Package 'finity'

March 6, 2020

Type Package

Title Test for finiteness of moments in a distribution

Version 0.1

Date 2019-11-14

Author Torsten Heinrich <torsten.heinrich@oxfordmartin.ox.ac.uk>,
Julian Winkler <julian.winkler@economics.ox.ac.uk>

Maintainer Torsten Heinrich <torsten.heinrich@oxfordmartin.ox.ac.uk>

Description Tests whether a given moment of the distribution of a given sample is finite or not.

License GPL (>= 2)

Imports Rcpp (>= 1.0.3)

LinkingTo Rcpp, RcppArmadillo, BH

RoxygenNote 7.0.0

Encoding UTF-8

R topics documented:

---

finity-package *Test for finiteness of moments in a distribution*

Description

Tests whether a given moment of the distribution of a given sample is finite or not.

Details

The DESCRIPTION file:

For heavy-tailed distributions with tail exponent a, only moments of order $< a$ are finite. The tail index and heavy- tailedness are is notoriously difficult to ascertain. But the finiteness of moments (including fractional moments) can be tested directly. This package does that following the test suggested by Trapani (2016).

The main function is finite_moment_test().

Author(s)

Torsten Heinrich <torsten.heinrich@oxfordmartin.ox.ac.uk>, Julian Winkler <julian.winkler@economics.ox.ac.uk>

Maintainer: Torsten Heinrich <torsten.heinrich@oxfordmartin.ox.ac.uk>

References

Trapani, Lorenzo (2016), 'Testing for (in)finite moments', *Journal of Econometrics* **191**(1), 57 – 68.

See Also

https://github.com/x0range/finity

Examples

```
library(stabledist)
# Generate sample
rvs <- rstable(50000000, 1.9, 0.5, 1, 0, pm = 0)
# Perform test
result <- finite_moment_test(rvs, 2)
# Print results
message(paste("Test statistic:", result[1], "p-value:", result[2], "\n\n"))

# More examples are included in https://github.com/x0range/finity/examples
```

---

compute_absolute_moment

Computes absolute moment of order k in sample of observations obs.

Description

Computes absolute moment of order k in sample of observations obs.

Usage

```
compute_absolute_moment(obs, k)
```

Arguments

obs	Observations (type: armadillo numeric vector).
k	Moment order (type: double)

Value

Moment value (type: double)

Examples

```
library(stabledist)
rvs <- rstable(50000000, 1.9, 0.5, 1, 0, pm = 0)
absolute_moment <- compute_absolute_moment(rvs, 2)
```

---

finite_moment_test *Computes Trapani's (2016) finite moment test*

Description

Computes Trapani's (2016) finite moment test

Usage

```
finite_moment_test(
  obs,
  k,
  r = 0L,
  psi = 2,
  u = 1,
  force_random_variate_sample = 0L,
  ignore_errors = 0L,
  verbose = 0L,
  random_salting = 0L
)
```

Arguments

obs	Observations (type: armadillo numeric vector).
k	Moment order (type: double)
r	Artificial sample size (type: int). Default is N^0.8.
psi	Pescaling moment (type: double). Must be <k. Default is 2.0.
u	Sampling range width for sampling range [-u, u] (type: double) Default is 1.0.
force_random_variate_sample	If True, draw random variates for xi and u_series. If False, use quantile function values from a regular percentile space grid. This represents the density function better. Defaiult is False.
verbose	If True, print detailed output for debugging. Default is False.
random_salting	Salt number to be added to the random seed (type: int). This prevents identical random variate series if multiple instances are started and run in parallel. Default is 0.

Value

Trapani's Theta test statistic (type: double).

Corresponding p-value (Chi^2(1) percentile) (type: double.

Examples

```
library(stabledist)
rvs <- rstable(50000000, 1.9, 0.5, 1, 0, pm = 0)
result <- finite_moment_test(rvs, 2)
```

---

get_chisq1_percentile *Returns Chi^2(1) percentile for test.*

Description

Returns Chi^2(1) percentile for test.

Usage

```
get_chisq1_percentile(value)
```

Arguments

value	Chi^2(1) value (type: double).

Value

Chi^2(1) percentile (type: double).

Examples

```
get_chisq1_percentile(20.0)
```

Index