

# Package ‘MPDiR’

November 10, 2020

**Type** Package

**Title** Data Sets and Scripts for Modeling Psychophysical Data in R

**Version** 0.1-19

**Date** 2020-11-09

**Author** Kenneth Knoblauch and Laurence T. Maloney

**Maintainer** Ken Knoblauch <ken.knoblauch@inserm.fr>

**Depends** R (>= 3.0.0), stats, graphics

**Suggests** lattice

**LazyLoad** yes

**LazyData** yes

**Description** Data sets and scripts for Modeling Psychophysical Data in R (Springer).

**License** GPL-2

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2020-11-10 11:20:07 UTC

## R topics documented:

MPDiR-package . . . . .	2
binom.diagnostics . . . . .	2
Chromatic . . . . .	4
ConfRates . . . . .	5
Context . . . . .	6
CorticalCells . . . . .	7
Faces . . . . .	8
Faces2 . . . . .	9
Gabor . . . . .	9
Grue . . . . .	10
HSP . . . . .	11
lpois . . . . .	12
ModelFest.df . . . . .	13

Motion . . . . .	14
panel.psyfun . . . . .	15
polmer . . . . .	16
psyfun.boot . . . . .	17
StairCase . . . . .	18
thresh.est . . . . .	19
Vernier . . . . .	20

<b>Index</b>	<b>22</b>
--------------	-----------

---

MPDiR-package	<i>Modeling Psychophysical Data in R</i>
---------------	--

---

### Description

The package contains data sets, functions and scripts for the book Modeling Psychophysical Data in R

### Details

Package: MPDiR  
 Type: Package  
 Version: 1.0  
 Date: 2007-08-31  
 License: What license is it under?

### Author(s)

Kenneth Knoblauch and Laurence T. Maloney  
 Maintainer: Ken Knoblauch <ken.knoblauch@inserm.fr>

### References

K. Knoblauch and L. T. Maloney, Modeling Psychophysical Data in R, Springer, (2012) ISBN 978-1-4614-4475-6.

---

binom.diagnostics	<i>Diagnostics for Binary GLM</i>
-------------------	-----------------------------------

---

### Description

Two techniques for evaluating the adequacy of the binary glm model used in mlds, based on code in Wood (2006).

**Usage**

```
binom.diagnostics(obj, nsim = 200, type = "deviance", no.warn = TRUE)

## S3 method for class 'binom.diag'
plot(x, alpha = 0.025, breaks = "Sturges", ...)
```

**Arguments**

obj	Object of class 'glm' from a fit to data with a binary response.
nsim	number of bootstrap simulations to run.
type	character indicating type of residuals (defaults to "deviance") to use in the simulations.
no.warn	logical indicating whether to suppress warnings from glm. Defaults to TRUE.
x	object of class 'binom.diag', typically obtained from running <code>binom.diagnostics</code>
alpha	confidence level for envelope of the distribution of residuals
breaks	character indicating algorithm for choosing the breaks in the histogram of runs of residuals. Defaults to "Sturges". See <a href="#">hist</a> .
...	additional parameters specifications for the empirical cdf plot

**Details**

Wood (2006) describes two diagnostics of the adequacy of a binary glm model based on analyses of residuals (see, p. 115, Exercise 2 and his solution on pp 346-347). The first one compares the empirical cdf of the deviance residuals to a bootstrapped confidence envelope of the curve. The second examines the number of runs in the sorted residuals with those expected on the basis of independence in the residuals, again using a resampling based on the models fitted values. The plot method generates two graphs, the first being the empirical cdf and the envelope. The second is a histogram of the number of runs from the bootstrap procedure with the observed number indicated by a vertical line. Currently, this only works if the 'glm' method is used to perform the fit and *not* the 'optim' method

**Value**

`binom.diagnostics` returns a list of class 'binom.diag' with components

NumRuns	integer vector giving the number of runs obtained for each simulation
resid	numeric matrix giving the sorted deviance residuals in each column from each simulation
Obs.resid	numeric vector of the sorted observed deviance residuals
ObsRuns	integer giving the observed number of runs in the sorted deviance residuals
p	numeric giving the proportion of runs in the simulation less than the observed value.

**Author(s)**

Ken Knoblauch

## References

Wood, SN *Generalized Additive Models: An Introduction with R*, Chapman & Hall/CRC, 2006.

## Examples

```
## Not run:
# generate psychometric function and binary data
levs <- 10^seq(-2, 0, len = 5)
psyF <- pnorm(levs, mean = 0.2, sd = 0.2)
Ntrials <- 100
Resp <- rbinom(Ntrials * length(psyF), 1, psyF)
d.df <- data.frame(resp = Resp, levs = levs)
fit.glm <- glm(resp ~ levs, binomial, d.df)
fit.diag <- binom.diagnostics(fit.glm)
plot(fit.diag)

## End(Not run)
```

---

Chromatic

*Chromatic Thresholds across the Life Span*

---

## Description

Thresholds for equiluminant chromatic differences as a function of age along 3 axes in the CIE xy chromaticity diagram. Data were obtained using a spatial, two-alternative forced-choice paradigm. Responses were obtained from preverbal observers using forced-choice preferential looking.

## Usage

```
data(Chromatic)
```

## Format

A data frame with 511 observations on the following 4 variables.

**Log2Age** a numeric vector giving the log base 2 age of each observer

**Age** a numeric vector giving the age of each observer in years

**Thresh** a numeric vector giving the threshold modulation in distance along the axis from the 'white' point in CIE xy space along the axis being tested

**Axis** a factor with levels Deutan Protan Tritan giving the nominal axis in CIE xy space along which the threshold was measured. The terms refer to the confusion lines through the 'white' for each of three types of congenitally dichromatic observer. In the CIE xy-plane, these correspond to projections of axes that isolate modulations in each class of human cone photoreceptor.

**Details**

Equiluminant modulations along the ‘Protan’ and ‘Deutan’ axes projected in the CIE xy-plane are not cone-isolating. However, in the experiment, the chromatic modulations were presented mixed with dynamic, random luminance noise that had been shown to mask luminance components of the stimulus, thus favoring detection by chromatic pathways.

**References**

Knoblauch, K, Vital-Durand, F, Barbur, JL (2001). Variation of chromatic sensitivity across the life span. *Vision Research.*, 41, 1:23–36.

**Examples**

```
data(Chromatic)
```

---

ConfRates

*Confidence Ratings for Detection of Luminance Patch*

---

**Description**

Data from a detection task in which the observer gave ratings (1–5) indicating his confidence that a signal was presented on that trial. The signal was a 3 deg square patch of luminance, fixated centrally, embedded in a 10 deg field of random luminance pixels. The signal was 64 x 64 pixels. During the experiment, 6 luminance contrast levels were randomly presented and at an equal rate, randomly interspersed catch-trials in which the signal was absent.

**Usage**

```
data(ConfRates)
```

**Format**

A data frame with 196 observations on the following 3 variables.

**Contrast** a numeric vector indicating the luminance contrast of the signal, 0 is a catch trial.

**Rating** a numeric vector indicating the confidence rating of the observer as to the presence of the signal

**Stim** a logical vector indicating whether the signal was present or not on the trial

**Source**

Data kindly provided by Frédéric Devinck.

**References**

Devinck, F. (2003) *Les traitements visuels chez l'homme : stratégies de classification de la forme*. Université Lyon 2, Doctoral Thesis.

**Examples**

```
data(ConfRates)
```

---

Context	<i>Psychometric Functions for the Effects of Context on Contrast Detection</i>
---------	--

---

**Description**

Yes-No psychometric functions in which observers reported whether a vertical bar (the target) was present on each trial in the presence of contextual bars.

**Usage**

```
data(Context)
```

**Format**

A data frame with 120 observations on the following 5 variables.

Obs a factor with levels A B C D E F identifying each observer.

ContCntr a numeric vector indicating the contrast of the contextual elements.

TargCntr a numeric vector indicating the contrast of the target.

NumYes an integer vector indicating the number trials on which the observer responded as having seen the target.

NumNo an integer vector indicating the number of trials on which the observer responded as not having seen the target.

**Details**

The number of trials on which the target was detected is reported as a function of its contrast. The target bar was presented in the context of 6 aligned vertical bars (3 above and 3 below) and the 4 conditions correspond to 4 levels of contrast of the contextual bars, including a 0 contrast condition, i.e., no contextual bars.

**Source**

Data kindly provided by Li Zhaoping

**References**

Zhaoping L. & Jingling L. (2008) Filling-in and suppression of visual perception from context—a Bayesian account of perceptual biases by contextual influences, *PLoS Computational Biology* 4(2): e14 doi:10.1371/journal.pcbi.0040014

**Examples**

```
data(Context)
Context <- within(Context, Pc <- NumYes/(NumYes + NumNo))
lattice::xyplot(Pc ~ TargCntr | ContCntr, data = Context,
groups = Obs,type = c("l", "p"),
auto.key = list(space = "right"))
```

CorticalCells

*Contrast Response Functions from Visual Cortical Cells***Description**

The responses in impulses/second of 6 cells recorded in the macaque cortex in visual areas V1 and V2 to a series of contrasts of drifting sinusoidal gratings of optimal location, orientation, size and spatial frequency.

**Usage**

```
data(CorticalCells)
```

**Format**

A data frame with 42 observations on the following 7 variables.

**Contrast** numeric vector of the Michelson contrast of the grating

**Response** numeric vector of the mean response of the cell in impulse/second (action potentials)

**SEM** numeric vector of the standard error of the mean of the Response

**N** numeric vector of the number of responses recorded

**Type** factor with levels CX SM corresponding to the cell type classification, complex or simple

**Area** a factor with levels V1 V2 corresponding to the cortical area in which the cell was recorded

**Cell** a factor with levels a, ..., f as an identifier of the cell. The numbers, also, correspond to the panels of Figure 2 from the reference (see below) in which the data were published.

**Details**

Extracellular recordings of cells from macaque visual cortical areas V1 and V2. Simple and complex cells were differentiated by whether or not the response rate,  $f_0$ , was greater than the amplitude of modulation at the drift rate,  $f_1$ . If  $f_1 > f_0$ , the cell was classified as a simple cell and the value of  $f_1$  is reported; otherwise the cell is classified as a complex cell and the value of  $f_0$  is reported. In both cases, the units are in impulses/second.

**Source**

Data collected in Peter Lennie's lab, analysis by Jon Peirce (see reference below)

**References**

Peirce, J. W. (2007). The potential importance of saturating and supersaturating contrast response functions in visual cortex. *Journal of Vision*, 7(6):13, 1–10, <https://jov.arvojournals.org/7/6/13/>.

**Examples**

```
data(CorticalCells)
lattice::xyplot(Response ~ Contrast | Cell, CorticalCells,
type = "b")
```

---

Faces

*Perceived kinship in image pairs of children*

---

**Description**

Observers rated relatedness of pairs of images of children that were either siblings or not.

**Usage**

```
data(Faces)
```

**Format**

A data frame with 960 observations on the following 6 variables.

`SimRating` a numeric vector containing integer values from 0–10 indicating the observers confidence rating that the pair of images were siblings or not.

`sibs` a factor with levels 0 1, indicating whether the pair of images was, in fact, not related or related as siblings, respectively.

`agediff` a numeric vector indicating the age difference between the face pairs in months.

`gendiff` a factor with levels diff same indicating whether the images were of the same of different gender.

`Obs` a factor with levels S1–S32 coding the identities of individual observers.

`Image` a factor with levels Im1–Im30 coding the image pairs of the stimulus set.

**Details**

Observers used an 11–point scale (0–10) to rate their confidence in whether the image pairs were of siblings.

**References**

Maloney, L. T., and Dal Martello, M. F. (2006). Kin recognition and the perceived facial similarity of children. *Journal of Vision*, 6(10):4, 1047–1056, <https://jov.arvojournals.org/6/10/4/>.

**Examples**

```
data(Faces)
```



---

Faces2

*Perceived kinship in image pairs of children*

---

**Description**

Observers classified relatedness of pairs of images of children that were either siblings or not.

**Usage**

```
data(Faces2)
```

**Format**

A data frame with 960 observations on the following 3 variables.

Resp a factor with levels 1 or 0 indicating the observers classifications that the pair of images was of siblings or not, respectively.

Stim a factor with levels A P, indicating whether the pair of images was, in fact, not related or related as siblings, respectively.

Obs a factor with levels S1–S32 coding the identities of individual observers.

Image a factor with levels Im1 to Im30 coding the image pair presented on a given trial.

**Details**

Observers judged whether pairs of images of infants were siblings or not in a Yes/No fashion.

**References**

Maloney, L. T., and Dal Martello, M. F. (2006). Kin recognition and the perceived facial similarity of children. *Journal of Vision*, 6(10):4, 1047–1056, <https://jov.arvojournals.org//6/10/4/>.

**Examples**

```
data(Faces)
```

---

Gabor

*Detection of a 2 cycle/window Gabor temporal modulation*

---

**Description**

The data record the observations of one observer performing a Yes/No detection task. The signal was a temporal luminance modulation of a Gaussian spatial profile ( $\sigma = 2.5$  deg) on a CRT screen. The temporal modulation followed a Gabor function with an envelope of  $\sigma = 160$  ms and duration of  $4\sigma$ . The carrier was in sine-phase and of frequency 2 cycles/window width. On each trial random uniform luminance modulations were presented, and on half of the trials, the signal was added in. The data represent 16 sessions of 224 trials each.

**Usage**

```
data(Gabor)
```

**Format**

A data frame with 114688 observations on the following 4 variables.

`resp` a factor with levels H F A M CR. These correspond to the response classifications of the observer and correspond to ‘Hit’, ‘False Alarm’, ‘Miss’ and ‘Correct Rejection’.

`time` a factor with levels 1, 2, ..., 32 indicating the time points of the stimulus coded as factor levels.

`Time` a numeric vector indicating time points of the stimulus in seconds.

`N` a numeric vector of random uniform noise, the random part of the stimulus. On half of the trials the Gabor signal was added to the noise.

**Details**

More details on the stimulus conditions are given in the reference indicated below.

**Source**

J. P. Thomas, and K. Knoblauch (2005) Frequency and phase contributions to the detection of temporal luminance modulation. *Journal of the Optical Society of America A*, 22, 2257–2261.

**Examples**

```
data(Gabor)
str(Gabor)
```

---

Grue

*Distribution of Grue Languages with UV-B Exposure*

---

**Description**

Data reported by Lindsey and Brown (2002) based on a corpus of 203 languages (only 201 represented in this data set) of the numbers of languages with separate words for ‘blue’ and ‘green’, a combined word for the two, so-called grue-languages, or the word ‘dark’ for the two, as a function of the average annual exposure to UV-B light.

**Usage**

```
data(Grue)
```

**Format**

A data frame with 4 observations on the following 4 variables.

UV\_B a numeric vector indicating average annual UV-B exposure in kiloJoules/m<sup>2</sup>

blue-green a numeric vector giving the numbers of languages with a separate terms for 'blue' and 'green'

grue a numeric vector, the numbers of languages with a single term for 'green/blue'

dark a numeric vector, the numbers of languages with a term 'dark' for green or blue

**Source**

Data kindly provided by Angela M. Brown.

**References**

D. T. Lindsey and A. M. Brown (2002) Color naming and the phototoxic effects of sunlight on the eye. *Psychological Science*, **13**, 506-512.

**Examples**

```
data(Grue)
barplot(t(as.matrix(Grue[, -1])), names.arg = round(Grue$UV_B, 1),
beside = TRUE, legend.text = c("blue-green", "grue", "dark"),
col = c("white", "grey", "black"),
ylim = c(0, 40), cex.lab = 1.5,
xlab = expression(paste("Average UV-B (kJ/", m^2, ")")),
ylab = "Numbers of Languages"
)
```

---

HSP

*Hecht, Schlaer and Pirenne (1942) psychometric functions*


---

**Description**

The data frame contains the per cent of correct responses as a function of average number of quanta per flash of light at the cornea and the frequency with which it was seen for 5 psychometric functions reported in Table V of Hecht et al. (1942)

**Usage**

```
data(HSP)
```

**Format**

A data frame with 30 observations on the following 5 variables.

Q a numeric vector corresponding to the column labelled as ‘No. of quanta’ in the original article.

p a numeric vector corresponding to the column labelled as ‘Frequency’ with a subtitle ‘per cent’ in the original article

N a numeric vector indicating the number of presentations of the flash

Obs a factor with levels MHP SH SS indicating the initials of the observers

Run a factor with levels R1 R2 indicating the session or run of the experiment

**Details**

Hecht et al. (1942) reported the data from 5 psychometric functions for 3 observers (the authors), two of which are replications on the same observer. Each flash intensity was presented 50 times except for the data for observer SH, for which the presentations numbers are 35 and 40, respectively, for the two runs.

**Source**

S. Hecht, S. Schlaer and M.H. Pirenne (1942) Energy, quanta and vision. *Journal of General Physiology*, 25, 819–840.

**Examples**

```
data(HSP)
lattice::xyplot(p/100 ~ Q | Obs * Run, HSP,
  type = c("p", "l"),
  scales = list(x = list(log = TRUE)),
  as.table = TRUE)
```

---

lpois

*Likelihoods for Poisson and Gaussian psychometric functions*


---

**Description**

These functions define likelihoods for binomial models with Gaussian or Poisson psychometric functions.

**Usage**

```
lpois(p, d)
lpois1(q, p, d)
lnorm(p, d)
```

**Arguments**

p	numeric vector of parameters (in lpois and lnorm) or fixed Poisson parameter (lpois2).
d	data frame with number of correct (nyes) and incorrect (nno) responses and intensity levels (Q)
q	numeric vector of length 1 giving scale parameter (lpois2)

**Details**

These functions calculate the log likelihood for binomial models with Gaussian or Poisson psychometric functions and can be used with [optim](#) to find the best parameters.

**Value**

Returns a numeric value giving minus the log likelihood for the given model.

**Author(s)**

Kenneth Knoblauch

**See Also**

See [optim](#)

**Examples**

```
data(HSP)
SHR2 <- subset(HSP, Obs == "SH" & Run == "R2")
SHR2 <- within(SHR2, {
  nyes <- N * p/100
  nno <- N - nyes
})
optim(par = c(5, 0.35), lnorm, d = SHR2)
```

**Description**

The data set contains the log contrast sensitivity measurements for the 43 stimuli used in the ModelFest consortium with 4 repeats for each of the 16 observers.

**Usage**

```
data(ModelFest.df)
```

**Format**

A data frame with 2752 observations on the following 3 variables.

LContSens a numeric vector indicating the measured log contrast sensitivity, inverse of the log contrast threshold

Obs a factor with 16 levels, indicating the initials of the observers

Stim a factor with 43 levels coding each of the stimuli standardized for this study

**Details**

The ModelFest consortium was composed of a group of investigators from 10 laboratories who undertook to obtain measures of contrast sensitivity on a set of standardized achromatic stimuli, varying in spatial configuration, using a standardized set of procedures.

**Source**

<https://jov.arvojournals.org//5/9/6/modelfestbaselinedata.csv>

**References**

Carney, T., Tyler, C.W., Watson, A.B., Makous, W., Beutte, B., Chen, C.C., Norcia, A.M., Klein, S.A.(2000) Modelfest: year one results and plans for future years. In: B.E. Rogowitz, T.N. Pappas (eds.) *Proceedings of SPIE: Human Vision and Electronic Imaging V*, vol. 3959, pp. 140–151. SPIE

Watson, A.B., Ahumada, A.J. (2005) A standard model for foveal detection of spatial contrast. *Journal of Vision* 5, 717–740

Watson, A. (2006) The spatial standard observer: A human vision model for display inspection. *SID Symposium Digest of Technical Papers*, vol. 37, pp. 1312–1315, SID

**Examples**

```
data(ModelFest.df)
str(ModelFest.df)
plot(ModelFest.df)
```

---

Motion

*Threshold for First- and Second-order Movement with Age*

---

**Description**

These data were reported by Thibault et al (2007) (their Figure 4) and provide developmental data on the contrast modulation threshold for detection of a moving stimulus for two kinds of motion, called First-order (or luminance based) and Second-order (contrast based).

**Usage**

```
data(Motion)
```

**Format**

A data frame with 112 observations on the following 5 variables.

Subject a factor with levels S01,S02, ..., S70 coding the id of each observer

LnAge a numeric vector giving the natural log of age in months

Mtype a factor with levels F0 S0 coding the type of stimulus, first-order or second-order

Sex a factor with levels f m indicating the sex of the observer

LnThresh a numeric vector giving the natural log of the percent contrast modulation threshold

**Details**

The data were collected using preferential-looking techniques. Percent contrast varies between 0 and 100.

**Source**

Data kindly provided by F. Vital-Durand.

**References**

Thibault, D., Brosseau-Lachaine, O., Faubert, J. and Vital-Durand, F. (2007) Maturation of the sensitivity for luminance and contrast modulated patterns during development of normal and pathological human children. *Vision Research*, 47, 1561–1569

**Examples**

```
data(Motion)
```

---

```
panel.psyfun
```

*Panel Function for Adding Psychometric Function Fit to Each Panel*

---

**Description**

Provides a panel function for use with [xyplot](#) from the **lattice** package so that a psychometric function will be fitted and plotted in each panel of a lattice plot similar to [panel.lmline](#) for fitting a linear regression line.

**Usage**

```
panel.psyfun(x, y, n, lnk = "logit", ...)
```

**Arguments**

x	vector of covariate values
y	vector of responses, usually proportion of correct responses
n	vector of integers specifying number of trial for each proportion in y
lnk	link function to use in fitting glm with binomial family (default is "logit").
...	Other parameters passed from the formal arguments of the panel function.

**Value**

Nothing returned. Function just used for its side-effects of producing a graph within a panel function.

**Author(s)**

Kenneth Knoblauch

**See Also**

See also as [lattice](#)

**Examples**

```
lattice::xyplot(Pc ~ Phaseshift | WaveForm + TempFreq +
Direction, Vernier, layout = c(4, 2),
panel = function(x, y, n = 20, ...) {
lattice::panel.xyplot(x, y)
panel.psyfun(x, y, 20, lnk = "probit")
}
)
```

---

polmer

*Ordinal Regression with Mixed-effects*

---

**Description**

Regression models are fit responses that are ordered factors with (or without) random effects.

**Usage**

```
polmer( ...)
```

**Arguments**

... dummy arguments as the function is deprecated. See below for where to find code.

**Details**

This function is a wrapper that calls `glmer` from the **lme4** package if any random effect terms appear in the formula or `glm`, if not. The response term should be of class ‘integer’, as the function will coerce it to ‘ordered’. In the formula object, the random effect should be specified as the second level of random effect with the intercept removed. See the example below. This is a glitch for the moment. This function is now deprecated but the code can be found in the scripts directory in the file, ‘Ch9.R’.



**Value**

An object of class `mer` or `glm` depending on whether or not any random effect terms are included in the formula object.

**Author(s)**

Kenneth Knoblauch

**See Also**

[glm](#), [glmer](#), [polr](#)

**Examples**

```
# data(Faces)
# if(require(lme4.0, quietly = TRUE)){
# no random effects specified - calls glm
# Faces.glm <- polmer(SimRating ~ sibs, Faces)
# random effect of observer - call glmer
# GLITCH: must specify random effect as second level of factor (levels of sibs are 0/1)

# Faces.glmer <- polmer(SimRating ~ sibs + (sibs1 - 1 | Obs),
# Faces)
# }
```

---

psyfun.boot

*Bootstrapping Standard Errors of Psychometric Function Parameters*

---

**Description**

A function that will run a bootstrap on the estimated parameters of a psychometric function fit given a model object.

**Usage**

```
psyfun.boot(obj, N = 100)
```

**Arguments**

<code>obj</code>	object inheriting from class 'glm' from a fit of a psychometric function
<code>N</code>	integer indicating number of bootstrap replications.

**Details**

The function computes new binomial responses based on the fitted probabilities of the model object for each bootstrap replication. A psychometric function is then fit to each one and the fitted coefficients returned as a bootstrap replicate.

**Value**

Returns a matrix with one row for each coefficient of the model and one column for each bootstrap replication.

**Author(s)**

Kenneth Knoblauch

**References**

- Maloney, L. T. (1990) Confidence interval for the parameters of psychometric functions. *Perception & Psychophysics*, **47(2)**, 127–134.
- Foster, D.H., Bischof, W.F.(1997) Bootstrap estimates of the statistical accuracy of thresholds obtained from psychometric functions. *Spatial Vision*, **11(1)**, 135–139.
- Treutwein, B., Strasburger, H. (1999) Fitting the psychometric function. *Perception & Psychophysics*, **61(1)**, 87–106.

**Examples**

```
data(HSP)
SHR2 <- subset(HSP, Obs == "SH" & Run == "R2")
SHR2 <- within(SHR2, {
  nyes <- N * p/100
  nno <- N - nyes
})
SHR2.glm <- glm(cbind(nyes, nno) ~ log(Q), binomial, SHR2)
### For a real problem, set N to 10000 or so
SHR2.boot <- psyfun.boot(SHR2.glm, 10)
```

---

StairCase

*3-1 Randomly Interleaved Staircase Detection Experiment*

---

**Description**

Data from one observer performing a 2-alternative forced-choice detection experiment. A 3-1 staircase procedure was used with two randomly interleaved staircases.

**Usage**

```
data(StairCase)
```

**Format**

A data frame with 96 observations on the following 4 variables.

**Trial** a numeric vector indicating the trial number for the staircase

**Contrast** a numeric vector indicating the contrast of the test stimulus

Response a numeric vector taking the values of 1, when the correct interval was detected, and 0 when it was not

StairCase a factor with levels S1 S2 indicating to which staircase the trial belonged.

### Details

The data were obtained using a 3-down, 1-up staircase procedure. The observer was required to make 3 successive correct judgments before the stimulus contrast was lowered; one incorrect response resulted in an increase in contrast at the next trial. Initially, the contrast was reduced by a factor of 2 for each sequence of 3 correct responses. After a first incorrect response, successive increments and decrements were by a factor of 1.26. Each staircase was 48 trials long. In the actual experiment, the two staircases were run concurrently, with the contrast level chosen by switching from one staircase to the other at random.

### Examples

```
clr = c("red", "blue")
with(StairCase, plot(Contrast ~ Trial, log = "y",
  col = clr[unclass(StairCase)], pch = 16))
```

---

thresh.est

*Estimate Threshold Level from Model Object*

---

### Description

Estimates stimulus level at which probability of a response is some fixed value given a model object for a probit model.

### Usage

```
thresh.est(p, obj)
```

### Arguments

p numeric vector giving the performance level for the desired threshold  
 obj object inheriting from class 'glm' from a fit with a binomial family.

### Details

Calculates the stimulus level(s) for a probit model of a psychometric function for given levels of performance. For a more general version, see [dose.p](#) in pkgMASS.

### Value

Vector of numeric indicating the stimulus levels for the given performance levels.

### Author(s)

Kenneth Knoblauch

**See Also**

See also [dose.p](#)

**Examples**

```
data(HSP)
SHR1.glm <- glm(p/100 ~ log(Q), binomial(probit), HSP,
subset = Obs == "SH" & Run == "R1")
thresh.est(1:3/4, SHR1.glm)
```

---

Vernier

*Psychometric Functions for Vernier Judgement of Phase Difference  
between Drifting Gratings*

---

**Description**

A pair of adjacent, horizontal luminance gratings (sine- or square-wave spatial profiles, 0.8 cycles/degree and equal contrast = 0.30) drifting at 2 or 8 Hz. The gratings were separated by a 30-arcmin lateral gap. On each trial, the grating pair was drifting either upward or downward and the phase shift between the gratings was chosen at one of 8 values between -50 and 50 degrees. The observer was instructed to report which grating appeared shifted upward. The method of constant stimuli was used.

**Usage**

```
data(Vernier)
```

**Format**

A data frame with 64 observations on the following 8 variables.

**Phaseshift** a numeric vector giving the Phaseshift between the pair of gratings

**WaveForm** a factor with levels Sine Square giving the spatial luminance profile of the gratings

**TempFreq** a factor with levels 2 8 indicating the temporal frequency in Hertz of the moving gratings

**Pc** a numeric vector, indicating the proportion of trials on which a particular grating appeared to be shifted upward

**Direction** a factor with levels Downward Upward indicating the direction of motion of the grating pair.

**N** a numeric vector indicating the number of trials on which a condition was presented

**NumUpward** a numeric vector indicating the number of trials on which the observer responded that a particular grating was shifted upward

**NumDownward** a numeric vector indicating the number of trials on which the observer reported a particular grating as shifted downward

**Source**

Data kindly provided by H. Sun.

**References**

H. Sun, B. B. Lee and R. C. Baraas Systematic misestimation in a vernier task arising from contrast mismatch, *Visual Neuroscience*, 2008, 25, 365–370.

**Examples**

```
data(Vernier)
```

# Index

- \* **aplot**
  - panel.psyfun, 15
- \* **datasets**
  - Chromatic, 4
  - ConfRates, 5
  - Context, 6
  - CorticalCells, 7
  - Faces, 8
  - Faces2, 9
  - Gabor, 9
  - Grue, 10
  - HSP, 11
  - ModelFest.df, 13
  - Motion, 14
  - StairCase, 18
  - Vernier, 20
- \* **dplot**
  - panel.psyfun, 15
- \* **hplot**
  - binom.diagnostics, 2
  - panel.psyfun, 15
- \* **models**
  - binom.diagnostics, 2
  - lpois, 12
  - polmer, 16
  - psyfun.boot, 17
  - thresh.est, 19
- \* **nonparametric**
  - psyfun.boot, 17
- \* **package**
  - MPDiR-package, 2
- \* **univar**
  - thresh.est, 19
- binom.diagnostics, 2
- Chromatic, 4
- ConfRates, 5
- Context, 6
- CorticalCells, 7
- dose.p, 19, 20
- Faces, 8
- Faces2, 9
- Gabor, 9
- glm, 17
- glmer, 17
- Grue, 10
- hist, 3
- HSP, 11
- lattice, 16
- Inorm(lpois), 12
- lpois, 12
- lpois1(lpois), 12
- ModelFest.df, 13
- Motion, 14
- MPDiR (MPDiR-package), 2
- MPDiR-package, 2
- optim, 13
- panel.lmline, 15
- panel.psyfun, 15
- plot.binom.diag(binom.diagnostics), 2
- polmer, 16
- polr, 17
- psyfun.boot, 17
- StairCase, 18
- thresh.est, 19
- Vernier, 20
- xyplot, 15