Package 'rt.test'

October 14, 2022

Title Robustified t-Test

Version 1.18.7.9

Date 2018-07-09

Author Chanseok Park [aut, cre],
Min Wang [ctb]

Maintainer Chanseok Park <statpnu@gmail.com>

Depends R (>= 3.2.3)

Description Performs one-sample t-test based on robustified statistics using median/MAD (TA) and Hodges-Lehmann/Shamos (TB). For more details, see Park and Wang (2018)<arXiv:1807.02215>. This work was partially supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (No. NRF-2017R1A2B4004169).

License GPL-2 | GPL-3

URL https://github.com/statpnu/R-package

LazyData yes

NeedsCompilation no

Repository CRAN

Date/Publication 2018-07-10 17:30:03 UTC

R topics documented:

Hodges-Lehmann-estimate *Hodges-Lehmann estimate*

Description

Calculates the Hodges-Lehmann estimate.

Usage

```
HL.estimate(x, na.rm = FALSE, IncludeEqual = FALSE)
```

Arguments

x	vector of observations.
na.rm	a logical value indicating whether NA values should be stripped before the computation proceeds.
IncludeEqual	FALSE (default) calculates median of ([x[i]+x[j])/2) with i<j. TRUE calculates median of ([x[i]+x[j])/2) with i<=j.

Value

If x is not logical (coerced to numeric), numeric (including integer) or complex, NA_real_ is returned, with a warning.

Author(s)

Chanseok Park and Min Wang

References

Hodges, J. L. and E. L. Lehmann (1963). Estimates of location based on rank tests. *Annals of Mathematical Statistics*, **34**, 598–611.

See Also

mean for calculating sample mean.

Examples

```
x = c(0:10, 50)
HL.estimate(x)
```

`q.robustified.t` *Lower quantiles of TA or TB*

Description

Calculates the quantiles of the robustified t-test statistic (TA or TB).

Usage

```
q.robustified.t(p, n, test.stat=c("TA", "TB"), lower.tail=TRUE)
```

Arguments

`p`	vector of probabilities.
`n`	the sample size
`test.stat`	a character string specifying the test statistic.
`lower.tail`	logical; if TRUE (default), probabilities are p=P[X <= x], otherwise, p=P[X > x].

Details

Using the empirical distributions of TA and TB statistics, it calculates the quantile.

Author(s)

Chanseok Park and Min Wang

References

Park, C. and M. Wang (2018). Empirical distributions of the robustified *t*-test statistics. *ArXiv e-prints*, 1807.02215. https://arxiv.org/abs/1807.02215

See Also

`qt` for obtaining quantile value of Student t-distribution.

Examples

```
# quantile value of TA (using median and MAD) statistic
q.robustified.t(p=0.01, n=10, test.stat="TA")

# quantile value of TB (using Hodges-Lehmann and Shamos) statistic
q.robustified.t(p=0.01, n=10, test.stat="TB")
```

Quantiles.TA *Quantile values of the robustified statistic, TA.*

Description

Quantiles of the robustified statistic, TA. They are obtained using the extensive Monte Carlo with 1E08 replicates.

Usage

```
Quantiles.TA
```

Format

This data frame contains 97 rows and 500 columns.

Author(s)

Chanseok Park and Min Wang

References

Park, C. and M. Wang (2018). Empirical distributions of the robustified *t*-test statistics. *ArXiv e-prints*, 1807.02215. https://arxiv.org/abs/1807.02215

Quantiles.TB *Quantile values of the robustified statistic, TB.*

Description

Quantiles of the robustified statistic, TB. They are obtained using the extensive Monte Carlo with 1E08 replicates.

Usage

```
Quantiles.TB
```

Format

This data frame contains 97 rows and 500 columns.

Author(s)

Chanseok Park and Min Wang

References

Park, C. and M. Wang (2018). Empirical distributions of the robustified *t*-test statistics. *ArXiv e-prints*, 1807.02215. https://arxiv.org/abs/1807.02215

`Robustified-t-test` *Robustified t-test*

Description

Performs robustified one-sample t-test on a vector of data.

Usage

```
rt.test(x, alternative = c("two.sided", "less", "greater"),
  mu = 0, test.stat = c("TA", "TB"), conf.level = 0.95)
```

Arguments

`x`	vector of quantiles.
`alternative`	a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less". You can specify just the initial letter.
`mu`	a number indicating the true value of the mean.
`test.stat`	a character string specifying the test statistic.
`conf.level`	confidence level of the interval.

Details

Based on the empirical distributions of the TA statistic (based on median and MAD) and the TB statistic (based on Hodges-Lehmann and Shamos), this function performs one-sample robustified t-test.

Value

A list with class "htest" containing the following components:

`statistic`	the value of the test statistic.
`parameter`	sample size (non-missing observations in the sample).
`p.value`	the p-value for the test.
`conf.int`	a confidence interval for the mean appropriate to the specified alternative hypothesis.
`estimate`	the specified hypothesized value of the median (TA) or the Hodges-Lehmann (TB).
`sample.size`	numeric scalar containing the number of non-missing observations in the sample used for the hypothesis test

`null.value`	the specified hypothesized value of the true mean.
`alternative`	a character string describing the alternative hypothesis.
`method`	a character string indicating which statistic (TA or TB) is used.
`data.name`	a character string giving the name(s) of the data.

Author(s)

Chanseok Park and Min Wang

References

Park, C. and M. Wang (2018). Empirical distributions of the robustified *t*-test statistics. *ArXiv e-prints*, 1807.02215. https://arxiv.org/abs/1807.02215

Jeong, R., S. B. Son, H. J. Lee, and H. Kim (2018). On the robustification of the z-test statistic. Presented at KIIE Conference, Gyeongju, Korea. April 6, 2018.

Park, C. (2018). Note on the robustification of the Student *t*-test statistic using the median and the median absolute deviation. *ArXiv e-prints*, 1805.12256. https://arxiv.org/abs/1805.12256

See Also

`t.test` for performing the Student t-test.
`prop.test` for testing the proportion.

Examples

```
# For robustified t-test (two-sided) using median and MAD (TA).
#    test.stat="TA" (default)
x = rnorm(10)
rt.test(x)

# For robustified t-test (two-sided) using Hodges-Lehmann and Shamos (TB).
x = rnorm(10)
rt.test(x, test.stat="TB")

# 90% CI (two sides).
x = rnorm(10)
rt.test(x, conf.level=0.9)

# 90% CI (one side).
x = rnorm(10)
rt.test(x, alternative="less", conf.level=0.9)
```

Index