Package 'EstimDiagnostics'

October 12, 2022

Type Package

Title Diagnostic Tools and Unit Tests for Statistical Estimators

Version 0.0.3

Author Dmitry Otryakhin [aut, cre] (<https://orcid.org/0000-0002-4700-7221>)

Maintainer Dmitry Otryakhin <d.otryakhin.acad@protonmail.ch>

Description Extension of 'testthat' package to make unit tests on empirical distributions of estimators and functions for diagnostics of their finite-sample performance.

License GPL-3

URL https://gitlab.com/Dmitry_Otryakhin/diagnostics-and-tests-for-statistical-estimators

Encoding UTF-8

Imports foreach (>= 1.5.1), reshape2 (>= 1.4.4), ggplot2 (>= 3.3.2), goftest (>= 1.2-2), testthat (>= 3.0.0), rlang

RoxygenNote 7.1.1

Suggests knitr, rmarkdown, doParallel, gridExtra

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2021-04-16 21:40:03 UTC

R topics documented:

estims_boxplot *Boxplot of estimates*

Description

Plot boxplots of estimators for different sample sizes.

Usage

```
estims_boxplot(data, sep = FALSE)
```

Arguments

`data`	data frame returned by `Estim_diagnost`
`sep`	indicates whether all plots will be stacked together or returned as elements of a list

Value

ggplot2 object

Examples

```
Nmc=400
s<-seq(from = 1, to = 10, by = 2)*1e3
Inference<-function(s){
rrr<-rnorm(n=s)
list(Mn=mean(rrr), Sd=sd(rrr))
}

data <- Estim_diagnost(Nmc, s, Inference)
estims_boxplot(data)

estims_boxplot(data, sep=TRUE)
```

estims_qqplot *QQ-plot of estimator empirical distributions*

Description

Plot QQ-plots of estimators’ empirical distributions for different sample sizes.

Usage

```
estims_qqplot(data, sep = FALSE, ...)
```

Arguments

`data`	data frame returned by `Estim_diagnost`
`sep`	indicates whether all plots will be stacked together or returned as elements of a list
`...`	parameters to pass to stat_qq function

Value

ggplot2 object

Examples

```
library(ggplot2)
Nmc=500
s<-c(1e3,4e3)

Inference<-function(s){
  rrr<-rnorm(n=s)
  list(Mn=mean(rrr), Sd=sd(rrr))
}

data <- Estim_diagnost(Nmc, s, Inference)
lisst <- estims_qqplot(data, sep=TRUE)
lisst[2][[1]] + geom_abline(intercept = 1)

pl_joint<-estims_qqplot(data)
pl_joint + geom_abline(slope=1)

pl_joint<-estims_qqplot(data, distribution = stats::qt, dparams = list(df=3, ncp=0.1))
pl_joint + geom_abline(slope=1)
```

`Estim_diagnost` *Sample estimators' values for different sample sizes*

Description

For every sample size value the function creates a sample and evaluates the estimators Nmc times.

Usage

```
Estim_diagnost(Nmc, s, Inference, packages = NULL)
```

Arguments

`Nmc`	number of repetitions
`s`	numeric vector of sample sizes
`Inference`	function of s creating a sample and evaluating estimators (see details)
`packages`	list of packages to pass to foreach loop

Value

data frame with estimators' values

Examples

```
Nmc=400
s<-c(1e2,1e3)

Inference<-function(s){
  rrr<-rnorm(n=s)
  list(Mn=mean(rrr), Sd=sd(rrr))
}
data <- Estim_diagnost(Nmc, s, Inference)
estims_qqplot(data)
estims_boxplot(data)

#
Inference<-function(s){
rrr<-2/0
list(Mn=mean(rrr), Sd=sd(rrr))
}
head(Estim_diagnost(Nmc, s, Inference))

#
Inference<-function(s){
rrr<-rnorm(n=s)
rrr[2]<-"dwq"
list(Mn=mean(rrr), Sd=sd(rrr))
}
head(Estim_diagnost(Nmc, s, Inference))
```

`expect_distfit` *Test a parametric distribution*

Description

Expectation checking whether a given sample comes from a certain parametric distribution. The underlying procedure is Anderson-Darling test of goodness-of-fit `ad.test`. The expectation throws an error when the test's p-value is smaller than the threshold p-value.

Usage

```
expect_distfit(sample, p_value = 0.001, nulldist, ...)
```

Arguments

`sample`	to test
`p_value`	threshold p-value of the test
`nulldist`	null distribution
`...`	parameters to pass to the null distribution

Value

Invisibly returns a p-value of the test.

Examples

```
 # Gaussianity test
## Not run:
x<-rnorm(n=1e4,5,6)
expect_distfit(sample=x, nulldist="pnorm", mean=5, sd=6.3)
expect_distfit(sample=x, nulldist="pnorm", mean=5, sd=6)

## End(Not run)

# Uniformity test
x<-runif(n=1e4,-1,6)
expect_distfit(sample=x, nulldist="punif", min=-1, max=6)
```

`expect_gaussian` *Test a Gaussian distribution*

Description

Expectation checking whether a given sample comes from Gaussian distribution with arbitrary parameters. The underlying procedure is Shapiro- Wilk's test of normality `shapiro.test`. The expectation throws an error when the test's p-value is smaller than the threshold p-value.

Usage

```
expect_gaussian(sample, p_value = 0.001)
```

Arguments

`sample`	to test
`p_value`	threshold p-value of the test

Details

shapiro.test allows the number of non-missing values to be between 3 and 5000.

Value

Invisibly returns a p-value of the test.

Examples

```
x<-rnorm(n=1e3,5,6)
expect_gaussian(sample=x)

#The following test doesn't pass
## Not run:
x<-runif(n=1e2,-1,6)
expect_gaussian(sample=x)

## End(Not run)
```

`expect_mean_equal` *Test a mean-value using t-test*

Description

Expectation checking whether values from a given sample have a certain mean or that two samples have the same mean. The underlying procedure is Student's t-test `t.test`. The expectation throws an error when the test's p-value is smaller than the threshold p-value.

Usage

```
expect_mean_equal(p_value = 0.001, ...)
```

Arguments

`p_value`	threshold p-value of the test
`...`	parameters to pass to t.test function including data sample(s)

Value

Invisibly returns a p-value of the test

Examples

```
# This test doesn't pass
## Not run:
x<-1:1e3
expect_mean_equal(x=x)

## End(Not run)

# This one passes, but shouldn't
```

```
x<-rnorm(1e3) + 0.01
expect_mean_equal(x=x)

x<-rnorm(1e3)
expect_mean_equal(x=x)

# check if 2 samples have the same mean
x<-rnorm(1e3, mean=10)
y<-rnorm(1e3, mean=10)
expect_mean_equal(x=x, y=y)
```

Index