Package 'npsr'

May 18, 2018

Version 0.1.1

Type Package

Title Validate Instrumental Variables using NPS

Maintainer Fabian Sellmann <fa.sllmnn@gmail.com>

Description An R implementation of the Necessary and Probably Sufficient (NPS) test for finding valid instrumental variables, as suggested by Amit Sharma (2016, Working Paper) <http://amitsharma.in/pubs/necessary_probably_sufficient_iv_test.pdf>. The NPS test, compares the likelihood that a given set of observational data of the three variables Z, X and Y is generated by a valid instrumental variable model (Z -> X -> Y) to the likelihood that the data is generated by an invalid IV model.

License GPL-2

Encoding UTF-8

LazyData true

Collate npsCommon.R npsInvalid.R npsMain.R npsNecessary.R npsValid.R RNested.R

Imports infotheo, MASS, gmp

RoxygenNote 6.0.1

NeedsCompilation no

Author Fabian Sellmann [aut, cre],
Bojan Nicolic [ctb, cph],
Jörn Grahl [ths]

Repository CRAN

Date/Publication 2018-05-18 14:33:18 UTC

R topics documented:

M_air 2
M_air_excl 2
M_excl 3
nps.invalid 3
nps.necessary 4

nps.test 4
nps.valid 5
product_fraction 5

Index **6**

`M_air` *M_air*

Description

Calculates the marginal likelihood M_air

Usage

```
M_air(Q, l, m, n)
```

Arguments

`Q`	Histogram of dataset (l*m*n vector)
`l`	|Z|
`m`	|X|
`n`	|Y|

Value

The probability that the observations where created from a model which violates the as-if-randomness criterion but not the exclusion criterion

---

`M_air_excl` *m_air*

Description

Calculates the marginal likelihood M_air_excel

Usage

```
M_air_excl(Q, l, m, n)
```

Arguments

`Q`	Histogram of dataset (l*m*n vector)
`l`	|Z|
`m`	|X|
`n`	|Y|

Value

The probability that the observations where created from a model which violates the as-if-randomness criterion but not the exclusion criterion

---

`M_excl` *M_excl*

Description

Calculates the marginal likelihood of M_excl

Usage

```
M_excl(Q, l, m, n, N = sum(Q), S = sum(Q))
```

Arguments

`Q`	Histogram of dataset (l*m*n vector)
`l`	|Z|
`m`	|X|
`n`	|Y|
`N`	Number of Repetitions for Nested Sampling
`S`	Number of Starting Points for Nested Sampling

Value

The probability that the observations where created from a model which violates the exclusion criterion but not the as-if-randomness criterion

---

`nps.invalid` *M_Invalid*

Description

Calculates the ML_Invalid

Usage

```
nps.invalid(Q, l, m, n, N = sum(Q), S = sum(Q))
```

Arguments

Q	List of unique observations, should be l*m*n length
l	|Z|
m	|X|
n	|Y|
N	Number of Repetitions for Nested Sampling
S	Number of Starting Points for Nested Sampling

---

nps.necessary *testIc*

Description

Tests the instrumental constraints on the given dataframe using entropy

Usage

```
nps.necessary(df)
```

Arguments

df	Dataframe with z, x and y

Value

FALSE if the data violates the constraints otherwise TRUE

---

nps.test *Main function of the package.*

Description

Main function of the package.

Usage

```
nps.test(df, l, m, n, N, S)
```

Arguments

df	Dataframe with columns z,x and y
l	Number of bins used to discretize Z
m	Number of bins used to discretize X
n	Number of bins used to discretize Y
N	Number of Repetitions for Nested Sampling
S	Number of Starting Points for Nested Sampling

Value

result object of the test including the fields: nt, valid, invalid, ratio

Examples

```
nps.test(data.frame(x = runif(3), y = runif(3), z = runif(3)),2,2,2, 3, 3)
```

---

nps.valid *nps.valid*

Description

Calculates M_Valid

Usage

```
nps.valid(Q, l, m, n, N = sum(Q), S = sum(Q))
```

Arguments

Q	Histogram of dataset (l*m*n vector)
l	|Z|
m	|X|
n	|Y|
N	Number of Repetitions for Nested Sampling
S	Number of Starting Points for Nested Sampling

---

product_fraction *Reduces out factors of fraction of products and calculates the fraction Analog to prod(num)/prod(den)*

Description

Reduces out factors of fraction of products and calculates the fraction Analog to prod(num)/prod(den)

Usage

```
product_fraction(num, den)
```

Arguments

num	vector of factors of the numerator
den	vector of factors of the denominator

Index

M_air, 2
M_air_excl, 2
M_excl, 3

nps.invalid, 3
nps.necessary, 4
nps.test, 4
nps.valid, 5

product_fraction, 5