Package 'DataVeracity'

October 24, 2017

Type Package

Title Veracity assessment of reported data

Version 1.2

Maintainer Carel F.W. Peeters `<cf.peeters@vumc.nl>`

Author Carel F.W. Peeters [cre, aut], Chris A.J. Klaassen [aut], Mark A. van de Wiel [aut]

Description
DataVeracity contains functions for the assessment of the veracity of reported data for one-way ANOVA-type designs.

Depends R (>= 2.15.1)

Imports ggplot2, stats, graphics, grDevices, msm, emdbook

ByteCompile true

KeepSource yes

License GPL (>= 2)

BuildManual yes

URL https://github.com/CFWP/DataVeracity

R topics documented:

DataVeracity-package   *Veracity assessment of reported data*

Description

DataVeracity contains functions for the assessment of the veracity of reported data for one-way ANOVA-type designs.

Details

This package contains functions for the evaluation of the reported linearity of the trend across experimental conditions in one-way ANOVA-type designs under the assumption that perfect linearity of the trend holds in the population. It allows for the evaluation of excessive linearity or lack of deviation from linearity in reported results. It thus contains the packaged version of the functions that supported the investigation – commissioned by the University of Amsterdam – into the scientific veracity of reported data in publications (co)authored by prof. dr Jens Foerster (Peeters, Klaassen, & van de Wiel, 2015). These functions support veracity evaluation on the basis of the reported summary statistics alone. Moreover, the veracity evaluation can be approached from various angles.

The first angle is the evidential value as developed by Klaassen (2015). The evidential value assesses, for individual (sub-)experiments, the hypothesis of a dependence structure in the underlying data (which is indicative of low data veracity) against the standard ANOVA model assumption of independence (which is indicative of high data veracity). The evidential value in a sense quantifies excessive closeness of the cell averages to perfect linearity given the reported cell variances, under the assumption that perfect linearity in the cell means holds in the population.

The second angle is nested F-testing paired with Fisher's method (Whistleblower, 2012). It can be used to assess if the variation in deviations from perfect linearity is statistically insufficient. Hence, it evaluates if the reported effects across experiments deviate, given the reported standard deviations and sample sizes, too little from linearity, even under the assumption of perfect linearity in the population.

The third angle is a simulation approach proposed by Nelson & Simonsohn (2014) and Simonsohn (2014). It evaluates how deviant the reported (averaged absolute) deviation from perfect linearity is by simulating the publication many times and assessing how many times we observe a result at least as extreme as the reported result.

Function overview:

- Functions for the evidential value
    - evi.val
    - evi.valMult
    - evi.valP
    - evi.valPi
    - evi.valTrial
- Functions for nested F-testing
    - deltaF.linear
    - Fisher.method
- Function for veracity assessment by simulation
    - lackNoise

- Functions for visualization
    - visual.trend
    - visual.trend.pooled

Author(s)

Carel F.W. Peeters, Chris A.J. Klaassen, Mark A. van de Wiel
Maintainer: Carel F.W. Peeters <cf.peeters@vumc.nl>

References

Klaassen, C.A.J. (2015). Evidential value in ANOVA-regression results in scientific integrity studies. arXiv:1405.4540v2 [stat.ME].

Nelson, L., & Simonsohn, U. (2014). [21] Fake-Data Colada.

Peeters, C.F.W., Klaassen, C.A.J., & van de Wiel, M.A. (2015). Evaluating the Scientific Veracity of Publications by dr. Jens Foerster. Available from the University of Amsterdam.

Simonsohn, U. (2014). R Code Behind DataColada[21]

Whistleblower Report. (2012). Suspicion of scientific misconduct by dr. Jens Foerster. Available from Retraction Watch.

---

deltaF.linear	*Compute F-test for nested (ANOVA) models*

Description

Function that computes the F-test for nested (ANOVA) models.

Usage

```
deltaF.linear(meanVec, sdVec, n.fl)
```

Arguments

meanVec	A numeric vector containing means for each factor level.
sdVec	A numeric vector containing standard deviations for each factor level.
n.fl	A scalar numeric or integer indicating cell size/nr. of replicates per factor level.

Details

The ANOVA F-model for one-way factorial designs with k levels of an experimental factor has $k-1$ regression parameters. A linear regression between the low and high levels of the experimental factor has only 1 regression parameter (reduced model). This function then compares the reduced linear model with the more complex model bearing $k-1$ regression parameters. A nested F-test (ΔF-test) is performed to assess if the more complex model significantly contributes to model fit. The null hypothesis postulates perfect linearity in underlying means. The test-statistic follows an F-distribution under the null hypothesis. The p-value is obtained in reference to this ditribution.

The following assumptions are made:

- An ANOVA-type setting with one-way factorial design;
- Random assignment of subjects to experimental conditions;
- A balanced design;
- Normally-distributed measurement errors;
- The input vectors are ordered according to linearity of effect.

The function is based on the approach stated in the Whistleblower report (2012). For additional details see Peeters, Klaassen, & van de Wiel (2015).

Value

The function returns an object of class `list`:

`$deltaF`	A numeric scalar representing the observed value of the ΔF statistic.
`$deltaF.p`	A numeric scalar representing the *p*-value associated with the observed value of the ΔF statistic

Author(s)

Carel F.W. Peeters <cf.peeters@vumc.nl>, Chris A.J. Klaassen, Mark A. van de Wiel

References

Peeters, C.F.W., Klaassen, C.A.J., & van de Wiel, M.A. (2015). Evaluating the Scientific Veracity of Publications by dr. Jens Foerster. Available from the University of Amsterdam.

Whistleblower Report. (2012). Suspicion of scientific misconduct by dr. Jens Foerster. Available from Retraction Watch

See Also

`Fisher.method`

Examples

```
## Means and standard deviations
Means <- c(2.08, 2.48, 3.33)
SDs   <- c(0.76, 0.36, 0.99)

## delta F testing
Test <- deltaF.linear(Means, SDs, n.fl = 72/6)

## Observed value delta F statistic
Test$deltaF

## p-value
Test$deltaF.p
```

evi.val	*Compute evidential value*

Description

Function that computes the evidential value in favor of the hypothesis of a dependence-structure in the underlying data (i.e., correlated measurement errors, indicating incorrect data collection) versus the hypothesis of independence (the standard ANOVA assumption).

Usage

```
evi.val(meanVec, sdVec, n.fl)
```

Arguments

meanVec	A `numeric` vector containing means for each factor level.
sdVec	A `numeric vector` containing standard deviations for each factor level.
n.fl	A scalar `numeric` or `integer` indicating cell size/nr. of replicates per factor level.

Details

The evidential value computed by this function can be found in Klaassen (2015). It is based on the basic premise that humans tend to underestimate variation due to randomness when fabricating data. Within the framework of the ANOVA model this is incorporated by allowing for dependence between the measurement errors of the respective factor-levels. The evidential value then assesses the hypothesis "of a dependence structure in the underlying data", which indicates low scientific veracity, "versus the hypothesis of independence, which is the ANOVA model assumption" (Klaassen, 2015). The following assumptions are made:

- An ANOVA-type setting with one-way factorial design and 3 factor-levels;
- Random assignment of subjects to experimental conditions;
- A balanced design;
- Normally-distributed measurement errors;
- The input vectors are ordered according to linearity of effect;
- The effect is perfectly linear in the population, $\mu_1 - 2\mu_2 + \mu_3 = 0$;
- Dependence between the measurement errors undermines (the assumption of) veracity.

The evidential value V is always greater than or equal to unity (1). Studies with data that adhere to the standards of scientific experimentation can be expected to sort an V close to unity. The larger the evidential value, the more the evidence favors the hypothesis of dependence, i.e., of fabrication or questionable research practices, versus the hypothesis of independence. The evidential value V may thus be used to assess individual constituent (sub)experiments within a publication.

For additional details, see Klaassen (2015) or the appendix to Peeters, Klaassen, & van de Wiel (2015).

Value

The function returns either a numeric scalar representing the evidential value or a list in which:

$EV.lower   is a numeric scalar representing the lower bound to the evidential value

$EV.upper   is a numeric scalar representing the upper bound to the evidential value

Author(s)

Carel F.W. Peeters <cf.peeters@vumc.nl>, Chris A.J. Klaassen, Mark A. van de Wiel

References

Klaassen, C.A.J. (2015). Evidential value in ANOVA-regression results in scientific integrity studies. arXiv:1405.4540v2 [stat.ME].

Peeters, C.F.W., Klaassen, C.A.J., & van de Wiel, M.A. (2015). Evaluating the Scientific Veracity of Publications by dr. Jens Foerster. Available from the University of Amsterdam.

See Also

evi.valMult, evi.valP, evi.valPi, evi.valTrial

Examples

```
## Means and standard deviations
Means <- c(2.08, 2.48, 3.33)
SDs   <- c(0.76, 0.36, 0.99)

## Evidential value
EV <- evi.val(Means, SDs, n.fl = 72/6)

## Result
EV
```

---

evi.valMult   *Compute overall evidential value*

Description

Function that computes the overall evidential value.

Usage

```
evi.valMult(EVvals)
```

Arguments

EVvals   A numeric vector containing multiple evidential values.

Details

When multiple (sub)experiments are available for independent samples, an (lower-bound to the) overall evidential value can be obtained by multiplication of individual (lower-bound) values for V. This convenience function simply computes the product of the evidential values listed in the `EVvals` argument.

Value

The function returns a `numeric` scalar representing the (lower bound to the) overall evidential value.

Author(s)

Carel F.W. Peeters <cf.peeters@vumc.nl>, Chris A.J. Klaassen, Mark A. van de Wiel

References

Klaassen, C.A.J. (2015). Evidential value in ANOVA-regression results in scientific integrity studies. arXiv:1405.4540v2 [stat.ME].

Peeters, C.F.W., Klaassen, C.A.J., & van de Wiel, M.A. (2015). Evaluating the Scientific Veracity of Publications by dr. Jens Foerster. Available from the University of Amsterdam.

See Also

`evi.val`, `evi.valP`, `evi.valPi`, `evi.valTrial`

Examples

```
############################################################################
## Example tied to Table 8.2. in Peeters, Klaassen, & van de Wiel (2015)
############################################################################

## Create data objects
## Means
means.exp1a.glob   <- c(668, 698, 756)
means.exp1b.globRT <- c(496, 526, 582)
means.exp3a        <- c(6.5, 7.5, 8.8)
means.exp3b.GTC    <- c(6.3, 7.5, 8.6)
means.exp4a.PA     <- c(2.08, 2.48, 3.33)
means.exp4a.NA     <- c(1.91, 2.70, 2.95)
means.exp4b.Aty    <- c(2.19, 2.44, 2.74)
means.exp5a        <- c(0.02, 0.13, 0.25)
means.exp5b        <- c(0.0, 0.14, 0.29)

## Standard deviations
sds.exp1a.glob   <- c(70, 123, 156)
sds.exp1b.globRT <- c(78, 86, 104)
sds.exp3a        <- c(1.9, 1.2, 0.08)
sds.exp3b.GTC    <- c(0.24, 0.11, 0.11)
sds.exp4a.PA     <- c(0.76, 0.36, 0.99)
sds.exp4a.NA     <- c(0.35, 0.49, 0.49)
sds.exp4b.Aty    <- c(0.47, 0.22, 0.26)
sds.exp5a        <- c(0.23, 0.22, 0.20)
sds.exp5b        <- c(0.19, 0.18, 0.18)

## Evidential values
```

```
EV.exp1a.glob    <- evi.val(means.exp1a.glob, sds.exp1a.glob, n.fl = 45/3)
EV.exp1b.globRT  <- evi.val(means.exp1b.globRT, sds.exp1b.globRT, n.fl = 48/3)
EV.exp3a         <- evi.val(means.exp3a, sds.exp3a, n.fl = 42/3)
EV.exp3b.GTC     <- evi.val(means.exp3b.GTC, sds.exp3b.GTC, n.fl = 53/3)
EV.exp4a.PA      <- evi.val(means.exp4a.PA, sds.exp4a.PA, n.fl = 72/6)
EV.exp4a.NA      <- evi.val(means.exp4a.NA, sds.exp4a.NA, n.fl = 72/6)
EV.exp4b.Aty     <- evi.val(means.exp4b.Aty, sds.exp4b.Aty, n.fl = 36/3)
EV.exp5a         <- evi.val(means.exp5a, sds.exp5a, n.fl = 60/3)
EV.exp5b         <- evi.val(means.exp5b, sds.exp5b, n.fl = 42/3)

## Overall evidential value
EVvec <- c(EV.exp1a.glob,
           EV.exp1b.globRT,
           EV.exp3a,
           EV.exp3b.GTC,
           EV.exp4a.PA,
           EV.exp4a.NA,
           EV.exp4b.Aty,
           EV.exp5a,
           EV.exp5b)
EVoverall <- evi.valMult(EVvec)

## Result
EVoverall
```

---

evi.valP   *Calculate upper bound to probability for a given evidential value*

Description

Function that calculates the upper bound to the probability for obtaining an evidential value that is at least as extreme as a specified value.

Usage

```
evi.valP(V, truncation = FALSE)
```

Arguments

V	A numeric scalar indicating the evidential value.
truncation	A logical indicating if upper bound should be calculated under a truncated and shifted 'standard' normal.

Details

The upper-bound to the evidential value is given by (Klaassen, 2015):

$$V = \frac{S_0}{\sqrt{nz^2}} \exp\left\{-\frac{1}{2}nz^2\left[\frac{1}{nz^2} - \frac{1}{S_0^2}\right]\right\} = \frac{S_0}{\sqrt{nz^2}} \exp\left\{\frac{1}{2}\left[\frac{nz^2}{S_0^2} - 1\right]\right\} \geq 1,$$

with $z = \bar{x}_1 - 2\bar{x}_2 + \bar{x}_3$, $S_0^2 = s_1^2 + 4s_2^2 + s_3^2$, and where $\bar{x}_i$ is a realization of factor-level mean μ_i while s_i^2 is an estimate of the associated variance. With n we denote the number of observations for each factor-level. Under the hypothesis of linearity $\sqrt{n}|z|/S_0$ has approximately a standard normal distribution.

One may be interested in determining the probability on a V at least as extreme as a given V^*. To this end one can solve the upper-bound to the evidential value (as given above) for $Z = \sqrt{n}|z|/S_0$ and relate the resulting score to a probability distribution. The solution is

$$Z^* = \sqrt{-W(-e^{-2c})},$$

with $c = 1/2 + \ln(V^*)$ and where $W(\cdot)$ denotes the Lambert W function. Under the default settings the resulting Z-score is treated as a quantile of the standard normal distribution. The function then returns $P(|Z| < Z^*)$ (or, equivalently, $P(Z < Z^*)$ for the standard normal truncated to the positive range).

We may also probe the behavior of the solution Z^* at the left and right-hand limits. It can be verified that (i) Z^* goes to 0 as V^* goes to infinity from the left and (ii) Z^* goes to 1 as V^* goes to unity from the right. This gives two insights: (1) evaluating Z^* on a probability distribution centered around 0 implies that high evidential values are considered common (as they are associated with high probability mass), and (ii) $Z^* \in [0, 1]$. When truncation = TRUE then Z^* is treated as a quantile on a mean-shifted standard normal (mean located at 1) truncated to the range $[0, 1]$. This takes the truncation explicitly into account while considering high evidential values to be more rare (more probability mass is now located at Z-values that indicate evidential values near 1). Note that the default option is thus much more lenient in evaluating the probabilities associated with high evidential values.

Value

The function returns a list in which:

$Z	is a numeric scalar representing the value of $\sqrt{n}|\bar{x}_1 - 2\bar{x}_2 + \bar{x}_3|/S_0$ associated with the specified V
$p	is a numeric scalar representing the upper bound to the probability for obtaining an evidential value at least as extreme as the specified V

Author(s)

Carel F.W. Peeters <cf.peeters@vumc.nl>, Chris A.J. Klaassen, Mark A. van de Wiel

References

Klaassen, C.A.J. (2015). Evidential value in ANOVA-regression results in scientific integrity studies. arXiv:1405.4540v2 [stat.ME].

Peeters, C.F.W., Klaassen, C.A.J., & van de Wiel, M.A. (2015). Evaluating the Scientific Veracity of Publications by dr. Jens Foerster. Available from the University of Amsterdam.

See Also

evi.val, evi.valMult, evi.valPi, evi.valTrial

Examples

```
## Obtain upper bound to probability on an evidential value being at least as extreme as 6
evi.valP(6)
```

---

evi.valPi   *Calculate lower-bound to evidential value for a given probability*

Description

Function that calculates, for a given probability, the lower-bound of the evidential value. It is the 'reverse' function to `evi.valP`.

Usage

```
evi.valPi(p, truncation = FALSE)
```

Arguments

p   A `numeric` scalar between 0 and 1 indicating the upper bound to probability.

truncation   A `logical` indicating if the lower-bound to V should be calculated under a truncated and shifted 'standard' normal.

Details

This function reverses the `evi.valP` function. It calculates, for a specified probability p the associated evidential value. Hence, for a given probability (that should be understood as an upper-bound) is finds the associated lower-bound for the evidential value.

Value

The function returns a `numeric` scalar representing the lower-bound to the evidential value associated with the specified probability p.

Author(s)

Carel F.W. Peeters <cf.peeters@vumc.nl>, Chris A.J. Klaassen, Mark A. van de Wiel

References

Klaassen, C.A.J. (2015). Evidential value in ANOVA-regression results in scientific integrity studies. arXiv:1405.4540v2 [stat.ME].

Peeters, C.F.W., Klaassen, C.A.J., & van de Wiel, M.A. (2015). Evaluating the Scientific Veracity of Publications by dr. Jens Foerster. Available from the University of Amsterdam.

See Also

`evi.val`, `evi.valMult`, `evi.valP`, `evi.valTrial`

Examples

```
## What lower-bound of V is associated with a (upper-bound) probability of 0.08093507
evi.valPi(0.08093507)
```

---

evi.valTrial	*Calculate upper-bound to probability of evidential value 'successes' in independent trials*

Description

Function calculating the upper-bound to the probability that at least X out of Y independent evidential values have a value above A.

Usage

```
evi.valTrial(subV, noV, trials, Round = 5, truncation = FALSE)
```

Arguments

subV	A `numeric` scalar indicating the value of V (considered substantial).
noV	A scalar `numeric` or `integer` indicating the minimum number of substantial evidential values.
trials	A scalar `numeric` integer indicating the no. of (sub)experiments/trials.
Round	A scalar `numeric` integer indicating the no. of decimals retained in representing the probability associated with the `subV` argument.
truncation	A `logical` indicating if the upper bound should be calculated under a truncated and shifted 'standard' normal.

Details

This function uses a binomial distribution with `trials` and `evi.valP(subV)$p` as the trial and success parameters. It calculates the upper-bound to the probability that at least `noV` out of `trials` independent evidential values have a value above `subV`.

Value

The function returns a `numeric` scalar representing the stated upper-bound to the desired probability.

Note

When `noV` contains decimal numbers only the integer part is retained.

Author(s)

Carel F.W. Peeters <cf.peeters@vumc.nl>, Chris A.J. Klaassen, Mark A. van de Wiel

References

Klaassen, C.A.J. (2015). Evidential value in ANOVA-regression results in scientific integrity studies. arXiv:1405.4540v2 [stat.ME].

Peeters, C.F.W., Klaassen, C.A.J., & van de Wiel, M.A. (2015). Evaluating the Scientific Veracity of Publications by dr. Jens Foerster. Available from the University of Amsterdam.

See Also

[evi.val](), [evi.valMult](), [evi.valP](), [evi.valPi]()

Examples

```
###################################################
## Obtaining information in the Table on page 3 of
## Peeters, Klaassen, & van de Wiel (2015)
###################################################

## 1 trial with 1 success, evidential value at least 6
evi.valTrial(6, 1, trials = 1)

## 5 trials with at least 2 successes, evidential value at least 6
evi.valTrial(6, 2, trials = 5)

## 11 trials with at least 3 successes, evidential value at least 6
evi.valTrial(6, 3, trials = 11)

## 21 trials with at least 4 successes, evidential value at least 6
evi.valTrial(6, 4, trials = 21)
```

Fisher.method   *Perform Fisher's combined probability test*

Description

Function that performs Fisher's combined probability test.

Usage

```
Fisher.method(pvalVec, plot = FALSE, x.label = "p-values", main = "")
```

Arguments

pvalVec	A numeric vector containing p-values.
plot	A logical indicating if the p-values should be plotted.
x.label	A character specifying the label for the x-axis. Only used when plot = TRUE.
main	A character specifying the main title.

Details

Suppose we have p-values for the ΔF test $(p(\Delta F))$ on $e = 1, \ldots, E$ *independent* samples for which *similar* null-hypotheses of linearity hold. We may then be interested to what extent the accumulation of tests favors the shared null hypothesis. To this end Fisher's method may be employed (Fisher, 1925). The statistic for this method is

$$G = -2 \sum_{e=1}^{E} \ln[p(\Delta F)_e],$$

which follows a χ^2 distribution with $2E$ degrees of freedom:

$$G \sim \chi^2_{2E}.$$

The p-value for the shared null hypothesis can be obtained in reference to this distribution. The left-tail probability $1 - p(G)$ then indicates how strongly the accumulation of tests favors the shared null.

When the observed trend moves away from linearity, $p(\Delta F)$ will move towards 0. When the observed trend will move towards perfect linearity, $p(\Delta F)$ will move towards 1. When the null hypothesis of perfect linearity holds in the population then, by definition, the $p(\Delta F)$ must be uniformly distributed. Observing p-values that consistently creep towards 1 then raises suspicion. The deviance of consistently high $p(\Delta F)$-values can be formalized with the Fisher method. Combining results on independent samples and usage of left-tail probabilities then indicates how strongly the accumulation of tests favors the shared null.

When `plot = TRUE` a histogram of the $p(\Delta F)$-values is produced. The blue line in the plot represents expectation under uniform behavior.

The function is based on the approach stated in the Whistleblower report (2012). For additional details see Peeters, Klaassen, & van de Wiel (2015).

Value

The function returns a `numeric`. This `numeric` represents the probability $p(G)$. Then, $1 - p(G)$ represents the left-tail probability.

Author(s)

Carel F.W. Peeters <cf.peeters@vumc.nl>, Chris A.J. Klaassen, Mark A. van de Wiel

References

Fisher, R. (1925). Statistical methods for research workers. Edinburg: Oliver and Boyd. Available from this link.

Peeters, C.F.W., Klaassen, C.A.J., & van de Wiel, M.A. (2015). Evaluating the Scientific Veracity of Publications by dr. Jens Foerster. Available from the University of Amsterdam.

Whistleblower Report. (2012). Suspicion of scientific misconduct by dr. Jens Foerster. Available from Retraction Watch.

See Also

deltaF.linear

Examples

```
##########################################################################
## Example tied to Table 8.2. in Peeters, Klaassen, & van de Wiel (2015)
##########################################################################

## Create data objects
## Means
means.exp1a.glob   <- c(668, 698, 756)
means.exp1b.globRT <- c(496, 526, 582)
means.exp3a        <- c(6.5, 7.5, 8.8)
means.exp3b.GTC    <- c(6.3, 7.5, 8.6)
```

```
means.exp4a.PA      <- c(2.08, 2.48, 3.33)
means.exp4a.NA      <- c(1.91, 2.70, 2.95)
means.exp4b.Aty     <- c(2.19, 2.44, 2.74)
means.exp5a         <- c(0.02, 0.13, 0.25)
means.exp5b         <- c(0.0, 0.14, 0.29)

## Standard deviations
sds.exp1a.glob   <- c(70, 123, 156)
sds.exp1b.globRT <- c(78, 86, 104)
sds.exp3a        <- c(1.9, 1.2, 0.08)
sds.exp3b.GTC    <- c(0.24, 0.11, 0.11)
sds.exp4a.PA     <- c(0.76, 0.36, 0.99)
sds.exp4a.NA     <- c(0.35, 0.49, 0.49)
sds.exp4b.Aty    <- c(0.47, 0.22, 0.26)
sds.exp5a        <- c(0.23, 0.22, 0.20)
sds.exp5b        <- c(0.19, 0.18, 0.18)

## Delta F testing
dF.exp1a.glob   <- deltaF.linear(means.exp1a.glob, sds.exp1a.glob, n.fl = 45/3)
dF.exp1b.globRT <- deltaF.linear(means.exp1b.globRT, sds.exp1b.globRT, n.fl = 48/3)
dF.exp3a        <- deltaF.linear(means.exp3a, sds.exp3a, n.fl = 42/3)
dF.exp3b.GTC    <- deltaF.linear(means.exp3b.GTC, sds.exp3b.GTC, n.fl = 53/3)
dF.exp4a.PA     <- deltaF.linear(means.exp4a.PA, sds.exp4a.PA, n.fl = 72/6)
dF.exp4a.NA     <- deltaF.linear(means.exp4a.NA, sds.exp4a.NA, n.fl = 72/6)
dF.exp4b.Aty    <- deltaF.linear(means.exp4b.Aty, sds.exp4b.Aty, n.fl = 36/3)
dF.exp5a        <- deltaF.linear(means.exp5a, sds.exp5a, n.fl = 60/3)
dF.exp5b        <- deltaF.linear(means.exp5b, sds.exp5b, n.fl = 42/3)

## Employ Fisher method
dFpVec <- c(dF.exp1a.glob$deltaF.p,
            dF.exp1b.globRT$deltaF.p,
            dF.exp3a$deltaF.p,
            dF.exp3b.GTC$deltaF.p,
            dF.exp4a.PA$deltaF.p,
            dF.exp4a.NA$deltaF.p,
            dF.exp4b.Aty$deltaF.p,
            dF.exp5a$deltaF.p,
            dF.exp5b$deltaF.p)
Fisher <- Fisher.method(dFpVec)

## Result (left-tail probability)
1 - Fisher
```

---

lackNoise   *Assess excessiveness of linearity and similarity*

Description

Function that assesses, by simulation, excessiveness of linearity and similarity.

Usage

```
lackNoise(meanMat, sdMat, n.flVec, F.vec, seed = sample(1:1e+06, 1),
          nrRuns, cat.data = FALSE, lower.cat, upper.cat,
          dir = getwd(), nameExt = "", verbose = TRUE)
```

Arguments

meanMat	A matrix containing means. Assumes that rows represent individual studies while columns contain factor level means.
sdMat	A matrix containing standard deviations. Assumes that rows represent individual studies while columns contain factor level standard deviations.
n.flVec	A numeric vector representing the number of replicates for each factor-level in each study.
F.vec	A numeric vector representing the reported omnibus F tests.
seed	An (random) integer used to initialize simulations at a particular seed.
nrRuns	An integer indicating the number of simulation runs desired.
cat.data	A logical indicating if the original data are of the categorical (Likert) type.
lower.cat	A numeric or integer vector representing the lowest reponse categories. Only used if cat.data = TRUE.
upper.cat	A numeric or integer vector representing the highest reponse categories. Only used if cat.data = TRUE.
dir	A character specifying the directory in which the visual output is stored.
nameExt	A character specifying the extension to default output names. Circumvents overwriting of output when working in single directory.
verbose	A logical indicating if progress should be printed on-screen.

Details

This function is based on ideas by Leif Nelson and Uri Simonsohn (2013, 2014) as well as code by Uri Simonsohn (2014). It performs simulations that aim to recreate an entire publication with all included studies. In many instances a publication is composed of multiple (sets of) independent one-way (sub)experiments. The simulations assess the (un)likeliness of reported linearity across conditions. The simulations also assess the (un)likeliness of the stability of reported omnibus F-values. The linearity measure is the average absolute distance between the middle (or control) condition and the midpoint between the high and low conditions. The stability measure is the standard deviation of omnibus F-tests. The function will draw samples from populations with means and standard deviations equal to their reported counterparts (in the publication under investigation). The population mean of the middle (or control) condition is assumed to be exactly in between the low and high conditions, i.e., $\mu_1 - 2\mu_2 + \mu_3 = 0$. Hence, the effects are assumed to be perfectly linear in the population. The simulations then proceed on the basis of the following (additional) assumptions:

- An ANOVA-type setting with one-way factorial design and 3 factor-levels;
- Random assignment of subjects to experimental conditions;
- A balanced design;
- The row-vectors of meanMat are ordered according to linearity of effect.

Each simulation run represents a simulated publication. The function then simply assesses how often we observe a linearity measure (stemming from the nrRuns simulations) equal or smaller than the linearity measure reported in the publication under investigation. This number raises suspicion when it is small relative to the number of simulations runs. The function also assesses how often we observe a stability measure (stemming from the nrRuns simulations) equal or smaller than the stability measure reported in the publication under investigation. Again, this number raises suspicion when it is small relative to the number of simulations runs.

The simulations truncate the datadraws to Likert-type data when `cat.data = TRUE`. When `n.flVec` contains decimal numbers only the integer part is retained. The function produces two PDF plots that are written away to the directory specified with the `dir` argument.

Value

The function produces 2 PDF plots that are written away to the directory specified with the `dir` argument. The 'Deviance_from_linearity' file contains a histogram of the average absolute distances between the middle (or control) condition and the midpoint between the high and low conditions as generated by the simulations. The red vertical line represents the average absolute distance between the middle (or control) condition and the midpoint between the high and low conditions as reported in the publication under investigation. The 'Similarity_F_values' file contains the analogous histrogram regarding the stability measure. The function also returns an object of class `list`:

`$seed.used`	an `integer` representing the seed with which the simulations were performed
`$nr.Sims.MDist.EQS.reported`	
	a numeric scalar representing the number of simulations in which we observe a linearity measure equal or smaller than the linearity measure reported in the publication under investigation
`$min.MDist.Sims`	
	a numeric scalar representing the minimum simulated linearity measure
`$nr.Sims.sdF.EQS.reported`	
	a numeric scalar representing the number of simulations in which we observe a stability measure equal or smaller than the stability measure reported in the publication under investigation
`$min.sdF.Sims`	a numeric scalar representing the minimum simulated stability measure
`$cor.MDist.sdF`	a numeric scalar representing the Spearman correlation between the simulated linearity and stability measures. If these measures are uncorrelated then they can be taken as independent evidence of low or high data veracity

Author(s)

Carel F.W. Peeters <cf.peeters@vumc.nl>, Chris A.J. Klaassen, Mark A. van de Wiel

References

Nelson, L., & Simonsohn, U. (2014). [21] Fake-Data Colada.

Simonsohn, U. (2013). Just Post it: The Lesson from Two Cases of Fabricated Data Detected by Statistics Alone. Psychological Science, 24:1875-1888.

Simonsohn, U. (2014). R Code Behind DataColada[21]

Examples

```
## Not run:
##########################
## Consider some data
##########################

## Vectors of means for low, mid, and high conditions
## Bound (by column) in matrix of means
M.low    = c(2.47, 2.51, 2.4, 2.41, 2.14, 3.19, 2.63, 2.87, 2.35, 2.55, 2.66, 2.42)
M.med    = c(3.04, 2.95, 2.90, 2.98, 2.82, 4.01, 3.73, 3.83, 3.66, 3.72, 3.69, 3.73)
M.high   = c(3.68, 3.35, 3.45, 3.64, 3.41, 4.79, 4.73, 4.79, 4.76, 4.78, 4.81, 5.02)
```

```
meanMat    <- cbind(M.low, M.med, M.high)

## Vectors of standard deviations for low, mid, and high conditions
## Bound (by column) in matrix of standard deviations
SD.low   = c(1.21, 0.71, 0.86, 1.07, 1.2, 1.07, 1.49, 1.24, 1.01, 1.16, 1.21, .82)
SD.med   = c(0.72, 0.49, 0.51, 0.51, 0.78, 1.21, 1.21, 1.09, 1.19, 1,   1.3, 1.28)
SD.high  = c(0.68, 0.64, 0.8, 0.95, 0.71, 0.82, 1.55, 1.53, 1.71, 1.47, 1.54, 1.45)
sdMat      <- cbind(SD.low, SD.med, SD.high)

## Specifying number of replicates for each factor level and omnibus F-tests
## Specyfing information on high and low categories Likert-type data
n.flVec   <- c(20,20,20,20,20,20,20,20,20,15.4,20,15)
lower.cat <- c(0,0,0,0,0,1,1,1,1,1,1,1)
upper.cat <- c(9,9,9,9,9,7,7,7,7,7,7,7)
F.vec     <- c(2,3,4,2,5,2,6,2,7,2,22,2)


###########################
## Run function
###########################

Test <- lackNoise(meanMat, sdMat, n.flVec, F.vec, seed = sample(1:1000000, 1),
                  nrRuns = 3000, cat.data = TRUE,
                  lower.cat, upper.cat, verbose = TRUE)

## Visualizations indicate deviance of linearity measure
## Also indicated by low number of simulated linearity measures equal or below reported one
Test$nr.Sims.MDist.EQS.reported

## End(Not run)
```

---

visual.trend   *Visualize the (linear) trend in (reported) means*

Description

Function that visualizes the trend in (possibly ordered) summary statistics. The main usage lies in visualizing the trend among means in ANOVA-type settings with a one-way factorial design. The visualization is by way of a line plot.

Usage

```
visual.trend(meanVec, sdVec, factor.label = as.character(1:length(meanVec)),
             x.label = "", y.label = "", main = "")
```

Arguments

meanVec	A numeric vector containing means for each factor level.
sdVec	A numeric vector containing standard deviations for each factor level.
factor.label	A character string for naming the factor levels.
x.label	A character specifying the label for the x-axis.
y.label	A character specifying the label for the y-axis.
main	A character specifying the main title.

Details

As stated: the main usage lies in visualizing the trend among means in ANOVA-type settings with a one-way factorial design. It can then be used to visualize the trend among the means when only summary/descriptive statistics are available (from the publication under investigation). In case one is interested in the linearity of effect among the means it is assumed that the input vectors are ordered, e.g., meanVec[1] < meanVec[2] < ... < meanVec[k].

The function produces a simple line plot using ggplot2 (Wickham, 2009). The 'error' bars represent one standard deviation from the mean.

Value

The function returns a graph.

Author(s)

Carel F.W. Peeters <cf.peeters@vumc.nl>, Chris A.J. Klaassen, Mark A. van de Wiel

References

Peeters, C.F.W., Klaassen, C.A.J., & van de Wiel, M.A. (2015). Evaluating the Scientific Veracity of Publications by dr. Jens Foerster. Available from the University of Amsterdam.

Wickham, H. (2009). ggplot2: elegant graphics for data analysis. New York: Springer.

See Also

`visual.trend.pooled`

Examples

```
## Means and standard deviations
Means <- c(2.08, 2.48, 3.33)
SDs   <- c(0.76, 0.36, 0.99)

## Visualize trend
visual.trend(Means, SDs, factor.label = c("Condition 1", "Condition 2", "Condition 3"))
```

---

visual.trend.pooled   *Visualize the pooled (linear) trend in (reported) means*

Description

Function that visualizes the pooled trend in (possibly ordered) summary statistics. The main usage lies in visualizing the pooled trend among means in ANOVA-type settings with a two-way factorial design. The visualization is by way of a line plot.

Usage

```
visual.trend.pooled(meanVec1, meanVec2,
                    factor.label = as.character(1:length(meanVec1)),
                    x.label = "", y.label = "", main = "")
```

Arguments

meanVec1	A numeric vector containing means for each factor level of group 1.
meanVec2	A numeric vector containing means for each factor level of group 2.
factor.label	A character string for naming the factor levels.
x.label	A character specifying the label for the x-axis.
y.label	A character specifying the label for the y-axis.
main	A character specifying the main title.

Details

As stated: the main usage lies in visualizing the pooled trend among means in ANOVA-type settings with a two-way factorial design. It can then be used to tentatively visualize the pooled trend among the means when pooled data are not directly reported. For example, in a 3 between × 2 between design one might be interested in visualizing the trend in means over the first factor when pooling (collapsing) over the second factor. In case one is interested in the linearity of effect among the pooled means it is assumed that the input vectors are ordered, e.g., meanVec[1] < meanVec[2] < ... < meanVec[k].

The function produces a simple line plot using ggplot2 (Wickham, 2009). Three lines are depicted representing the trend over the factor levels of factor 1 for the first selected level of factor 2 (blue), the trend over the factor levels of factor 1 for the second selected level of factor 2 (red), and the pooled (black) trend, respectively. The pooling is performed simply by averaging of the mean vectors.

Value

The function returns a graph.

Author(s)

Carel F.W. Peeters <cf.peeters@vumc.nl>, Chris A.J. Klaassen, Mark A. van de Wiel

References

Peeters, C.F.W., Klaassen, C.A.J., & van de Wiel, M.A. (2015). Evaluating the Scientific Veracity of Publications by dr. Jens Foerster. Available from the University of Amsterdam.

Wickham, H. (2009). ggplot2: elegant graphics for data analysis. New York: Springer.

See Also

visual.trend

Examples

```
## Means
Means1 <- c(2.08, 2.48, 3.33)
Means2 <- c(1.91, 2.70, 2.95)

## Visualize pooled trend
visual.trend.pooled(Means1, Means2,
                    factor.label = c("Condition 1", "Condition 2", "Condition 3"))
```

Index