Package ‘perccalc’

December 17, 2019

Title  Estimate Percentiles from an Ordered Categorical Variable
Version  1.0.5
Description  An implementation of two functions that estimate values for percentiles from an ordered categorical variable as described by Reardon (2011, isbn:978-0-87154-372-1). One function estimates percentile differences from two percentiles while the other returns the values for every percentile from 1 to 100.
Depends  R (>= 3.4.0)
License  MIT + file LICENSE
URL  https://cimentadaj.github.io/perccalc/,

https://github.com/cimentadaj/perccalc
language  en-US
Encoding  UTF-8
LazyData  true
RoxygenNote  7.0.1
Imports  stats, tibble, multcomp
Suggests  magrittr, spelling, dplyr, knitr, rmarkdown, testthat,
    ggplot2, MASS, carData, tidyr (>= 1.0.0), covr
VignetteBuilder  knitr
NeedsCompilation  no
Author  Jorge Cimentada [aut, cre] (<https://orcid.org/0000-0001-5594-1156>)
Maintainer  Jorge Cimentada <cimentadaj@gmail.com>
Repository  CRAN
Date/Publication  2019-12-17 20:10:02 UTC

R topics documented:

perc_diff 2
perc_dist 3
pisa_2006 4
pisa_2012 5
perc_diff

Description

Calculate percentile differences from an ordered categorical variable and a continuous variable.

Usage

perc_diff(df, categorical_var, continuous_var, weights = NULL, percentiles = c(90, 10))

perc_diff_df(df, categorical_var, continuous_var, weights = NULL, percentiles = c(90, 10))

Arguments

data_model: A data frame with at least the categorical and continuous variables from which to estimate the percentile differences

categorical_var: The bare unquoted name of the categorical variable. This variable SHOULD be an ordered factor. If not, will raise an error.

continuous_var: The bare unquoted name of the continuous variable from which to estimate the percentiles

weights: The bare unquoted name of the optional weight variable. If not specified, then estimation is done without weights

percentiles: A numeric vector of two numbers specifying which percentiles to subtract

Details

perc_diff drops missing observations silently for calculating the linear combination of coefficients.
perc_dist

Value

perc_diff returns a vector with the percentile difference and its associated standard error. perc_diff_df returns the same but as a data frame.

Examples

```r
set.seed(23131)
N <- 1000
K <- 20

toy_data <- data.frame(id = 1:N,
                        score = rnorm(N, sd = 2),
                        type = rep(paste0("inc", 1:20), each = N/K),
                        wt = 1)

# perc_diff(toy_data, type, score)
# type is not an ordered factor!

toy_data$type <- factor(toy_data$type, levels = unique(toy_data$type), ordered = TRUE)

perc_diff(toy_data, type, score, percentiles = c(90, 10))
perc_diff(toy_data, type, score, percentiles = c(50, 10))

perc_diff(toy_data, type, score, weights = wt, percentiles = c(30, 10))

# Results as data frame
perc_diff_df(toy_data, type, score, weights = wt, percentiles = c(30, 10))
```

calculate a distribution of percentiles from an ordered categorical variable and a continuous variable.

Description

Calculate a distribution of percentiles from an ordered categorical variable and a continuous variable.

Usage

```r
perc_dist(data_model, categorical_var, continuous_var, weights = NULL)
```

Arguments

data_model A data frame with at least the categorical and continuous variables from which to estimate the percentiles
categorical_var
The bare unquoted name of the categorical variable. This variable should be an ordered factor. If not, will raise an error.

continuous_var
The bare unquoted name of the continuous variable from which to estimate the percentiles

weights
The bare unquoted name of the optional weight variable. If not specified, then equal weights are assumed.

Details
perc_dist drops missing observations silently for calculating the linear combination of coefficients.

Value
A data frame with the scores and standard errors for each percentile

Examples

set.seed(23131)
N <- 1000
K <- 20

toy_data <- data.frame(id = 1:N,
                      score = rnorm(N, sd = 2),
                      type = rep(paste0("inc", 1:20), each = N/K),
                      wt = 1)

# perc_diff(toy_data, type, score)
# type is not an ordered factor!

toy_data$type <- factor(toy_data$type, levels = unique(toy_data$type), ordered = TRUE)

perc_dist(toy_data, type, score)

pisa_2006
Mathematics test scores of Spain, Germany and Estonia in the PISA 2006 test

Description
A dataset containing the test scores and other household information of students from Spain, Germany and Estonia from the PISA 2006 test.

Usage
pisa_2006
Format
A data frame with 25884 rows and 10 variables:

- **year** Year of the survey
- **CNT** Long country names
- **STIDSTD** Unique student id
- **father_edu** The father's highest achieved degree in the ISCED scale
- **household_income** The household's total income in categories
- **avg_math** The average math test score out of the 5 plausible values in Mathematics

Source

| pisa_2012 | Mathematics test scores of Spain, Germany and Estonia in the PISA 2012 test |

Description
A dataset containing the test scores and other household information of students from Spain, Germany and Estonia from the PISA 2012 test.

Usage
pisa_2012

Format
A data frame with 35093 rows and 10 variables:

- **year** Year of the survey
- **CNT** Long country names
- **STIDSTD** Unique student id
- **father_edu** The father's highest achieved degree in the ISCED scale
- **household_income** The household's total income in categories
- **avg_math** The average math test score out of the 5 plausible values in Mathematics

Source
A subset extracted from the PISA2012lite R package, https://github.com/pbiecek/PISA2012lite
Index

*Topic datasets
    pisa_2006, 4
    pisa_2012, 5

perc_diff, 2
perc_diff_df (perc_diff), 2
perc_dist, 3
pisa_2006, 4
pisa_2012, 5