Package‘DVDtest’

March 25, 2019

Type Package

Title Difference between Varying Distributions Test (DVDtest)

Version 0.1

Date 2019-03-24

Author Meng Xu, Philip Reiss

Maintainer Meng Xu <mxu@campus.haifa.ac.il>

Description See DVDtest.

RoxygenNote 6.1.1

License GPL (>= 2)

Encoding UTF-8

Imports gamlss, mgcv, ggplot2, reshape2, parallel, gamlss.dist

R topics documented:

`DVDtest-package` *Difference between Varying Distributions Test (DVDtest)*

Description

This package contains functions of a test on the difference between varying distributions.

Author(s)

Meng Xu, Philip Reiss

References

reiss-EMR18.pdf

See Also

`DVDtest`

`DVDtest` *Difference between Varying Distributions Test (DVDtest)*

Description

Testing the difference of two varying distributions.

Usage

```
DVDtest(ydata1, ydata2, nperm, eval.index.grid, dist.method = "wass",
  mgcv.gam = TRUE, ..., exclude = NULL, permadj = FALSE,
  mc.cores = 1)
```

Arguments

`ydata1` a `data.frame` or a `list` of `data.frame` containing at least 3 columns called ’`.obs`’,’`.index`’,’`.value`’ which specify which curve the point belongs to (`.obs`) at which (’`.index`’) it was observed and the observed value (`'.value'`). Other variables are available for modelling the varying distribution as well.

`ydata2` same type as `ydata1`. If the type of `ydata1` and `ydata2` is a `list` of `data.frame`, the lenghs of two lists must be the same.

`nperm` a scalar, number of permutation

`eval.index.grid` a vector, evaluation grid of `.index`

`dist.method` the methods, Wasserstein(`'wass'`), L2(`L2`), L1(`'L1'`) and Hellinger(`'Hellinger'`) to calculate the distances between distributions. Defaults to `'wass'`.

`mgcv.gam` a logical variable, whether to apply `mgcv::gam` for eastimating distributions, whose parameters are a smooth function of a continuous variable. If `FALSE`, `gamlss::gamlss` is adopted.

`...` If `mgcv.gam = TRUE`, `...` should include `formula` and `family(=gaulss())` and other arguments in `mgcv::gam`. Otherwise, the arguments in gamlss::gamlss are included.

`exclude` works only in the case of `mgcv.gam = TRUE`, to exclude the random effect

`permadj` a logical variable, whether to adjust the permutated data to cover the entire range, esp. working in case of sparsity. Defaults to `FALSE`

`mc.cores` the same argument in `mclapply` (not work on Windows). Defaults to 1.

Details

This is the Details section

Value

.indexa vector, evaluation grid

Note

- If `mgcv.gam` is TRUE and `exclue` is NULL (default settings), then `formula. <- list(.value~s(.index)+s(.obs` and `exclude. <- "s(.obs)"`

Author(s)

Meng Xu, Philip Reiss

References

reiss-EMR18.pdf

Examples

```
p=6
mu1<-function(t) 0.2*(p-1)*sin(pi*t)+t+1
mu2<-function(t) -0.2*(p-1)*sin(pi*t)+t+1
sig1 <- function(t) t+1
sig2 <- sig1
nperson=10
fun1<-function(t) rnorm(nperson,mu1(t),sig1(t))
fun2<-function(t) rnorm(nperson,mu2(t),sig2(t))
tp<-seq(0,1,,10)
data1<-sapply(tp,fun1)
data2<-sapply(tp,fun2)

library(reshape2)
colnames(data1)<-tp
dg1<-melt(data1)
colnames(dg1)<-c('.obs','.index','.value')
dg1$.obs<-as.factor(dg1$.obs)

colnames(data2)<-tp
rownames(data2)<-1:nperson+2*nperson
dg2<-melt(data2)
colnames(dg2)<-c('.obs','.index','.value')
dg2$.obs<-as.factor(dg2$.obs)
```

```
# library(ggplot2)
# ggplot()+geom_line(data=dg1,aes(x=.index,y=.value,col=factor(.obs)))+
#   geom_line(data=dg2,aes(x=.index,y=.value,col=factor(.obs)))

ngrid=50
ev.grid <- seq(0, 1, , ngrid)
nperm. <- 50

simu.test<-DVDtest(dg1,dg2,nperm.,ev.grid)

#ggplot(data.frame(simu.test),aes(x=.index,y=pval))+geom_line()+
# geom_hline(yintercept=0.05,linetype=2,col="red")
```

get.params *Fitting the generalized gamma (Pearson type III) distribution for each k*

Description

Fitting by the permuted-data distances

Usage

```
get.params(k, nperm, permarray, eval.index.grid)
```

Arguments

k	a scalar, kth data.frame in ydata1&2
nperm	a scalar, number of permutation
permarray	an array, permuted-data distances from wass_perm
eval.index.grid	a vector, evaluation grid of .index

Value

mu	a vector of mean
,	
sigma	a vector of sigma
,	
nu	a vector of nu
,	
aic	the Akaike information criterion in gamlss

Author(s)

Philip Reiss, Meng Xu

See Also

get_params

get.pval *Calculate the corrected P values*

Description

Calculate the corrected P values

Usage

```
get.pval(permarray, param.array, realdists, nroi, eval.index.grid, nperm)
```

Arguments

permarray	an array, permuted-data distances from wass_perm
param.array	an array, permuted-data distances from get_params
realdists	a matrix or vector, the value from get_realdist
nroi	a scalar, the length of ydata1
eval.index.grid	a vector, evaluation grid of .index
nperm	a scalar, number of permutation

Value

a vector or matrix of p value

Author(s)

Philip Reiss, Meng Xu

See Also

DVDtest

get.realdist *Calculating the distances under the null hypothesis for each roi*

Description

Calculating the distances under the null hypothesis for each roi

Usage

```
get.realdist(k, vdFun, ydata1, ydata2, ind.grid, ..., excl, dist.method)
```

Arguments

`k`	a scalar, kth data.frame in `ydata1&2`
`vdFun`	a function, `gam` or `gamlss`, for fitting the varying distributions
`ydata1`	see `DVDtest`
`ydata2`	see `DVDtest`
`ind.grid`	see `eval.index.grid` in `DVDtest`
`...`	arguments of `vdFun`
`excl`	an argument of `predict` for `gam`
`dist.method`	see `DVDtest`

Value

a vector of the distances

Author(s)

Philip Reiss, Meng Xu

See Also

`DVDtest`

`get_params` *Fitting the generalized gamma (Pearson type III) distribution*

Description

Fitting the generalized gamma (Pearson type III) distribution

Usage

```
get_params(nroi, nperm, permarray, eval.index.grid)
```

Arguments

`nroi`	a scalar, the length of `ydata1` or `ydata2`
`nperm`	a scalar, number of permutation
`permarray`	an array, permuted-data distances from `wass_perm`
`eval.index.grid`	a vector, evaluation grid of `.index`

Value

an array, `param.array`

Author(s)

Meng Xu, Philip Reiss

See Also

`DVDtest`

get_realdist *Calculating the distances under the null hypothesis*

Description

Calculating the distances under the null hypothesis

Usage

```
get_realdist(vdFun, ydata1, ydata2, ind.grid, ..., excl, mc.cores,
  dist.method)
```

Arguments

vdFun	a function, gam or gamlss, for fitting the varying distributions
ydata1	see DVDtest
ydata2	see DVDtest
ind.grid	see eval.index.grid in DVDtest
...	arguments of vdFun
excl	an argument of predict for gam
mc.cores	a scalar, an argument in mclapply
dist.method	see DVDtest

Value

a vector or matrix of the distances

Author(s)

Meng Xu, Philip Reiss

See Also

DVDtest

make.perms *Making permutated index*

Description

Making permutated index

Usage

```
make.perms(dat1, dat2, nperm, .index, adj)
```

Arguments

`dat1`	an element of `ydata1`
`dat2`	an element of `ydata2`
`nperm`	a scalar, number of permutation
`.index`	see `eval.index.grid` in `DVDtest`
`adj`	see `permadj` in `DVDtest`

Value

a matrix, permuted indices

Author(s)

Philip Reiss, Meng Xu

See Also

`DVDtest`

`multiwass` *Calculating the distances between two gam/gamlss objects*

Description

Calculating the distances between two gam/gamlss objects

Usage

```
multiwass(obj1, obj2, newdata1, newdata2, dist.method, ...)
```

Arguments

`obj1`	a gam/gamlss object
`obj2`	another gam/gamlss object
`newdata1`	related evaluation grids
`newdata2`	related evaluation grids
`dist.method`	see `DVDtest`
`...`	partial arguments in `predict`

Value

a vector, distances

Author(s)

Philip Reiss, Meng Xu

params2qfunc *Getting the quantile/density function via parameters*

Description

Getting the quantile/density function via parameters

Usage

```
params2qfunc(params, family, dist.method)
```

Arguments

`params`	a vector, parameters for certain distributions
`family`	a specific distribution from `gam` or `gamlss`
`dist.method`	see `DVDtest`

Value

quantile or density functions

Author(s)

Meng Xu, Philip Reiss

qfuncs2wass2 *Distance functions*

Description

Distance functions

Usage

```
qfuncs2wass2(qfunc1, qfunc2, dist.method = dist.method, ...)
```

Arguments

`qfunc1`	quantile or density functions from `params2qfunc`
`qfunc2`	quantile or density functions from `params2qfunc`
`dist.method`	see `DVDtest`
`...`	extra arguments in `integrate`

Value

distance functions

Author(s)

Meng Xu, Philip Reiss

wass.perm *Calculating the distances via permuted data for each* k

Description

Calculating the distances via permuted data for each k

Usage

```
wass.perm(k, vdFun, dat1, dat2, ..., permat, .index, report.every = 10,
  exclude, dist.method)
```

Arguments

k	a scalar, kth data.frame of ydata1&2
vdFun	a function, gam or gamlss, for fitting the varying distributions
dat1	kth data.frame of ydata1
dat2	kth data.frame of ydata2
...	arguments of vdFun
permat	a result of make.perm
.index	see eval.index.grid in DVDtest
report.every	a scalar, reporting the number permutation
exclude	an argument of predict
dist.method	see DVDtest

Value

a matrix of permuted-data distances

Author(s)

Philip Reiss, Meng Xu

See Also

wass_perm

wass_perm *Calculating the distances via permuted data*

Description

Calculating the distances via permuted data

Usage

```
wass_perm(vdFun, nperm, ydata1, ydata2, eval.index.grid, permat, ...,
  exclude, mc.cores, dist.method)
```

Arguments

vdFun	a function, gam or gamlss, for fitting the varying distributions
nperm	a scalar, number of permutation
ydata1	see DVDtest
ydata2	see DVDtest
eval.index.grid	see DVDtest
permat	a result of make.perm
...	partial arguments of vdFun
exclude	an argument of predict
mc.cores	a scalar, an argument of mclapply
dist.method	see DVDtest

Value

an array of distances

Author(s)

Meng Xu, Philip Reiss

See Also

wass.perm

Index