Package ‘MRS’

January 23, 2018

Type Package
Title Multi-Resolution Scanning for Cross-Sample Differences
Version 1.2.4
Date 2018-01-13
Author Jacopo Soriano and Li Ma
Maintainer Li Ma <li.ma@duke.edu>
Description An implementation of the MRS algorithm for comparison across distributions, as described in Jacopo Soriano, Li Ma (2016) <doi:10.1111/rssb.12180>.

The model is based on a nonparametric process taking the form of a Markov model that transitions between a “null” and an “alternative” state on a multi-resolution partition tree of the sample space. MRS effectively detects and characterizes a variety of underlying differences.

These differences can be visualized using several plotting functions.

License GPL (>= 3)
Imports Rcpp (>= 0.11.0), igraph
Suggests KernSmooth
LinkingTo Rcpp, RcppArmadillo
NeedsCompilation yes
Repository CRAN
Date/Publication 2018-01-23 09:10:10 UTC

R topics documented:

andova ................................................................. 2
mrs ................................................................. 3
plot1D ............................................................... 5
plot2D ............................................................... 6
plotTree ............................................................ 7
print.summary.mrs ................................................ 9
summary.mrs ....................................................... 9

Index 11
**Description**

This function executes the Multi Resolution Scanning algorithm to detect differences across the distributions of multiple groups having multiple replicates.

**Usage**

```r
andova(X, G, H, n_groups = length(unique(G)), n_subgroups = NULL,
        Omega = "default", K = 6, init_state = c(0.8, 0.2, 0), beta = 1,
        gamma = 0.07, delta = 0.4, eta = 0, alpha = 0.5,
        nu_vec = 10*(seq(-1, 4)), return_global_null = TRUE, return_tree = TRUE)
```

**Arguments**

- `X`: Matrix of the data. Each row represents an observation.
- `G`: Numeric vector of the group label of each observation. Labels are integers starting from 1.
- `H`: Numeric vector of the replicate label of each observation. Labels are integers starting from 1.
- `n_groups`: Number of groups.
- `n_subgroups`: Vector indicating the number of replicates for each group.
- `Omega`: Matrix defining the vertices of the sample space. The "default" option defines a hyperrectangle containing all the data points. Otherwise the user can define a matrix where each row represents a dimension, and the two columns contain the associated lower and upper limit.
- `K`: Depth of the tree. Default is `K = 6`, while the maximum is `K = 14`.
- `init_state`: Initial state of the hidden Markov process. The three states are `null`, `alternative` and `prune`, respectively.
- `beta`: Spatial clustering parameter of the transition probability matrix. Default is `beta = 1.0`.
- `gamma`: Parameter of the transition probability matrix. Default is `gamma = 0.07`.
- `delta`: Parameter of the transition probability matrix. Default is `delta = 0.4`.
- `eta`: Parameter of the transition probability matrix. Default is `eta = 0.0`.
- `alpha`: Pseudo-counts of the Beta random probability assignments.
- `nu_vec`: The support of the discrete uniform prior on nu.
- `return_global_null`: Boolean indicating whether to return the marginal posterior probability of the global null.
- `return_tree`: Boolean indicating whether to return the posterior representative tree.
Value

An mrs object.

References


Examples

```r
set.seed(1)
n = 1000
M = 5
class_1 = sample(M, n, prob = 1:5, replace=TRUE )
class_2 = sample(M, n, prob = 5:1, replace=TRUE )

Y_1 = rnorm(n, mean=class_1, sd = .2)
Y_2 = rnorm(n, mean=class_2, sd = .2)

X = matrix( c(Y_1, Y_2), ncol = 1)
G = c(rep(1,n),rep(2,n))
H = sample(3,2*n, replace = TRUE )

ans = andova(X, G, H)
ans$PostGlobNull
plot1D(ans)
```

---

**mrs**

*Multi Resolution Scanning*

Description

This function executes the Multi Resolution Scanning algorithm to detect differences across multiple distributions.

Usage

```r
mrs(X, G, n_groups = length(unique(G)), Omega = "default", K = 6,
init_state = NULL, beta = 1, gamma = 0.3, delta = NULL, eta = 0.3,
alpha = 0.5, return_global_null = TRUE, return_tree = TRUE,
min_n_node = 0)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Matrix of the data. Each row represents an observation.</td>
</tr>
<tr>
<td>G</td>
<td>Numeric vector of the group label of each observation. Labels are integers starting from 1.</td>
</tr>
<tr>
<td>n_groups</td>
<td>Number of groups.</td>
</tr>
</tbody>
</table>

---
Omega  
Matrix defining the vertices of the sample space. The "default" option defines a hyperrectangle containing all the data points. Otherwise the user can define a matrix where each row represents a dimension, and the two columns contain the associated lower and upper limits for each dimension.

K  
Depth of the tree. Default is $K = 6$, while the maximum is $K = 14$.

init_state  
Initial state of the hidden Markov process. The three states are null, alternative and prune, respectively.

beta  
Spatial clustering parameter of the transition probability matrix. Default is $\beta = 1$.

gamma  
Parameter of the transition probability matrix. Default is $\gamma = 0.3$.

delta  
Optional parameter of the transition probability matrix. Default is $\delta = \text{NULL}$.

eta  
Parameter of the transition probability matrix. Default is $\eta = 0.3$.

alpha  
Pseudo-counts of the Beta random probability assignments. Default is $\alpha = 0.5$.

return_global_null  
Boolean indicating whether to return the posterior probability of the global null hypothesis.

return_tree  
Boolean indicating whether to return the posterior representative tree.

min_n_node  
Node in the tree is returned if there are more than $\text{min}_n\text{node}$ data-points in it.

Value

An mrs object.

References


Examples

```r
set.seed(1)
n = 20
p = 2
X = matrix(c(runif(p*n/2), rbeta(p*n/2, 1, 4)), nrow=n, byrow=TRUE)
G = c(rep(1,n/2), rep(2,n/2))
ans = mrs(X=X, G=G)
```
plot1D

*Plot regions of the representative tree in 1D*

**Description**

This function visualizes the regions of the representative tree of the output of the `mrs` function. For each region the posterior probability of difference (PMAP) or the effect size is plotted.

**Usage**

```r
plot1D(ans, type = "prob", group = 1, dim = 1, regions = rep(1,
    length(ans$RepresentativeTree$Levels)), legend = FALSE, main = "default",
    abs = TRUE)
```

**Arguments**

- **ans**: An `mrs` object.
- **type**: What is represented at each node. The options are `type = c("eff", "prob")`. Default is `type = "prob"`.
- **group**: If `type = "eff"`, which group effect size is used. Default is `group = 1`.
- **dim**: If the data are multivariate, `dim` is the dimension plotted. Default is `dim = 1`.
- **regions**: Binary vector indicating the regions to plot. The default is to plot all regions.
- **legend**: Color legend for type. Default is `legend = FALSE`.
- **main**: Overall title for the plot.
- **abs**: If TRUE, plot the absolute value of the effect size. Only used when `type = "eff"`.

**References**


**Examples**

```r
set.seed(1)
p = 1
n1 = 200
n2 = 200
mu1 = matrix( c(0,10), nrow = 2, byrow = TRUE)
muz2 = mu1; mu2[2] = mu1[2] + .01
sigma = c(1,.1)

Z1 = sample(2, n1, replace=TRUE, prob=c(0.9, 0.1))
Z2 = sample(2, n2, replace=TRUE, prob=c(0.9, 0.1))
```
X1 = mu1[Z1] + matrix(rnorm(n1*p), ncol=p)*sigma[Z1]
X2 = mu2[Z2] + matrix(rnorm(n2*p), ncol=p)*sigma[Z1]
X = rbind(X1, X2)
G = c(rep(1, n1), rep(2, n2))
ans = mrs(X, G, K=10)
plot1D(ans, type = "prob")
plot1D(ans, type = "eff")

### plot2D

_plot regions of the representative tree in 2D_

#### Description

This function visualizes the regions of the representative tree of the output of the _mrs_ function.

#### Usage

```r
plot2D(ans, type = "prob", data.points = "all", background = "none",
        group = 1, dim = c(1, 2),
        levels = sort(unique(ans$RepresentativeTree$Levels)), regions = rep(1,
        length(ans$RepresentativeTree$Levels)), legend = FALSE, main = "default",
        abs = TRUE)
```

#### Arguments

- **ans**
  - An _mrs_ object.
- **type**
  - Different options on how to visualize the rectangular regions. The options are type = c("eff", "prob", "empty", "none"). Default is type = "prob".
- **data.points**
  - Different options on how to plot the data points. The options are data.points = c("all", "differential"). Default is data.points = "all".
- **background**
  - Different options on the background. The options are background = c("smeared", "none").
- **group**
  - If type = "eff", which group effect size is used. Default is group = 1.
- **dim**
  - If the data are multivariate, dim are the two dimensions plotted. Default is dim = c(1,2).
- **levels**
  - Vector with the level of the regions to plot. The default is to plot regions at all levels.
- **regions**
  - Binary vector indicating the regions to plot. The default is to plot all regions.
- **legend**
  - Color legend for type. Default is legend = FALSE.
- **main**
  - Overall title for the legend.
- **abs**
  - If TRUE, plot the absolute value of the effect size. Only used when type = "eff".
References


Examples

```r
set.seed(1)
p = 2
n1 = 200
n2 = 200
mu1 = matrix(c(9,9,0,4,-2,-10,3,6,6,-10), nrow = 5, byrow=TRUE)
mu2 = mu1; mu2[2,] = mu1[2,] + 1

Z1 = sample(5, n1, replace=TRUE)
Z2 = sample(5, n2, replace=TRUE)
X1 = mu1[Z1,] + matrix(rnorm(n1*p), ncol=p)
X2 = mu2[Z2,] + matrix(rnorm(n2*p), ncol=p)
X = rbind(X1, X2)
colnames(X) = c(1,2)
G = c(rep(1, n1), rep(2,n2))

ans = mrs(X, G, K=8)
plotTree(ans, type = "prob", legend = FALSE)
plotTree(ans, type = "none", data.points = "differential",
background = "smeared", levels = 4)
```

plotTree

Plot nodes of the representative tree

Description

This function visualizes the representative tree of the output of the mrs function. For each node of the representative tree, the posterior probability of difference (PMAP) or the effect size is plotted. Each node in the tree is associated to a region of the sample space. All non-terminal nodes have two children nodes obtained by partitioning the parent region with a dyadic cut along a given direction. The numbers under the vertices represent the cutting direction.

Usage

```r
plotTree(ans, type = "prob", group = 1, legend = FALSE, main = "",
node.size = 5, abs = TRUE)
```
Arguments

ans
A mrs object.

type
What is represented at each node. The options are type = c("eff", "prob").

group
If type = "eff", which group effect size is used.

legend
Color legend for type. Default is legend = FALSE.

main
Main title. Default is main = "".

node.size
Size of the nodes. Default is node.size = 5.

abs
If TRUE, plot the absolute value of the effect size. Only used when type = "eff".

Note

The package igraph is required.

References


Examples

```r
set.seed(1)
p = 2
n1 = 200
n2 = 200
mu1 = matrix(c(9,9,0,4,-2,-10,3,6,6,-10), nrow = 5, byrow=TRUE)
mu2 = mu1; mu2[2,] = mu1[2,] + 1

Z1 = sample(5, n1, replace=TRUE)
Z2 = sample(5, n2, replace=TRUE)
X1 = mu1[Z1,] + matrix(rnorm(n1*p), ncol=p)
X2 = mu2[Z2,] + matrix(rnorm(n2*p), ncol=p)
X = rbind(X1, X2)
colnames(X) = c(1,2)
G = c(rep(1, n1), rep(2, n2))

ans = mrs(X, G, K=8)
plotTree(ans, type = "prob", legend = TRUE)
```
print.summary.mrs

Description

This function print the summary the output of the mrs function. It provides the marginal prior and posterior of the null and the top regions of the representative tree.

Usage

## S3 method for class 'summary.mrs'
print(x, ...)

Arguments

x A summary.mrs object

... Additional print parameters.

References


Examples

set.seed(1)
n = 100
p = 2
X = matrix(c(runif(p*n/2),rbeta(p*n/2, 1, 4)), nrow=n, byrow=TRUE)
G = c(rep(1,n/2), rep(2,n/2))
x = mrs(X=X, G=G)
fit = summary(x, rho = 0.95, abs_eff = 1)
print(fit)
Usage

```r
## S3 method for class 'mrs'
summary(object, rho = 0.5, abs_eff = 0, sort_by = "eff",
         ...)```

Arguments

- `object`: A `mrs` object
- `rho`: Threshold for the posterior alternative probability. All regions with posterior alternative probability larger than `rho` are reported. Default is `rho = 0.5`.
- `abs_eff`: Threshold for the effect size. All regions with effect size larger than `abs_eff` in absolute value are reported. Default is `abs_eff = 0`.
- `sort_by`: Define in which order the regions are reported. The options are `sort_by = c("eff", "prob")` and the default is `sort_by = "eff"`.
- `...`: Additional summary parameters.

Value

A list with information about the top regions.

References


Examples

```r
set.seed(1)
n = 100
p = 2
X = matrix(c(runif(p*n/2), rbeta(p*n/2, 1, 4)), nrow=n, byrow=TRUE)
G = c(rep(1,n/2), rep(2,n/2))
object = mrs(X=X, G=G)
fit = summary(object, rho = 0.5, abs_eff = 0.1)
```
Index

andova, 2

mrs, 3, 5–7, 9

plot1D, 5
plot2D, 6
plotTree, 7
print.summary.mrs, 9
summary.mrs, 9