepack'), and complex survey designs (`svyglm()` from 'survey'). Inference is available via the delta method (conditional and marginal standardisation) and via bootstrap (normal-approximation and percentile intervals). Continuous covariates are handled through user-specified or median-based reference values; flexible baseline specification allows any reference category to be chosen for factor predictors. Based on the methodology described in Amorim & Ospina (2021) [doi:10.1590/0001-3765202120190316](https://doi.org/10.1590/0001-3765202120190316).

**License** GPL (>= 2)

**URL** <https://github.com/Raydonal/prLogistic>,  
<https://raydonal.github.io/prLogistic/>

**BugReports** <https://github.com/Raydonal/prLogistic/issues>

**Encoding** UTF-8

**LazyData** true

**LazyDataCompression** xz

**Depends** R (>= 4.1.0)

**Imports** boot, lme4, stats, graphics

**Suggests** geepack, survey, MASS, testthat (>= 3.0.0), knitr, rmarkdown, ggplot2, dplyr

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Config/roxygen2/version** 8.0.0

**NeedsCompilation** no

**Author** Raydonal Ospina [aut, cre] (ORCID:  
<https://orcid.org/0000-0002-9884-9090>),  
 Leila D. Amorim [aut] (ORCID: <https://orcid.org/0000-0002-1112-2332>)

**Maintainer** Raydonal Ospina <raydonal@de.ufpe.br>

**Repository** CRAN

**Date/Publication** 2026-06-19 12:00:02 UTC

## Contents

coef.prLogistic . . . . .	2
confint.prLogistic . . . . .	3
downer . . . . .	3
LBW . . . . .	4
plot.prLogistic . . . . .	5
print.prLogistic . . . . .	6
prLogisticBootCond . . . . .	7
prLogisticBootMarg . . . . .	8
prLogisticDelta . . . . .	10
prLogisticGEE . . . . .	12
prLogisticSurvey . . . . .	14
summary.prLogistic . . . . .	16
Thailand . . . . .	16
titanic . . . . .	17
Toenail . . . . .	18
UIS . . . . .	19
<b>Index</b>	<b>21</b>

---

coef.prLogistic	<i>Extract prevalence ratio point estimates</i>
-----------------	---

---

### Description

Extract prevalence ratio point estimates

### Usage

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

**Arguments**

object            A prLogistic object.  
 ...                Currently ignored.

**Value**

Named numeric vector of PR estimates.

---

confint.prLogistic     *Extract confidence intervals for prevalence ratios*

---

**Description**

Extract confidence intervals for prevalence ratios

**Usage**

```
## S3 method for class 'prLogistic'
confint(object, parm, level, type = "percentile", ...)
```

**Arguments**

object            A prLogistic object.  
 parm              Ignored (all parameters returned).  
 level             Ignored (level is stored in the object).  
 type              For bootstrap objects: "normal" or "percentile".  
 ...                Currently ignored.

**Value**

Numeric matrix with lower and upper bounds.

---

downer                *Downer Cow Survival Data*

---

**Description**

Veterinary study of downer cows (cattle unable to rise after calving). The outcome is whether the animal survived to discharge.

**Usage**

downer

**Format**

A data frame with 216 rows and 5 variables:

**AST** Aspartate aminotransferase (enzyme marker): 0 = normal, 1 = elevated.

**CK** Creatine kinase (enzyme marker): 0 = normal, 1 = elevated.

**Calving** Whether the downer condition was related to calving: 0 = No, 1 = Yes.

**Myopathy** Presence of myopathy: factor with levels "No", "Yes".

**Survival** Outcome: factor with levels "Died", "Survived". Binary outcome.

**Source**

Dohoo, I., Martin, W. & Stryhn, H. (2003). *Veterinary Epidemiologic Research*. AVC Inc., Prince Edward Island, Canada.

**Examples**

```
data(downer)
prop.table(table(downer$Survival))

fit <- glm(as.integer(Survival == "Survived") ~ Myopathy + AST + CK + Calving,
           family = binomial, data = downer)
prLogisticDelta(fit)
```

---

 LBW

*Low Birth Weight – Longitudinal Study (Salvador, Brazil)*

---

**Description**

Data from a longitudinal study of 244 mothers followed during two pregnancies in Salvador, Bahia, Brazil. The outcome is whether the newborn had low birth weight (< 2500 g). The study illustrates clustered binary data (two births per mother) and is the primary motivating example in Amorim & Ospina (2021).

**Usage**

LBW

**Format**

A data frame with 488 rows and 6 variables:

**ID** Mother identifier (integer).

**birth** Birth order within mother: 1 or 2.

**smoke** Maternal smoking during pregnancy: factor with levels "No", "Yes".

**race** Maternal race: factor with levels "White", "Non-white".

**age** Maternal age at delivery (years, centred).

**low** Birth weight category: factor with levels "Normal" ( $\geq 2500$  g), "Low" (< 2500 g). This is the binary outcome of interest.

## Details

The dataset contains repeated observations: each mother contributes two records (one per birth). Models should account for this clustering – either with a random intercept (glmer) or via GEE (geeglm).

Prevalence of low birth weight across both births: approximately 18%.

## Source

Amorim, L. D. & Ospina, R. (2021). Prevalence ratio estimation using R. *Anais da Academia Brasileira de Ciencias*, **93**(4), e20190316. doi:[10.1590/00013765202120190316](https://doi.org/10.1590/00013765202120190316)

## Examples

```
data(LBW)
table(LBW$low, LBW$smoke)

# GEE model accounting for within-mother correlation

library(geepack)
fit_gee <- geeglm(as.integer(low == "Low") ~ smoke + race + age,
                 family = binomial, id = ID,
                 corstr = "exchangeable", data = LBW)
prLogisticGEE(fit_gee)
```

---

plot.prLogistic      *Forest plot of prevalence ratios*

---

## Description

Produces a simple forest plot (no external dependencies beyond base R).

## Usage

```
## S3 method for class 'prLogistic'
plot(
  x,
  main = NULL,
  xlab = "Prevalence Ratio",
  col = "steelblue",
  ci_col = "steelblue",
  ref_line = TRUE,
  type = "percentile",
  ...
)
```

**Arguments**

<code>x</code>	A <code>prLogistic</code> object.
<code>main</code>	Plot title. If <code>NULL</code> , a default is used.
<code>xlab</code>	x-axis label.
<code>col</code>	Color for the point estimates.
<code>ci_col</code>	Color for the CI lines.
<code>ref_line</code>	Logical: draw a vertical reference line at $PR = 1$ ?
<code>type</code>	For bootstrap objects: "normal" or "percentile".
<code>...</code>	Further graphical parameters passed to <code>plot()</code> .

**Value**

No return value, called for its side effect of drawing a forest plot of the prevalence ratio estimates and their confidence intervals.

---

`print.prLogistic`      *Print a prLogistic object*

---

**Description**

Print a `prLogistic` object

**Usage**

```
## S3 method for class 'prLogistic'
print(x, digits = 4, ...)
```

**Arguments**

<code>x</code>	A <code>prLogistic</code> object.
<code>digits</code>	Number of significant digits (default 4).
<code>...</code>	Currently ignored.

**Value**

Invisibly returns the `prLogistic` object `x`. Called for its side effect of printing a formatted summary of the prevalence ratio estimates and confidence intervals to the console.

## Description

Estimates adjusted prevalence ratios (PR) using conditional standardisation and obtains confidence intervals via bootstrap resampling (normal- approximation and percentile methods).

## Usage

```
prLogisticBootCond(
  fit,
  data,
  conf = 0.95,
  R = 999L,
  ref_values = NULL,
  ref_continuous = c("median", "mean")
)
```

## Arguments

<code>fit</code>	A fitted model object of class <code>glm</code> (binomial family), <code>glmerMod</code> (from <code>lme4::glmer()</code> ), <code>geeglm</code> (from <code>geepack::geeglm()</code> ), or <code>svyglm</code> (from <code>survey::svyglm()</code> ). Must use the logit link.
<code>data</code>	Data frame used to fit <code>fit</code> . Required for bootstrapping.
<code>conf</code>	Numeric scalar in (0, 1): confidence level. Default 0.95.
<code>R</code>	Integer: number of bootstrap replicates. Default 999.
<code>ref_values</code>	Named list of reference values for specific predictors, e.g. <code>list(age = 40, bmi = 25)</code> . Overrides automatic reference-value selection. For factor/dummy predictors the value should be 0 (the default) or 1.
<code>ref_continuous</code>	Character string: how to compute the reference value for continuous predictors when not supplied in <code>ref_values</code> . Either "median" (default) or "mean".

## Details

At each bootstrap replicate the model is refitted on a resampled dataset and conditional PRs are computed. Two CI types are returned:

**Normal** Bootstrap normal-approximation interval.

**Percentile** Empirical quantiles of the bootstrap distribution.

Use `confint.prLogistic()` with `type = "normal"` or `type = "percentile"` to extract a single CI type.

**Value**

An object of class "prLogistic" with components:

table Numeric matrix with columns Estimate, lower and upper CI.

conf Confidence level used.

method "delta".

standardisation "conditional" or "marginal".

model\_type Class of the fitted model.

call The matched call.

**References**

Amorim, L. D. & Ospina, R. (2021). *An Acad Bras Cienc*, **93**(4). doi:10.1590/00013765202120190316

Davison, A. C. & Hinkley, D. V. (1997). *Bootstrap Methods and their Application*. Cambridge University Press.

**See Also**

[prLogisticDelta\(\)](#), [prLogisticBootMarg\(\)](#)

**Examples**

```
fit_glm <- glm(case ~ induced + spontaneous + parity,
              family = binomial, data = infert)

set.seed(42)
res <- prLogisticBootCond(fit_glm, data = infert, R = 199)
print(res)
plot(res)
```

---

prLogisticBootMarg      *Bootstrap CI for Prevalence Ratios – Marginal Standardisation*

---

**Description**

Estimates adjusted prevalence ratios (PR) using marginal standardisation (population-averaged) and obtains confidence intervals via bootstrap resampling.

**Usage**

```
prLogisticBootMarg(
  fit,
  data,
  conf = 0.95,
  R = 999L,
  ref_values = NULL,
  ref_continuous = c("median", "mean")
)
```

**Arguments**

<code>fit</code>	A fitted model object of class <code>glm</code> (binomial family), <code>glmerMod</code> (from <code>lme4::glmer()</code> ), <code>geeglm</code> (from <code>geepack::geeglm()</code> ), or <code>svyglm</code> (from <code>survey::svyglm()</code> ). Must use the logit link.
<code>data</code>	Data frame used to fit <code>fit</code> . Required for bootstrapping.
<code>conf</code>	Numeric scalar in (0, 1): confidence level. Default 0.95.
<code>R</code>	Integer: number of bootstrap replicates. Default 999.
<code>ref_values</code>	Named list of reference values for specific predictors, e.g. <code>list(age = 40, bmi = 25)</code> . Overrides automatic reference-value selection. For factor/dummy predictors the value should be 0 (the default) or 1.
<code>ref_continuous</code>	Character string: how to compute the reference value for continuous predictors when not supplied in <code>ref_values</code> . Either "median" (default) or "mean".

**Details**

Marginal standardisation averages counterfactual predicted probabilities over the empirical covariate distribution, giving a population-averaged PR. At each bootstrap replicate the model is refitted and marginal PRs are recomputed.

**Value**

An object of class "prLogistic" with components:

`table` Numeric matrix with columns Estimate, lower and upper CI.  
`conf` Confidence level used.  
`method` "delta".  
`standardisation` "conditional" or "marginal".  
`model_type` Class of the fitted model.  
`call` The matched call.

**See Also**

[prLogisticDelta\(\)](#), [prLogisticBootCond\(\)](#)

**Examples**

```
fit_glm <- glm(case ~ induced + spontaneous + parity,
              family = binomial, data = infert)

set.seed(42)
res <- prLogisticBootMarg(fit_glm, data = infert, R = 199)
print(res)
```

prLogisticDelta

*Estimate Prevalence Ratios via Logistic Regression – Delta Method***Description**

Estimates adjusted prevalence ratios (PR) and confidence intervals using the delta method, from a fitted logistic regression model. Supports four model types covering independent, clustered, longitudinal and complex-survey data.

**Usage**

```
prLogisticDelta(
  fit,
  standardisation = c("conditional", "marginal"),
  conf = 0.95,
  ref_values = NULL,
  ref_continuous = c("median", "mean")
)
```

**Arguments**

<code>fit</code>	A fitted model object of class <code>glm</code> (binomial family), <code>glmerMod</code> (from <code>lme4::glmer()</code> ), <code>geeglm</code> (from <code>geepack::geeglm()</code> ), or <code>svyglm</code> (from <code>survey::svyglm()</code> ). Must use the logit link.
<code>standardisation</code>	Character string: "conditional" (default) or "marginal". See <i>Details</i> .
<code>conf</code>	Numeric scalar in (0, 1): confidence level. Default 0.95.
<code>ref_values</code>	Named list of reference values for specific predictors, e.g. <code>list(age = 40, bmi = 25)</code> . Overrides automatic reference-value selection. For factor/dummy predictors the value should be 0 (the default) or 1.
<code>ref_continuous</code>	Character string: how to compute the reference value for continuous predictors when not supplied in <code>ref_values</code> . Either "median" (default) or "mean".

**Details****Standardisation procedures:**

**Conditional standardisation** fixes all covariates at their reference values (median/mean for continuous, 0 for binary/dummy) and computes the PR for each predictor by contrasting *exposed* (predictor = 1) vs *unexposed* (predictor = 0) profiles:

$$\widehat{PR}_j = \frac{\text{expit}(\hat{\beta}_0 + \hat{\beta}_j + \sum_{k \neq j} \hat{\beta}_k r_k)}{\text{expit}(\hat{\beta}_0 + \sum_{k \neq j} \hat{\beta}_k r_k)}$$

where  $r_k$  are the reference values of the remaining covariates.

**Marginal standardisation** computes counterfactual prevalences using the observed covariate distribution of the entire sample:

$$\widehat{PR}_j = \frac{n^{-1} \sum_i \text{expit}(\hat{\eta}_i^{(1)})}{n^{-1} \sum_i \text{expit}(\hat{\eta}_i^{(0)})}$$

where  $\hat{\eta}_i^{(1)}$  and  $\hat{\eta}_i^{(0)}$  are the linear predictors with predictor  $j$  set to 1 and 0, respectively.

Variance estimates use the delta method (first-order Taylor expansion) as described in Oliveira et al. (1997) and Amorim & Ospina (2021).

#### **Baseline / reference category:**

By default, the reference level of each factor predictor is determined by the contrasts of the fitted model (typically the first level of the factor()). You can override this using `ref_values` for any predictor column present in the model matrix.

#### **Supported model types:**

Class	Package	Use case
glm	stats	Independent observations
glmerMod	lme4	Clustered / multilevel data
geeglm	geepack	Longitudinal / GEE
svyglm	survey	Complex survey designs

#### **Value**

An object of class "prLogistic" with components:

`table` Numeric matrix with columns Estimate, lower and upper CI.

`conf` Confidence level used.

`method` "delta".

`standardisation` "conditional" or "marginal".

`model_type` Class of the fitted model.

`call` The matched call.

#### **References**

Amorim, L. D. & Ospina, R. (2021). Prevalence ratio estimation using R. *Anais da Academia Brasileira de Ciencias*, **93**(4), e20190316. doi:10.1590/00013765202120190316

Oliveira, N. F., Santana, V. S. & Lopes, A. A. (1997). Razoes de proporcoes e uso da regressao log?stica em estudos transversais. *Revista de Sa?de P?blica*, **31**, 90-99.

Wilcosky, T. C. & Chambless, L. E. (1985). A comparison of direct adjustment and regression adjustment of epidemiologic measures. *Journal of Chronic Diseases*, **38**, 849-856.

#### **See Also**

[prLogisticBootCond\(\)](#), [prLogisticBootMarg\(\)](#), [prLogisticGEE\(\)](#), [prLogisticSurvey\(\)](#)

## Examples

```
# --- Independent observations (glm) --- infert is a built-in dataset ----
# outcome: case (spontaneous abortion), prevalence ~33%
fit_glm <- glm(case ~ induced + spontaneous + parity,
              family = binomial, data = infert)

# Conditional PR (continuous covariates at median)
prLogisticDelta(fit_glm, standardisation = "conditional")

# Marginal PR
prLogisticDelta(fit_glm, standardisation = "marginal")

# Custom reference values
prLogisticDelta(fit_glm,
                standardisation = "conditional",
                ref_values = list(parity = 2))

# --- Clustered data (glmer) -----
library(lme4)
fit_glmer <- glmer(case ~ induced + spontaneous + (1 | stratum),
                  family = binomial, data = infert)
prLogisticDelta(fit_glmer, standardisation = "marginal")

# --- Longitudinal / GEE -----
library(geepack)
data(ohio, package = "geepack")
fit_gee <- geeglm(resp ~ smoke + age,
                  family = binomial,
                  id = id,
                  corstr = "exchangeable",
                  data = ohio)
prLogisticDelta(fit_gee, standardisation = "marginal")

# --- Complex survey design -----
library(survey)
data(api, package = "survey")
dclus2 <- svydesign(id = ~dnum + snum, fpc = ~fpc1 + fpc2, data = apiclus2)
fit_svy <- svyglm(sch.wide ~ meals + stype,
                 design = dclus2, family = quasibinomial)
prLogisticDelta(fit_svy, standardisation = "conditional")
```

## Description

A convenience wrapper around `prLogisticDelta()` for models fitted with `geepack::geeglm()`. GEE provides population-averaged (marginal) estimates suitable for longitudinal or clustered binary

outcomes.

### Usage

```
prLogisticGEE(
  fit,
  standardisation = c("marginal", "conditional"),
  conf = 0.95,
  method = c("delta", "bootstrap"),
  data = NULL,
  R = 999L,
  ref_values = NULL,
  ref_continuous = c("median", "mean")
)
```

### Arguments

fit	A <code>geeglm</code> object fitted with <code>family = binomial</code> and <code>link = "logit"</code> (or <code>family = binomial(link = "logit")</code> ).
standardisation	Character: "marginal" (default, recommended for GEE) or "conditional".
conf	Confidence level. Default 0.95.
method	Inference method: "delta" (default) or "bootstrap".
data	Data frame (required when <code>method = "bootstrap"</code> ).
R	Number of bootstrap replicates (only used when <code>method = "bootstrap"</code> ). Default 999L.
ref_values	Named list of reference values. See <a href="#">prLogisticDelta()</a> .
ref_continuous	"median" (default) or "mean".

### Details

GEE accounts for within-subject correlation through a working correlation structure (`const` argument of `geeglm()`). Common choices:

"independence" No correlation assumed (equivalent to GLM).

"exchangeable" Constant correlation across time points.

"ar1" First-order autoregressive; suitable for ordered time.

"unstructured" Estimates all pairwise correlations freely.

The robust (sandwich) variance-covariance matrix returned by `vcov()` on a `geeglm` object is used automatically, giving valid inference even when the working correlation structure is misspecified.

### Value

A "prLogistic" object. See [prLogisticDelta\(\)](#).

## References

Zeger, S. L. & Liang, K.-Y. (1986). Longitudinal data analysis for discrete and continuous outcomes. *Biometrics*, **42**, 121-130.

Højsgaard, S., Halekoh, U. & Yan, J. (2006). The R package geepack for generalised estimating equations. *Journal of Statistical Software*, **15**(2), 1-11.

Amorim, L. D. & Ospina, R. (2021). *An Acad Bras Cienc*, **93**(4). doi:10.1590/00013765202120190316

## See Also

[prLogisticDelta\(\)](#), [geepack::geeglm\(\)](#)

## Examples

```
library(geepack)
data(ohio, package = "geepack")

# Model respiratory symptoms over time with exchangeable correlation
fit_gee <- geeglm(
  resp ~ smoke + age,
  family = binomial,
  id = id,
  corstr = "exchangeable",
  data = ohio
)

# Marginal PR (recommended for GEE)
prLogisticGEE(fit_gee)

# With bootstrap CIs (small R for a fast example; use R >= 999 in practice)
prLogisticGEE(fit_gee, method = "bootstrap", data = ohio, R = 25)
```

---

prLogisticSurvey

*Prevalence Ratios for Complex Survey Data*

---

## Description

A convenience wrapper around [prLogisticDelta\(\)](#) for logistic regression models fitted on complex survey data using [survey::svyglm\(\)](#).

## Usage

```
prLogisticSurvey(
  fit,
  standardisation = c("conditional", "marginal"),
  conf = 0.95,
  ref_values = NULL,
  ref_continuous = c("median", "mean")
)
```

## Arguments

fit	A svyglm object fitted with family = quasibinomial (or binomial) and the logit link.
standardisation	Character: "conditional" (default) or "marginal".
conf	Confidence level. Default 0.95.
ref_values	Named list of reference values. See <a href="#">prLogisticDelta()</a> .
ref_continuous	"median" (default) or "mean".

## Details

svyglm() incorporates sampling weights and complex design features (stratification, clustering, finite-population corrections) into parameter estimation. The design-consistent variance-covariance matrix is extracted automatically via vcov() and used in the delta-method calculations.

**Note:** bootstrap resampling for survey data requires design-aware resampling (e.g., survey bootstrap, balanced repeated replication). This is currently not automated; use [prLogisticDelta\(\)](#) with a bootstrap-replicate survey design if needed.

## Value

A "prLogistic" object. See [prLogisticDelta\(\)](#).

## References

Lumley, T. (2004). Analysis of complex survey samples. *Journal of Statistical Software*, **9**(1), 1-19.

Lumley, T. (2010). *Complex Surveys: A Guide to Analysis Using R*. Wiley, New Jersey.

Amorim, L. D. & Ospina, R. (2021). *An Acad Bras Cienc*, **93**(4). doi:10.1590/00013765202120190316

## See Also

[prLogisticDelta\(\)](#), [survey::svyglm\(\)](#)

## Examples

```
library(survey)
data(api, package = "survey")

# Create binary outcome
apiclus2$target_met <- as.numeric(apiclus2$sch.wide == "Yes")

# Stratified two-stage cluster sample
dclus2 <- svydesign(
  id = ~dnum + snum,
  fpc = ~fpc1 + fpc2,
  data = apiclus2
)

fit_svy <- svyglm(
```

```

target_met ~ meals + stype,
design = dclus2,
family = quasibinomial
)

prLogisticSurvey(fit_svy, standardisation = "conditional")
prLogisticSurvey(fit_svy, standardisation = "marginal")

```

---

summary.prLogistic      *Summarise a prLogistic object*

---

### Description

Summarise a prLogistic object

### Usage

```

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

```

### Arguments

object	A prLogistic object.
...	Currently ignored.

### Value

Invisibly returns the prLogistic object. Called for its side effect of printing the model call followed by the formatted estimates.

---

Thailand      *Thailand Education Study – Clustered Binary Data*

---

### Description

Data from a survey of primary school students in Thailand. The outcome is whether the student repeated a grade (rgi). Students are nested within schools, making this a clustered binary outcome dataset.

### Usage

Thailand

**Format**

A data frame with 8582 rows and 4 variables:

**schoolid** School identifier (integer). There are 411 schools.

**sex** Student sex: factor with levels "Girl", "Boy".

**pped** Pre-primary education: factor with levels "No", "Yes".

**rgi** Repeated a grade: factor with levels "No", "Yes". Binary outcome of interest.

**Details**

Prevalence of grade repetition is approximately 16%, making PR a more appropriate measure than OR. The clustering by school should be accounted for with `glmer` or `geeglm`.

**Source**

Raudenbush, S. W. & Bryk, A. S. (2002). *Hierarchical Linear Models*, 2nd ed. Sage.

Amorim, L. D. & Ospina, R. (2021). *An Acad Bras Cienc*, **93**(4). doi:10.1590/00013765202120190316

**Examples**

```
data(Thailand)
prop.table(table(Thailand$rgi))

# Mixed model (random intercept per school)

library(lme4)
fit_ml <- glmer(as.integer(rgi == "Yes") ~ sex + pped + (1 | schoolid),
               family = binomial, data = Thailand)
prLogisticDelta(fit_ml, standardisation = "marginal")
```

---

titanic

*Titanic Passenger Survival*

---

**Description**

Survival data for 1307 passengers aboard the RMS Titanic. The outcome is whether the passenger survived.

**Usage**

```
titanic
```

**Format**

A data frame with 1307 rows and 4 variables:

**pclass** Passenger class: factor with levels "1", "2", "3".

**survived** Survived: factor with levels "No", "Yes". Binary outcome.

**sex** Sex: factor with levels "Female", "Male".

**embarked** Port of embarkation: 0 = Southampton, 1 = Cherbourg/ Queenstown.

**Details**

Overall survival rate is approximately 38%, making this a common outcome – a setting where OR meaningfully diverges from PR.

**Source**

Harrell, F. E. (2001). *Regression Modeling Strategies*. Springer, New York.

**Examples**

```
data(titanic)
prop.table(table(titanic$survived, titanic$sex), margin = 2)

fit <- glm(as.integer(survived == "Yes") ~ sex + pclass,
           family = binomial, data = titanic)

# OR vs PR comparison
OR <- exp(coef(fit))
PR <- prlogisticDelta(fit, standardisation = "marginal")
print(PR)
```

---

Toenail

*Toenail Infection Trial – Longitudinal Binary Outcome*

---

**Description**

Data from a randomised clinical trial comparing two oral antifungal treatments (itraconazole vs terbinafine) for toenail dermatophyte infection. Patients were measured at up to 7 visits over 18 months.

**Usage**

Toenail

**Format**

A data frame with 1908 rows and 5 variables:

**ID** Patient identifier. There are 294 patients.

**Response** Presence of moderate or severe onycholysis (nail separation): factor with levels "Not moderate/severe", "Moderate/severe". Binary outcome.

**Treatment** Antifungal treatment: factor with levels "Itraconazole", "Terbinafine".

**Month** Time since randomisation (months, continuous).

**Visit** Visit number (1 to 7, integer).

**Details**

The dataset illustrates a longitudinal binary outcome with dropout (not all patients have 7 visits). GEE with an unstructured or exchangeable correlation is commonly used.

**Source**

De Backer, M. et al. (1998). Twelve weeks of continuous oral therapy for toenail onychomycosis caused by dermatophytes. *Journal of the American Academy of Dermatology*, **38**, S57-S63.

**Examples**

```
data(Toenail)
table(Toenail$Response, Toenail$Treatment)

library(geepack)
Toenail$resp_bin <- as.integer(Toenail$Response == "Moderate/severe")
fit_gee <- geeglm(resp_bin ~ Treatment + Month,
                 family = binomial, id = ID,
                 corstr = "exchangeable", data = Toenail)
prLogisticGEE(fit_gee)
```

**Description**

Data from the University of Massachusetts AIDS Research Unit (UMARU) Impact Study, a 5-year study comparing two residential treatment programmes for drug abuse.

**Usage**

UIS

**Format**

A data frame with 575 rows and 7 variables:

**ID** Patient identifier.

**Age** Age at enrolment (years, centred).

**DrugUse** History of intravenous drug use: factor with levels "Short" ( $\leq 3$  years), "Long" ( $> 3$  years).

**race** Race: factor with levels "White", "Other".

**trt** Treatment assignment: factor with levels "Short" (3-month), "Long" (6-month).

**site** Treatment site: factor with levels "A", "B".

**drugFree** Drug-free at 6 months: factor with levels "No", "Yes". Binary outcome.

**Source**

Hosmer, D. W. & Lemeshow, S. (2000). *Applied Logistic Regression*, 2nd ed. Wiley, New York.

**Examples**

```
data(UIS)
prop.table(table(UIS$drugFree))

fit <- glm(as.integer(drugFree == "Yes") ~ trt + Age + DrugUse + race + site,
           family = binomial, data = UIS)
prLogisticDelta(fit, standardisation = "conditional")
```

# Index

## \* datasets

- downer, 3
- LBW, 4
- Thailand, 16
- titanic, 17
- Toenail, 18
- UIS, 19

- coef.prLogistic, 2
- confint.prLogistic, 3
- confint.prLogistic(), 7

- downer, 3

- geepack::geeglm(), 7, 9, 10, 12, 14

- LBW, 4

- lme4::glmer(), 7, 9, 10

- plot.prLogistic, 5
- print.prLogistic, 6
- prLogisticBootCond, 7
- prLogisticBootCond(), 9, 11
- prLogisticBootMarg, 8
- prLogisticBootMarg(), 8, 11
- prLogisticDelta, 10
- prLogisticDelta(), 8, 9, 12–15
- prLogisticGEE, 12
- prLogisticGEE(), 11
- prLogisticSurvey, 14
- prLogisticSurvey(), 11

- summary.prLogistic, 16
- survey::svyglm(), 7, 9, 10, 14, 15

- Thailand, 16
- titanic, 17
- Toenail, 18

- UIS, 19