Package ‘TOHM’

October 12, 2022

Type    Package
Title   Testing One Hypothesis Multiple Times
Version 1.4
Date    2021-03-09
Author  Sara Algeri
Maintainer Sara Algeri <salgeri@umn.edu>

Description

Imports igraph, Matrix, nloptr, Rmpfr, EQL, methods, stats
License GPL-2
RoxygenNote 6.1.1
NeedsCompilation no
Repository CRAN
Date/Publication 2021-03-09 20:30:05 UTC

R topics documented:

TOHM-package ...................................................... 2
chi2_ECden ............................................................ 3
ECden_vec .............................................................. 4
EC_LRT ................................................................. 5
EC_T ............................................................... 7
find_max ................................................................. 8
Gauss_ECden .......................................................... 9
global_p ............................................................... 10
gLRT ................................................................. 12
TOHM_LRT ........................................................... 13

Index 16
Description


Details

The functions collected in TOHM mainly focus on the implementation of the Likelihood Ratio Tests (see TOHM_LRT). However, several functions (e.g., EC_T, global_p) can be used to obtain global p-values for other test statistics and to compute the Euler characteristic using the graph algorithm described in Algeri and van Dyk (2018).

Author(s)

Sara Algeri Maintainer: Sara Algeri <salgeri@umn.edu>

References


Examples

#generating data of interest
N<-100
x<-as.matrix(cbind(runif(N*2,172.5,217.5),runif(N*2,-2,58)))
x2<-x[(x[,1]<=217.5)&(x[,1]>=172.5),]
x_sel<-x2[(x2[,2]<=(28+sqrt(30^2-(x2[,1]-195)^2)))&(x2[,2]>(28-sqrt(30^2-(x2[,1]-195)^2)))]
data<-x_sel[sample(seq(1:(dim(x_sel)[1])),N),]

#Specifying minus-log-likelihood
kg<-function(theta){integrate(Vectorize(function(x) {
exp(-0.5*((x-theta[1])/0.5)^2)*integrate(function(y) {
exp(-0.5*((y-theta[2])/0.5)^2) }, 28-sqrt(30^2-(x-195)^2),
28+sqrt(30^2-(x-195)^2))$value}) , 172.5, 217.5)$value}
mll<-function(eta,x,theta){
-sum(log((1-eta)/(pi*(30)^2)+eta*exp(-0.5*((x[,1]-theta[1])/0.5)^2-
0.5*((x[,2]-theta[2])/0.5)^2))/kg(theta)))

#Specifying search region
theta1<-seq(172.5,217.5,by=15)
theta2<-seq(-2,58,by=10)
THETA<-as.matrix(expand.grid(theta1,theta2))
originalR<-dim(THETA)[1]
rownames(THETA)<-1:(dim(THETA)[1])
THETA2<-THETA[(THETA[,1]<=217.5)&(THETA[,1]>=172.5),]
THETA_sel<-THETA2[(THETA2[,2]<=(28+sqrt(30^2-(THETA2[,1]-
195)^2)))&(THETA2[,2]>=(28-sqrt(30^2-(THETA2[,1]-195)^2))),]

#Generating toy EC
ECs<-cbind(rpois(100,1.5),rpois(100,1))
TOHM_LRT(data,mll,null0=0,init=c(0.1),lowlim=c(0),uplim=c(1),
THETA=THETA_sel,ck=c(1,8),type=c("Chi-bar^2"),
k=NULL,k_vec=c(0,1),weights=c(0.5,0.5),
ECdensities=NULL,ECs=ECs)

---

**chi2_ECden**

*Compute Euler characteristic density for Chi-square random fields*

**Description**

Computes the Euler characteristic (EC) density of a given order for Chi-squared random fields.

**Usage**

chi2_ECden(c, k, j)

**Arguments**

- `c` Value on which the EC density is evaluated.
- `k` Degrees of freedom of the Chi-square random field.
- `j` Order of the EC density to be implemented.

**Value**

Returns the value of the EC density of order `j` evaluated at `c` for a Chi-square random field with `k` degrees of freedom.

**Author(s)**

Sara Algeri

**References**


**See Also**

`Gauss_ECden, ECden_vec`
**Examples**

```r
c<-1
k<-1
j<-2
chi2_ECden(c,k,j)
```

**ECden_vec**

Compute the Euler characteristic densities

**Description**

Compute the Euler characteristic (EC) densities for Gaussian, Chi-square and Chi-bar-square random fields up to a given order.

**Usage**

```r
ECden_vec(c, D, type = c("Gaussian", "Chi^2", "Chi-bar^2"),
           k = NULL, k_vec = NULL, weights = NULL)
```

**Arguments**

- `c` Value on which the EC densities are evaluated.
- `D` Maximum order of the EC density to be computed.
- `type` Type of random field. The possible options are "Gaussian", "Chi^2", and "Chi-bar^2". See details.
- `k` If `type="Chi^2"`, degrees of freedom of the Chi-square random field.
- `k_vec` If `type="Chi-bar^2"` the degrees of freedom of the Chi-square random fields in the mixture.
- `weights` If `type="Chi-bar^2"` the weights of the mixture of Chi-square random fields in the mixture. The ordering should be the same as in `k_vec`.

**Details**

If `type="Chi-bar^2"` the degrees of freedom of the Chi-square random fields involved in the mixture, as well as the respective weights, have to be specified in the arguments `k_vec` and `weights`.

**Value**

Returns the values of the EC densities of order zero up to the dimension of the search area considered and evaluated at `c`.

**Author(s)**

Sara Algeri
**EC_LRT**

References


See Also

chi2_ECden, Gauss_ECden

Examples

ECden_vec(12,2,“Chi-bar^2”,k_vec=c(0,1),weights=c(0.5,0.5))

---

**EC_LRT**

*Compute the Euler characteristic for the generalized Likelihood Ratio Test field*

Description

It computes the Euler characteristic (EC) of the generalized Likelihood Ratio test (LRT) field above specified thresholds over a given search area.

Usage

EC_LRT(ck, x, mll, null0, init, lowlim, uplim, THETA)

Arguments

- **ck**: Vector of thresholds defining the excursions sets with respect to which the ECs are computed.
- **x**: A vector or matrix collecting the data on which the LRT is computed.
- **mll**: A function specifying the negative (profile) log-likelihood. See details.
- **null0**: A scalar or vector collecting the values of the free parameters under the null hypothesis. See details.
- **init**: Vector of initial values for the MLE.
- **lowlim**: Vector of lower bounds for the MLE.
- **uplim**: Vector of upper bounds for the MLE.
- **THETA**: A vector or matrix of grid values for the nuisance parameter with respect to which the search is performed.
Details

`mll` takes as first argument the vector of the parameters for which the MLE is generated. Other arguments of `mll` are the data vector or matrix (`x`) and a scalar or vector corresponding to the fixed value for the nuisance parameter with respect to which the profiling is computed (theta, see `gLRT`). If the latter is a vector it must be of same length of the rows in THETA. If the null model has nuisance parameters, `null0` takes as arguments the values of the parameters being tested under the null hypothesis, followed by the estimates of the nuisance parameters obtained assuming that the null hypothesis is true.

Value

Returns a vector of EC values with respect to the thresholds specified in `ck`.

Author(s)

Sara Algeri

References


See Also

EC_T

Examples

```R
# generating data of interest
N <- 100
x <- as.matrix(cbind(runif(N*2,172.5,217.5), runif(N*2,-2,58)))
x2 <- x [x[,1]<=217.5 & (x[,1]>=172.5), ]
x_sel <- x2 [x2[,2]<= (28+sqrt(30^2-(x2[,1]-195)^2)) & (x2[,2]>= (28-
sqrt(30^2-(x2[,1]-195)^2))) ]
data <- x_sel [sample(seq(1:(dim(x_sel)[1])), N), ]

# Specifying minus-log-likelihood
kg <- function(theta){
  integrate(Vectorize(function(x) {
    exp(-0.5*((x-theta[1])/0.5)^2)*integrate(function(y) {
      exp(-0.5*((y-theta[2])/0.5)^2), 28-sqrt(30^2-(x-195)^2),
      28+sqrt(30^2-(x-195)^2))$value}) , 172.5, 217.5)$value}
}
mll <- function(eta, x, theta){
  -sum(log((1-eta)/(pi*(30)^2)+eta*exp(-0.5*((x[,1]-theta[1])/0.5)^2-0.5*((x[,2]-theta[2])/0.5)^2)/kg(theta)))
}

# Specifying search region
theta1 <- seq(172.5, 217.5, by=15)
theta2 <- seq(-2, 58, by=10)
THETA <- as.matrix(expand.grid(theta1, theta2))
originalR <- dim(THETA)[1]
```
rownames(THETA)<-1:(dim(THETA)[1])
THETA2<-THETA[(THETA[,1]<=217.5)&(THETA[,1]>=172.5),]
THETA_sel<-THETA2[(THETA2[,2]<=(28+sqrt(30^2-(THETA2[,1]-195)^2)))&(THETA2[,2]>=(28-sqrt(30^2-(THETA2[,1]-195)^2))),]

EC_LRT(ck=c(1,8),x=data,mll=mll,null0=0,init=c(0.1),
lowlim=c(0),uplim=c(1), THETA_sel)

---

## EC_T

**Compute the Euler characteristic for a given field**

### Description

It computes the Euler characteristic (EC) of a given field above specified thresholds over a specified search area.

### Usage

`EC_T(ck, Ts, THETA)`

### Arguments

- `ck`: Vector of thresholds defining the excursions sets with respect to which the ECs are computed.
- `Ts`: Vector of values of the field for each grid point in `THETA`.
- `THETA`: A vector or matrix of grid values for the nuisance parameter with respect to which the search is performed.

### Value

Returns a vector of EC values with respect to the thresholds specified in `ck`.

### Author(s)

Sara Algeri

### References


### See Also

- [EC_LRT](#)

### Examples

`EC_T(ck=c(3,4),Ts=rnorm(10), THETA=cbind(1:10,21:30))`
find_max

Maximum of Likelihood Ratio Test field

Description
It computes the maximum of the generalized Likelihood Ratio Test (LRT) evaluated over a grid of values.

Usage
find_max(x, mll, null0, init, lowlim, uplim, THETA)

Arguments
- x: A vector or matrix collecting the data on which the LRT field is computed.
- mll: A function specifying the negative (profile) log-likelihood. See details.
- null0: A vector or scalar of the free parameters under the null hypothesis. See details.
- init: A vector or scalar of initial values for the MLE.
- lowlim: A vector or scalar of lower bounds for the MLE.
- uplim: A vector or scalar of upper bounds for the MLE.
- THETA: A vector or matrix of grid values of the nuisance parameter with respect to which the search is performed.

Details
mll takes as first argument the vector of the parameters for which the MLE is generated. Other arguments of mll are the data vector or matrix (x) and a scalar or vector corresponding to the fixed value for the nuisance parameter with respect to which the profiling is computed (theta, see gLRT). If the latter is a vector it must be of same length of the rows in THETA. If the null model has nuisance parameters, null0 takes as arguments the values of the parameters being tested under the null hypothesis, followed by the estimates of the nuisance parameters obtained assuming that the null hypothesis is true.

Value
- max_gLRT: Maximum observed of the LRT field.
- theta_max: Value of THETA at which the maximum is observed.

Author(s)
Sara Algeri

References
See Also

gLRT, global_p

Examples

```r
# generating data of interest
N<-100
x<-as.matrix(cbind(runif(N*2,172.5,217.5),runif(N*2,-2,58)))
x2<x[(x[,1]<=217.5)&(x[,1]>=172.5),]
x_sel<x2[(x2[,2]>(28+sqrt(30^2-(x2[,1]-195)^2))&(x2[,2]<(28-
sqrt(30^2-(x2[,1]-195)^2))),]
data<x_sel[sample(seq(1:(dim(x_sel)[1])),N),]

# specifying minus-log-likelihood
kg<-function(theta){integrate(Vectorize(function(x) {
  exp(-0.5*((x-theta[1])/0.5)^2)*integrate(function(y) {
    exp(-0.5*((y-theta[2])/0.5)^2) }, 28-sqrt(30^2-(x-195)^2),
    28+sqrt(30^2-(x-195)^2))$value}), 172.5, 217.5)$value}
mll<-function(eta,x,theta){
  -sum(log((1-eta)/(pi*(30)^2)+eta*exp(-0.5*((x[,1]-
    theta[1])/0.5)^2-
    0.5*((x[,2]-theta[2])/0.5)^2)/kg(theta)))
}

# specifying search region
theta1<-seq(172.5,217.5,by=15)
theta2<-seq(-2,58,by=10)
THETA<-as.matrix(expand.grid(theta1,theta2))
originalR<dim(THETA)[1]
rownames(THETA)<-1:(dim(THETA)[1])
THETA2<-THETA[(THETA[,1]<=217.5)&(THETA[,1]>=172.5),]
THETA_sel<--THETA2[(THETA2[,2]>(28+sqrt(30^2-(THETA2[,1]-
   195)^2))&(THETA2[,2]<(28-sqrt(30^2-(THETA2[,1]-195)^2))),]

find_max(x=data,mll=mll,null0=0,init=c(0.1),
  lowlim=c(0),uplim=c(1), THETA=THETA_sel)
```

Description

Computes the Euler characteristic (EC) density of a given order for Gaussian random fields.

Usage

Gauss_ECden(c, j)
Arguments

- **c**: Value on which the EC density is evaluated.
- **j**: Order of the EC density to be implemented.

**Value**

Returns the value of the EC density of order \( j \) evaluated at \( c \) for a Gaussian random field.

**Author(s)**

Sara Algeri

**References**


**See Also**

chi2_ECden, ECden_vec

**Examples**

```r
C<-1
j<-2
Gauss_ECden(C,4)
```

---

### global_p

**Compute global p-values**

**Description**

It computes the global p-value for a given value of the test statistic.

**Usage**

```r
global_p(c, ck, type = c("Gaussian", "Chi^2", "Chi-bar^2"), k = NULL, k_vec = NULL, weights = NULL, ECdensities = NULL, ECs)
```

**Arguments**

- **c**: Observed value of the test statistic.
- **ck**: Vector of thresholds defining the excursions sets with respect to which the ECs are computed.
- **type**: Type of random field. The possible options are "Gaussian", "Chi^2", and "Chi-bar^2". See details.
- **k**: If type="Chi^2", degrees of freedom of the Chi-square random field.
global_p

k_vec If type="Chi-bar^2" the degrees of freedom of the Chi-square random fields in the mixture.

weights If type="Chi-bar^2" the weights of the mixture of Chi-square random fields in the mixture. The ordering should be the same as in k_vec.

ECdensities See details.

ECs A vector or matrix containing the Euler characteristics (ECs) computed over a Monte Carlo simulation of the random field under the null model. Each column correspond to the ECs obtained for each of the thresholds in ck.

Details

If type="Chi-bar^2" the degrees of freedom of the Chi-square random fields involved in the mixture, as well as the respective weights, have to be specified in the arguments k_vec and weights. If the distribution of the random field is not available in type, the user can specify in ECdensities a function taking c as argument and returning the vector of the desired EC densities to be evaluated at c. Notice that the length of the vector returned by the function specified in ECdensities must corresponds to one plus the dimension of the search area (since the first value should correspond to the EC density of order zero (see ECden_vec)).

Value

global_p Global p-value.

MCerror Monte Carlo error associated to the global p-value.

Author(s)

Sara Algeri

References


See Also

find_max,TOHM_LRT,ECden_vec

Examples

ck<-c(1,2)
ECs<-cbind(rpois(100,1.5),rpois(100,1))
global_p(c=12,ck=ck,type="Gaussian",ECs=ECs)
Compute the generalized Likelihood Ratio Test

**Description**

Compute the generalized Likelihood Ratio Test (LRT) for a specified value of the nuisance parameter.

**Usage**

\[ \text{gLRT}(\theta, \text{mll}, x, \text{init}, \text{lowlim}, \text{uplim}, \text{null0}) \]

**Arguments**

- \( \theta \): A vector or scalar of the value of the nuisance parameter with respect to which the LRT is computed.
- \( \text{mll} \): A function specifying the negative (profile) log-likelihood. See details.
- \( x \): A vector or matrix collecting the data.
- \( \text{init} \): A vector or scalar of initial values for the MLE.
- \( \text{lowlim} \): A vector or scalar of lower bounds for the MLE.
- \( \text{uplim} \): A vector or scalar of upper bounds for the MLE.
- \( \text{null0} \): A vector or scalar of the free parameters under the null hypothesis. See details.

**Details**

mll takes as first argument the vector of the parameters for which the MLE is generated. Other arguments of mll are the data vector or matrix (x) and a scalar or vector corresponding to the fixed value for the nuisance parameter with respect to which the profiling is computed (\( \theta \), see gLRT). If the latter is a vector it must be of same length of the rows in THETA. If the null model has nuisance parameters, null0 takes as arguments the values of the parameters being tested under the null hypothesis, followed by the estimates of the nuisance parameters obtained assuming that the null hypothesis is true.

**Value**

The value of the generalized LRT for a specified value of \( \theta \).

**Author(s)**

Sara Algeri

**References**


TOHM_LRT

See Also

find_max, TOHM_LRT

Examples

#generating data of interest
N<-100
x<-as.matrix(cbind(runif(N*2,172.5,217.5),runif(N*2,-2,58)))
x2<-x[(x[,1]<=217.5)&(x[,1]==172.5),]
x_sel<-x2[(x2[,2]<=(28+sqrt(30^2-(x2[,1]-195)^2)))&(x2[,2]>(28-
sqrt(30^2-(x2[,1]-195)^2))),]
data<-x_sel[sample(seq(1:(dim(x_sel)[1])),N),]

#Specifying minus-log-likelihood
kg<-function(theta){integrate(Vectorize(function(x) {
  exp(-0.5*((x-theta[1])/0.5)^2)*integrate(function(y) {
    exp(-0.5*((y-theta[2])/0.5)^2 ), 28-
sqrt(30^2-(x-195)^2),
    28+sqrt(30^2-(x-195)^2))$value})) , 172.5, 217.5)$value}
mll<-function(eta,x,theta){
  -sum(log((1-eta)/(pi*(30)^2)+eta*exp(-0.5*((x[,1]-
    theta[1])/0.5)^2-
   0.5*((x[,2]-theta[2])/0.5)^2)/kg(theta)))
}
gLRT(theta=c(200,30),mll=mll,init=0.1,lowlim=0,uplim=1,null0=0,x=data)

TOHM_LRT

Compute the Likelihood Ratio Test under non-standard conditions.

Description

It implements the procedure described in Algeri and van Dyk (2018) to perform tests of hypothesis under non-regular conditions, and which can be formulated as test of hypothesis where a nuisance parameter is present only under the alternative.

Usage

TOHM_LRT(x, mll, null0, init, lowlim, uplim, THETA, ck,
  type = c("Chi^2", "Chi-bar^2"), k = NULL, k_vec = NULL,
  weights = NULL, ECdensities = NULL, ECs = NULL)

Arguments

x       A vector or matrix collecting the data.
mll     A function specifying the negative (profile) log-likelihood. See details.
null0   A vector or scalar of the free parameters under the null hypothesis. See details.
init    A vector or scalar of initial values for the MLE.
lowlim  A vector or scalar of lower bounds for the MLE.
uplim A vector or scalar of upper bounds for the MLE.

THETA A vector or matrix of grid values of the nuisance parameter with respect to which the search is performed.

ck Vector of thresholds defining the excursions sets with respect to which the ECs are computed.

type Type of random field. The possible options are "Gaussian", "Chi^2", and "Chi-bar^2". See details.

k If type="Chi^2", degrees of freedom of the Chi-square random field.

k_vec If type="Chi-bar^2" the degrees of freedom of the Chi-square random fields in the mixture.

weights If type="Chi-bar^2" the weights of the mixture of Chi-square random fields in the mixture. The ordering should be the same as in k_vec.

ECdensities See details.

ECs A vector or matrix containing the Euler characteristics (ECs) computed over a Monte Carlo simulation of the random field under the null model. Each column correspond to the ECs obtained for each of the thresholds in ck.

Details

mll takes as first argument the vector of the parameters for which the MLE is generated. Other arguments of mll are the data vector or matrix (x) and a scalar or vector corresponding to the fixed value for the nuisance parameter with respect to which the profiling is computed (theta, see glRT). If the latter is a vector it must be of same length of the rows in THETA. If the null model has nuisance parameters, null0 takes as arguments the values of the parameters being tested under the null hypothesis, followed by the estimates of the nuisance parameters obtained assuming that the null hypothesis is true. If type="Chi-bar^2" the degrees of freedom of the Chi-square random fields involved in the mixture, as well as the respective weights, have to be specified in the arguments k_vec and weights. If the distribution of the random field is not available in type, the user can specify in ECdensities a function taking c as argument and returning the vector of the desired EC densities to be evaluated at c. Notice that the length of the vector returned by the function specified in ECdensities must corresponds to one plus the dimension of the search area (since the first value should correspond to the EC density of order zero (see ECden_vec)).

Value

max_gLRT Maximum observed of the LRT field.

theta_max Value of THETA at which the maximum is observed.

global_p Global p-value.

MCerror Monte Carlo error associated to the global p-value.

Author(s)

Sara Algeri
TOHM_LRT

References


See Also

find_max, global_p, EC_T.

Examples

# generating data of interest
N<-100
x<-as.matrix(cbind(runif(N*2,172.5,217.5),runif(N*2,-2,58)))
x2<-x[(x[,1]<=217.5)&(x[,1]>=172.5),]
x_sel<-x2[(x2[,2]<=28+sqrt(30^2-(x2[,1]-195)^2))&(x2[,2]>=28-
    sqrt(30^2-(x2[,1]-195)^2))],]
data<-x_sel[sample(seq(1:(dim(x_sel)[1])),N),]

# specifying minus-log-likelihood
kg<-function(theta){integrate(Vectorize(function(x) {
    exp(-0.5*((x-theta[1])/0.5)^2)*integrate(function(y) {
        exp(-0.5*((y-theta[2])/0.5)^2) }, 28-sqrt(30^2-(x-195)^2),
        28+sqrt(30^2-(x-195)^2))$value}) , 172.5, 217.5)$value}
mll<-function(eta,x,theta){
    -sum(log((1-eta)/(pi*(30)^2)+eta*exp(-0.5*((x[,1]-
        theta[1])/0.5)^2-
        0.5*((x[,2]-theta[2])/0.5)^2)/kg(theta)))
}

# specifying search region
theta1<-seq(172.5,217.5,by=15)
theta2<-seq(-2,58,by=10)
THETA<-as.matrix(expand.grid(theta1,theta2))
originalR<-dim(THETA)[1]
rownames(THETA)<-1:(dim(THETA)[1])
THETA2<-THETA[(THETA[,1]<=217.5.5)&(THETA[,1]>=172.5),]
THETA_sel<-THETA2[(THETA2[,2]<=28+sqrt(30^2-(THETA2[,1]-195)^2))&(THETA2[,2]>=28-
                        sqrt(30^2-(THETA2[,1]-195)^2))],]

# generating toy EC
ECs<-cbind(rpois(100,1.5),rpois(100,1))

TOHM_LRT(data,mll,null0=0,init=c(0.1),lowlim=c(0),uplim=c(1),
    THETA=THETA_sel,ck=c(1,8),type=c("Chi-bar^2"),
    k=0,k_vec=c(0,1),weights=c(0.5,0.5),
    ECdensities=NULL,ECs=ECs)
# Index

* Chi-squared random field
  - chi2_ECden, 3
  - Gauss_ECden, 3, 5, 9
  - global_p, 2, 9, 10, 15
  - gLRT, 6, 8, 9, 12, 14
  - TOHM (TOHM-package), 2
  - TOHM-package, 2
  - TOHM_LRT, 2, 11, 13, 13

* Euler characteristic densities
  - chi2_ECden, 3
  - ECden_vec, 4
  - Gauss_ECden, 9

* Euler characteristic
  - EC_LRT, 5
  - EC_T, 7

* Gaussian random field
  - Gauss_ECden, 9

* Generalized Likelihood Ratio Test
  - gLRT, 12

* Generic random field
  - EC_T, 7
  - global_p, 10

* Global p-value
  - global_p, 10

* Maximum LRT field
  - find_max, 8

* Non regularity
  - TOHM_LRT, 13

* Profile Likelihood
  - gLRT, 12

* Test statistics
  - find_max, 8

* Testing One HYpothesis Multiple times
  - TOHM_LRT, 13

* Testing One Hypothesis Multiple TImes
  - TOHM-package, 2

* generalized Likelihood Ratio random field
  - EC_LRT, 5

  chi2_ECden, 3, 5, 10
  
  EC_LRT, 5, 7
  
  EC_T, 2, 6, 7, 15
  
  ECden_vec, 3, 4, 10, 11, 14

  find_max, 8, 11, 13, 15