

# Package: gofcat (via r-universe)

November 9, 2024

**Type** Package

**Title** Goodness-of-Fit Measures for Categorical Response Models

**Version** 0.1.3

**Description** A post-estimation method for categorical response models (CRM). Inputs from objects of class `serp()`, `clm()`, `polr()`, `multinom()`, `mlogit()`, `vglm()` and `glm()` are currently supported. Available tests include the Hosmer-Lemeshow tests for the binary, multinomial and ordinal logistic regression; the Lipsitz and the Pulkstenis-Robinson tests for the ordinal models. The proportional odds, adjacent-category, and constrained continuation-ratio models are particularly supported at ordinal level. Tests for the proportional odds assumptions in ordinal models are also possible with the Brant and the Likelihood-Ratio tests. Moreover, several summary measures of predictive strength (Pseudo R-squared), and some useful error metrics, including, the brier score, misclassification rate and logloss are also available for the binary, multinomial and ordinal models. Ugba, E. R. and Gertheiss, J. (2018) <[http://www.statmod.org/workshops\\_archive\\_proceedings\\_2018.html](http://www.statmod.org/workshops_archive_proceedings_2018.html)>.

**License** GPL-2 | file LICENSE

**URL** <https://github.com/ejikeugba/gofcat>

**BugReports** <https://github.com/ejikeugba/gofcat/issues>

**Depends** R (>= 3.2.0)

**Imports** utils, crayon, stats, Matrix, epiR, reshape, stringr, VGAM (>= 1.1-4)

**Encoding** UTF-8

**LazyData** True

**RoxygenNote** 7.1.2

**Suggests** serp, dfidx, mlogit, nnet, ordinal, MASS, testthat, covr

```
Config/pak/sysreqs libcairo2-dev libfontconfig1-dev libfreetype6-dev
libfribidi-dev libgdal-dev gdal-bin libgeos-dev make
libharfbuzz-dev libicu-dev libjpeg-dev libpng-dev libtiff-dev
libxml2-dev libssl-dev libproj-dev libsqlite3-dev
libudunits2-dev

Repository https://ejikeugba.r-universe.dev
RemoteUrl https://github.com/ejikeugba/gofcat
RemoteRef HEAD
RemoteSha 1ef288e95f48f6c933edcac711a70d442f0140d9
```

Contents

brant.test . . . . .	2
erroR . . . . .	4
Fishing . . . . .	5
hosmerlem . . . . .	6
lipsitz . . . . .	8
LR.test . . . . .	10
print.brant . . . . .	11
print.erroR . . . . .	11
print.hosmerlem . . . . .	12
print.lipsitz . . . . .	12
print.LRT . . . . .	13
print.pulkroben . . . . .	13
print.Rsquared . . . . .	14
pulkroben . . . . .	15
retinopathy . . . . .	16
Rsquared . . . . .	17
vaso . . . . .	18
<b>Index</b>	<b>19</b>

---

brant.test	<i>Brant Test of the Proportional Odds Assumption</i>
------------	---

---

Description

Provides the means of testing the parallel regression assumption in the ordinal regression models. Also available is the likelihood ratio test, LR.test().

Usage

```
brant.test(model, global= FALSE, call = FALSE)
```

**Arguments**

model	a single model object to be tested.
global	default to FALSE. When TRUE, a global test is made for the factor variables instead of the individual factor levels.
call	when TRUE the model call is printed alongside test results.

**Details**

The parallel regression assumption for the ordinal regression model can be tested With this function. The brant test (Brant, 1990) is currently available for objects of class: `serp()`, `clm()`, `polr()` and `vglm()`. Objects of class `serp()` should have the `slope` argument set to 'parallel', while objects of class `vglm()` should have the `model` argument TRUE, if not, the model is automatically updated to include the object 'model'. Moreover, family in `vglm()` must be either "cumulative" or "propodds", with the `parallel` argument TRUE.

**Value**

model	call of the model tested
df	the degrees of freedom
global	logical vector of TRUE or FALSE
modeltype	character vector of the class of model tested
Terms	original model terms
vnames	character vector of variable names used in the model
chisq	realized values of the chi-square statistic
rdf	residual degrees of freedom
rDev	residual deviance
prob	the p-values of test
call	a logical vector

**References**

Brant, R. (1990). Assessing proportionality in the proportional odds model for ordinal logistic regression. *Biometrics*, 46, 1171-1178.

**See Also**

[LR.test](#), [hosmerlem](#), [lipsitz](#), [pulkroben](#)

**Examples**

```
require(serp)

set.seed(1)
n <- 200
y <- ordered(rbinom(n, 2, 0.5))
x1 <- factor(rbinom(n, 2, 0.7))
```

```

x2 <- runif(n)

## proportional odds model
sp <- serp(y ~ x1 * x2, link = "logit", slope = "parallel", reverse = TRUE)

brant.test(sp)
brant.test(sp, global = TRUE, call=TRUE)

```

---

erroR

*Performance metrics for categorical models*


---

## Description

Calculates common error metrics of fitted binary and multi-categorical response models. Available measures include: the brier score, logloss and misclassification error.

## Usage

```
erroR(model, type = c("brier", "logloss", "misclass"), thresh = 5e-1)
```

## Arguments

model	a model object or data.frame of observed and predicted values. The following class of objects can be directly passed to the erroR function: glm(), vglm(), serp(), polr(), clm(), mlogit() and multinom(). Other class of objects require providing a data.frame of observed and predicted values.
type	specifies the type of error metrics
thresh	resets the default misclassification threshold

## Value

value	a numeric vector of the realized error value.
type	a character vector of error type.
threshold	a numeric vector of the misclassification threshold.

## See Also

[Rsquared](#)

## Examples

```

require(serp)

set.seed(1)
n <- 100
y <- factor(rbinom(n, 1, 0.3))
x <- rnorm(n)

```

```
#p <- runif(n)

m1 <- glm(y ~ x, family = binomial())

erroR(m1, type = "brier")
erroR(m1, type = "logloss")
erroR(m1, type = "misclass")
erroR(m1, type = "misclass", thresh=0.3)

# using data.frame
df <- data.frame(y, fitted(m1))
erroR(df, type = "brier")

m2 <- serp(rating ~ temp + contact, slope = "parallel", link = "logit",
           data = wine)
erroR(m2, type = "brier")
erroR(m2, type = "logloss")
erroR(m2, type = "misclass")
```

Fishing

*Choice of Fishing Mode***Description**

A sample of 1182 individuals in the United-States for the choice of 4 alternative fishing modes.

**Format**

A data frame containing :

- mode: recreation mode choice, one of : beach, pier, boat and charter, - price.beach: price for beach mode - price.pier: price for pier mode, - price.boat: price for private boat mode, - price.charter: price for charter boat mode, - catch.beach: catch rate for beach mode, - catch.pier: catch rate for pier mode, - catch.boat: catch rate for private boat mode, - catch.charter: catch rate for charter boat mode, - income: monthly income,

**Source**

Cameron A, Trivedi P (2005). Microeconometrics. Cambridge University Press. <https://EconPapers.repec.org/RePEc:cup:cb>

**References**

Herriges JA, Kling CL (1999). "Nonlinear Income Effects in Random Utility Models." The Review of Economics and Statistics, 81(1), 62-72. doi: 10.1162/003465399767923827

**See Also**

[retinopathy](#), [vaso](#)

---

 hosmerlem

*Hosmer-Lemeshow Test for Categorical Response Models*


---

## Description

This is a post estimation goodness-of-fit test for categorical response models. It currently supports the binary, multinomial and ordinal logistic regression models.

## Usage

```
hosmerlem(model, group = 10, tables = FALSE, customFreq = NULL)
```

## Arguments

model	a model object or data.frame of observed and estimated values. The following class of objects can be directly passed to the function: glm(), vglm(), serp(), polr(), clm(), and mlogit(). Other class of objects require providing a data.frame of observed and predicted values.
group	Default to 10. The number of groups to be formed from the original observations.
tables	Default to FALSE. When TRUE, both the observed and the expected frequency tables are printed alongside test results.
customFreq	a vector of custom group frequencies to be used instead of the default equal group frequencies. The vector should, however, sum up to the total number of original observations.

## Details

The implemented tests are discussed in Hosmer and Lemeshow (1980) (for the binary case), Fagerland, Hosmer, and Bofin (2008); Fagerland and Hosmer (2012) (for the multinomial case) and Fagerland and Hosmer (2013, 2016, 2017) (for the ordinal case). In each of the three settings, one makes a split of the original observations into  $k$  groups (10 by default), after ranking the observations by ordinal scores (OS). Ties are separated following an additional ranking based on the observed values. A Pearson chi-squared statistic is thereafter obtained from a  $(k \times c)$  expected and observed frequency tables. The Hosmerlem() function automatically dictates which of the three HL tests to run based on the type of model supplied or by inspecting the class/level of response category in the supplied data.frame of predicted and observed values.

On the choice of  $k$ -groups, Fagerland and Hosmer (2013, 2016) recommend using ten groups. It is believed that number of groups below six adversely affects the heterogeneity within groups thereby resulting in a test with low power. Also having too many groups could lead to a sparsely populated contingency tables, affecting the distributional assumptions of the test statistic. The test statistic follows the chi-squared distribution with  $(k - 2)(r - 1) + (r - 2)$  degrees of freedom. Where  $k$  and  $r$  are respectively the number of groups and response category. For chi-square estimation to hold, it is recommended to have the percentage of estimated frequencies greater than 80 percent of all estimated frequencies (Fagerland and Hosmer, 2017).

Finally, it is particularly recommended to compare the results of the ordinal Hosmer-Lemeshow test with the lipsitz and the Pulkstenis-Robinson tests (Fagerland and Hosmer, 2016; 2017).

**Value**

chi.sq	realized value of the chi-square statistic.
df	the degrees of freedom.
p.value	the p-value of the test.
observed	table of observed frequencies.
expected	table of estimated frequencies.
rho	percentage of estimated frequencies greater than one.
type	a character vector indicating the type of HL-test conducted.
tables	TRUE or FALSE logical vector.

**References**

- Hosmer, D. W. and Lemeshow, S. (1980). Goodness of fit tests for the multiple logistic regression model. *Communications in Statistics-Theory and Methods*, 9, 1043-1069.
- Fagerland, M. W., Hosmer, D. W. and Bofin, A. M. (2008). Multinomial goodness-of-fit tests for logistic regression models. *Statistics in Medicine*, 27, 4238-4253.
- Fagerland, M. W. and Hosmer, D. W. (2012). A generalized Hosmer-Lemeshow goodness-of-fit test for multinomial logistic regression models. *Stata Journal*, 12, 447-453.
- Fagerland, M. W. and Hosmer, D. W. (2013). A goodness-of-fit test for the proportional odds regression model. *Statistics in Medicine*, 32, 2235-2249.
- Fagerland, M. W. and Hosmer, D. W. (2016). Tests for goodness of fit in ordinal logistic regression models. *Journal of Statistical Computation and Simulation*, 86, 3398-3418.
- Fagerland, M. W. and Hosmer, D. W. (2017). How to test for goodness of fit in ordinal logistic regression models. *Stata Journal*, 17, 668-686.

**See Also**

[lipsitz](#), [pulkroben](#), [brant.test](#), [LR.test](#)

**Examples**

```
require(VGAM)

# Binary L-H test
set.seed(1)
gm <- glm(rbinom(100,1,0.5) ~ rnorm(100), family=binomial)
hosmerlem(gm, group = 10, customFreq = NULL, tables = TRUE)

# multinomial L-H test
vg <- vglm(RET ~ DIAB + GH + BP, model = TRUE,
           family = multinomial(parallel = TRUE), data = retinopathy)
hosmerlem(vg)

# ordinal L-H test
## proportional odds model
cu <- update(vg, family = cumulative(link = 'logitlink', parallel = TRUE))
```

```

hosmerlem(cu, tables=TRUE)
hosmerlem(cu, group = 5, tables=TRUE, customFreq = c(rep(100,5), 113))

## adjacent category model
ac <- update(vg, family = acat(parallel = TRUE))
hosmerlem(ac)

## continuation ratio model
cr <- update(vg, family = cratio(parallel = TRUE))
hosmerlem(cr)

## using data.frame
y <- ordered(retinopathy$RET)
df <- data.frame(y, fitted.values(cr))
hosmerlem(df, tables = TRUE)

```

---

lipsitz

*Lipsitz Test for Categorical Response Models*


---

## Description

This provides a post estimation goodness-of-fit test for the ordinal response models. Supported models include the proportional odds, adjacent-category, and constrained continuation-ratio models.

## Usage

```
lipsitz(model, group = 10, customFreq = NULL)
```

## Arguments

model	a model object or data.frame of observed and estimated values. The following class of objects can be directly passed to the <code>lipsitz</code> function: <code>vglm()</code> , <code>serp()</code> , <code>polr()</code> , and <code>clm()</code> . Other class of objects require providing a data.frame of observed and predicted values.
group	Default to 10. The number of groups to be formed from the original observations.
customFreq	a vector of custom group frequencies to be used instead of the default equal group frequencies. The vector should, however, sum up to the total number of original observations.

## Details

Similar to the ordinal Hosmer-Lemeshow test ([hosmerlem](#)), the Lipsitz test also group the observations into  $k$  separate groups using the ordinal scores of the estimated values. According to Lipsitz, Fitzmaurice, and Molenberghs (1996), the number of groups should be such that  $6 \leq k < n/5r$ , with  $r$  the number of response category. An indicator variable is used to denote the observations belonging to each group, producing additional pseudo variables with which the original model is

updated. Supposing the original model fits correctly, then the coefficients of the pseudo variables all equal zero. The likelihood ratio statistic calculated from the log likelihoods of the original and the refitted models is subsequently compared with the chi-squared distribution with  $k - 1$  degrees of freedom.

The Lipsitz test compliments the ordinal Hosmer-Lemeshow and the Pulkstenis-Robinson tests. Fagerland and Hosmer (2013, 2016, 2017) recommend comparing the three test.

## Value

LR	the realized value of the likelihood ratio statistic.
df	the degrees of freedom.
p.value	the p-value of the test.
newVar	a numeric vector of the newly generated variable.

## References

- Fagerland, M. W. and Hosmer, D. W. (2013). A goodness-of-fit test for the proportional odds regression model. *Statistics in Medicine*, 32, 2235-2249.
- Fagerland, M. W. and Hosmer, D. W. (2016). Tests for goodness of fit in ordinal logistic regression models. *Journal of Statistical Computation and Simulation*, 86, 3398-3418.
- Fagerland, M. W. and Hosmer, D. W. (2017). How to test for goodness of fit in ordinal logistic regression models. *Stata Journal*, 17, 668-686.

## See Also

[hosmerlem](#), [pulkroben](#), [brant.test](#), [LR.test](#)

## Examples

```
require(VGAM)

set.seed(1)
n <- 200
y <- ordered(rbinom(n, 2, 0.5))
x1 <- factor(rbinom(n, 1, 0.7))
x2 <- runif(n)

## proportional odds model
vg <- vglm(y ~ x1 + x2, model = TRUE,
           family = cumulative(link = "logitlink", parallel = TRUE))
lipsitz(vg, group=6)

## adjacent category model
ac <- update(vg, family = acat(parallel = TRUE))
lipsitz(ac)

## continuation ratio model
cr <- update(vg, family = cratio(parallel = TRUE))
lipsitz(cr)
```

---

LR.test

*Likelihood Ratio Test of the Proportional Odds Assumption*


---

**Description**

Provides the means of testing the parallel regression assumption in the ordinal regression models. Also available is the `brant.test()`.

**Usage**

```
LR.test(model, call=FALSE)
```

**Arguments**

<code>model</code>	a single model object to be tested.
<code>call</code>	a logical vector to indicate if the model call should be printed.

**Details**

The parallel regression assumption for the ordinal regression model can be tested With this function. It currently supports objects of class `serp()` and `vglm()`.

**Value**

<code>model</code>	the call of the model tested
<code>df</code>	the degrees of freedom
<code>rdf</code>	residual degrees of freedom
<code>rdev</code>	residual deviance
<code>LRT</code>	likelihood ratio test statistic
<code>prob</code>	the p-values of test
<code>call</code>	a logical vector

**See Also**

[brant.test](#), [hosmerlem](#), [lipsitz](#), [pulkroben](#)

**Examples**

```
require(serp)

sp <- serp(ordered(RET) ~ DIAB + GH + BP, link="logit",
           slope="parallel", reverse=TRUE, data = retinopathy)
LR.test(sp, call = TRUE)
```

---

print.brant	<i>Print method for an object of class brant</i>
-------------	--

---

**Description**

Prints out a summary table of brant-test for an object of class brant.

**Usage**

```
## S3 method for class 'brant'  
print(x, ...)
```

**Arguments**

x	An object of class brant.
...	additional arguments.

**Value**

No return value

**See Also**

[brant.test](#)

---

print.errorR	<i>Print method for an object of class errorR</i>
--------------	---

---

**Description**

Prints out a vector of name and calculated error value of fitted model.

**Usage**

```
## S3 method for class 'errorR'  
print(x, ...)
```

**Arguments**

x	An object of class errorR.
...	additional arguments.

**Value**

No return value

**See Also**

[erroR](#)

---

<code>print.hosmerlem</code>	<i>Print method for an object of class hosmerlem</i>
------------------------------	--

---

**Description**

Prints out a summary table of the goodness-of-fit test for binary, multinomial and ordinal models. class LRI.

**Usage**

```
## S3 method for class 'hosmerlem'
print(x, ...)
```

**Arguments**

- `x` An object of class hosmerlem.
- `...` additional arguments.

**Value**

No return value

**See Also**

[hosmerlem](#)

---

<code>print.lipsitz</code>	<i>Print method for an object of class lipsitz</i>
----------------------------	--

---

**Description**

Prints out a summary table of the goodness-of-fit test for an object of class lipsitz.

**Usage**

```
## S3 method for class 'lipsitz'
print(x, ...)
```

**Arguments**

- `x` An object of class lipsitz.
- `...` additional arguments.

**Value**

No return value

**See Also**

[lipsitz](#)

---

print.LRT	<i>Print method for an object of class LRT</i>
-----------	--

---

**Description**

Prints out a summary table of the likelihood ratio test for an object of class LRI.

**Usage**

```
## S3 method for class 'LRT'  
print(x, ...)
```

**Arguments**

x	An object of class LRI.
...	additional arguments.

**Value**

No return value

**See Also**

[LR.test](#)

---

print.pulkroben	<i>Print method for an object of class pulkroben</i>
-----------------	--

---

**Description**

Prints out a summary table of the goodness-of-fit test for an object of class pulkroben.

**Usage**

```
## S3 method for class 'pulkroben'  
print(x, ...)
```

**Arguments**

`x`                    An object of class `pulkroben`.  
`...`                additional arguments.

**Value**

No return value

**See Also**

[pulkroben](#)

---

<code>print.Rsquared</code>	<i>Print method for an object of class <code>r-squared</code></i>
-----------------------------	---

---

**Description**

Prints out a vector of name and value of an R2 of a fitted model.

**Usage**

```
## S3 method for class 'Rsquared'  
print(x, ...)
```

**Arguments**

`x`                    An object of class `rsquared`.  
`...`                additional arguments.

**Value**

No return value

**See Also**

[Rsquared](#)

pulkroben

*Pulkstenis-Robinson Test for Categorical Response Models***Description**

This provides a post estimation goodness-of-fit test for the ordinal response models. Supported models include the proportional odds, adjacent-category, and constrained continuation-ratio models.

**Usage**

```
pulkroben(model, test = c("chisq", "deviance"), tables = FALSE)
```

**Arguments**

model	a model object or data.frame of observed and estimated values. The following class of objects can be directly passed to the pulkroben function: vglm(), serp(), polr(), and clm(). Other class of objects require providing a data.frame of observed and predicted values.
test	chooses between the chi-squared and the deviance test statistic.
tables	Default to FALSE. When TRUE, both the observed and the expected frequency tables are printed alongside test results.

**Details**

The Pulkstenis-Robinson test groups the observations using the covariate patterns obtained from the categorical covariates. Each covariate pattern is subsequently split in two based on the median ordinal scores. The test statistic (chi-sq or deviance) is obtained using the tabulated observed and estimated frequencies. Assuming  $c$  is the number of covariate patterns,  $r$  the number of response categories and  $k$  the number of categorical variables in the model, the test statistic approximates the chi-sq distribution with  $(2c - 1)(r - 1) - k - 1$  degrees of freedom (Pulkstenis and Robinson (2004)). As recommended in Fagerland and Hosmer (2016, 2017), this test should be compared with the Hosmer-Lemeshow and the Lipsitz tests.

**Value**

stat	realized value of the chi-square or deviance statistic.
df	the degrees of freedom.
p.value	the p-value of the test statistic.
observed	a table of the observed frequencies.
expected	a table of the estimated frequencies.
rho	percentage of estimated frequencies greater than one.
test	a character vector of the type of test statistic used.
tables	a TRUE or FALSE logical vector.

## References

- Pulkstenis, E. and Robinson, T. J. (2004). Goodness-of-fit tests for ordinal response regression models. *Statistics in Medicine* 23, 999-1014.
- Fagerland, M. W. and Hosmer, D. W. (2016). Tests for goodness of fit in ordinal logistic regression models. *Journal of Statistical Computation and Simulation*, 86, 3398-3418.
- Fagerland, M. W. and Hosmer, D. W. (2017). How to test for goodness of fit in ordinal logistic regression models. *Stata Journal*, 17, 668-686.

## See Also

[hosmerlem](#), [lipsitz](#), [brant.test](#), [LR.test](#)

## Examples

```
require(VGAM)

set.seed(1)
n <- 200
y <- ordered(rbinom(n, 2, 0.5))
x1 <- factor(rbinom(n, 1, 0.7))
x2 <- runif(n)

## proportional odds model
vg <- vglm(y ~ x1 + x2, model = TRUE,
           family = cumulative(link = "logitlink", parallel = TRUE))
pulkroben(vg, tables = TRUE)

## adjacent category model
ac <- update(vg, family = acat(parallel = TRUE))
pulkroben(ac, tables = TRUE)

## continuation ratio model
cr <- update(vg, family = cratio(parallel = TRUE))
pulkroben(cr, tables = TRUE)
```

---

retinopathy

*Retinopathy*

---

## Description

The retinopathy data contains information on persons with retinopathy.

## Format

A data frame with 613 observations on the following 5 variables: - RET: RET=1: no retinopathy, RET=2 nonproliferative retinopathy, RET=3 advanced retinopathy or blind - SM: SM=1: smoker, SM=0: non-smoker - DIAB: diabetes duration in years - GH: glycosylated hemoglobin measured in percent - BP: diastolic blood pressure in mmHg

## References

Bender and Grouven (1998), Using binary logistic regression models for ordinal data with non-proportional odds, *J. Clin. Epidemiol.*, 51, 809-816.

## See Also

[vaso](#), [Fishing](#)

---

Rsquared	<i>R-squared for Categorical Response Models</i>
----------	--

---

## Description

computes the summary measures of predictive strength (i.e., pseudo-R2s) of several categorical outcome models.

## Usage

```
Rsquared(model, measure)
```

## Arguments

model	single model object for which R2 is determined.
measure	selects any of the different measures available.

## Details

Rsquared provides different R2 indices for both binary and multi-categorical response models. Supported classes include: glm, vglm, clm, polr, multinom, mlogit, serp. In other words, mainly models with binary or multi-categorical outcomes are supported. The non-likelihood based measures, including the Mckelvey, Tjur and Efron R2s are only available for binary models, while the rest of the measures (likelihood-based) are all available for both binary and multi-categorical models. The Ugba & Gertheiss's R2 in particular, computes the recently proposed modification of the popular Mcfadden's R2. The likelihood ratio index in the said R2 is penalized using either a square-root or logarithmic stabilizing function of the response category. The two approaches yield practically the same result.

## Value

measure	the name of the R-squared calculated.
R2	realized value of the computed R2.
adj	adjusted R2, only available when McFadden's R2 is computed.
sqrt.R2	Modified R2 with a square root penalty, only available when the Ugba & Gertheiss's R2 is computed.
log.R2	Modified R2 with a logarithmic penalty, only available when the Ugba & Gertheiss's R2 is computed.

## References

- Long, J.S. (1997). *Regression Models for Categorical and Limited Dependent Variables*. California: Sage Publications.
- Ugba, E. R. and Gertheiss, J. (2018). An Augmented Likelihood Ratio Index for Categorical Response Models. In *Proceedings of 33rd International Workshop on Statistical Modelling*, Bristol, 293-298.

## See Also

[erroR](#)

## Examples

```
require(serp)

pom <- serp(ordered(RET) ~ DIAB + GH + BP, link="logit",
           slope = "parallel", reverse = TRUE, data = retinopathy)
Rsquared(pom, measure = "mcfadden")
Rsquared(pom, measure = "ugba")
```

---

vaso

*Vasoconstriction and Breathing*

---

## Description

A binary data on the neural constriction of vasculature. Three test persons inhaled a certain amount of air with different rates. In some cases a vasoconstriction occurred at their skin. The objective of the study was to indicate a correlation between breathing and vasoconstriction. The test persons repeated the test 9, 8, 22 times, making a total of 39 observations.

## Format

A data frame with 39 observations on the following 3 variables: - vol: amount of air - rate: rate of breathing - vaso: condition of vasculature: no vasoconstriction = 1, vasoconstriction = 2

## Source

Data Archive Department of Statistics, LMU Munich.

## References

- Finney, D. J. (1971) *Probit Analysis*. 3rd edition. Cambridge University Press.
- Pregibon, D. (1982) Resistant fits for some commonly used logistic models. *Appl. Stat.* **29**, 15–24.
- Hastie, T. J. and Tibshirani, R. J. (1990) *Generalized Additive Models*. Chapman and Hall.

## See Also

[retinopathy](#), [Fishing](#)

# Index

## \* dataset

- Fishing, [5](#)
- retinopathy, [16](#)
- vaso, [18](#)

`brant.test`, [2](#), [7](#), [9–11](#), [16](#)

`erroR`, [4](#), [12](#), [18](#)

Fishing, [5](#), [17](#), [18](#)

`hosmerlem`, [3](#), [6](#), [8–10](#), [12](#), [16](#)

`lipsitz`, [3](#), [7](#), [8](#), [10](#), [13](#), [16](#)

`LR.test`, [3](#), [7](#), [9](#), [10](#), [13](#), [16](#)

`print.brant`, [11](#)

`print.erroR`, [11](#)

`print.hosmerlem`, [12](#)

`print.lipsitz`, [12](#)

`print.LRT`, [13](#)

`print.pulkroben`, [13](#)

`print.Rsquared`, [14](#)

`pulkroben`, [3](#), [7](#), [9](#), [10](#), [14](#), [15](#)

retinopathy, [5](#), [16](#), [18](#)

`Rsquared`, [4](#), [14](#), [17](#)

vaso, [5](#), [17](#), [18](#)