Package 'dgdecomp'

January 10, 2020

Type Package

Title Das Gupta Decomposition on Multiplicative Factors

Version 1.1.0

Description Takes any P number of factors, and decomposes them into additive factors.

Encoding UTF-8

LazyData TRUE

URL https://github.com/sadatnfs/dgdecomp

BugReports https://github.com/sadatnfs/dgdecomp/issues

NeedsCompilation yes

Author Nafis Sadat [aut]

Maintainer Nafis Sadat <sadatnfs@gmail.com>

LinkingTo Rcpp, RcppArmadillo

Depends data.table, Rcpp, RcppArmadillo

Suggests assertthat, testthat, foreach, data.table, matrixStats, MASS, knitr, rmarkdown

VignetteBuilder knitr

License MIT + file LICENSE

RoxygenNote 6.1.1

RemoteType github

RemoteHost api.github.com

RemoteRepo dgdecomp

RemoteUsername sadatnfs

RemoteRef master

RemoteSha 545f88d3b3677b192c3fca1b5a4cf0c39d2bef9a

RemoteSubdir dgdecomp

GithubRepo dgdecomp

GithubUsername sadatnfs

GithubRef master

GithubSHA1 545f88d3b3677b192c3fca1b5a4cf0c39d2bef9a

GithubSubdir dgdecomp

Archs i386, x64

R topics documented:

`dgdecomp-package` *Das Gupta Decomposition on Multiplicative Factors*

Description

Takes any P number of factors, and decomposes them into additive factors.

Details

The DESCRIPTION file:

Package:	dgdecomp
Type:	Package
Title:	Das Gupta Decomposition on Multiplicative Factors
Version:	1.1.0
Authors@R:	person("Nafis", "Sadat", email = "sadatnfs@gmail.com", role = c("aut", "cre"))
Description:	Takes any P number of factors, and decomposes them into additive factors.
Encoding:	UTF-8
LazyData:	TRUE
URL:	https://github.com/sadatnfs/dgdecomp
BugReports:	https://github.com/sadatnfs/dgdecomp/issues
NeedsCompilation:	yes
Author:	Nafis Sadat [aut]
Maintainer:	Nafis Sadat <sadatnfs@gmail.com>
LinkingTo:	Rcpp, RcppArmadillo
Depends:	data.table, Rcpp, RcppArmadillo
Suggests:	assertthat, testthat, foreach, data.table, matrixStats, MASS, knitr, rmarkdown
VignetteBuilder:	knitr
License:	MIT + file LICENSE
RoxygenNote:	6.1.1
RemoteType:	github
RemoteHost:	api.github.com
RemoteRepo:	dgdecomp
RemoteUsername:	sadatnfs
RemoteRef:	master
RemoteSha:	545f88d3b3677b192c3fca1b5a4cf0c39d2bef9a

RemoteSubdir:	dgdecomp
GithubRepo:	dgdecomp
GithubUsername:	sadatnfs
GithubRef:	master
GithubSHA1:	545f88d3b3677b192c3fca1b5a4cf0c39d2bef9a
GithubSubdir:	dgdecomp

Index of help topics:

This section should provide a more detailed overview of how to use the package, including the most important functions.

Author(s)

Nafis Sadat [aut]

Maintainer: Nafis Sadat <sadatnfs@gmail.com>

References

This optional section can contain literature or other references for background information.

See Also

Optional links to other man pages

Examples

```
## Optional simple examples of the most important functions
## Use \dontrun{} around code to be shown but not executed
```

`Decomp_Factors` *Compute the marginal decomposition effects from given input*

Description

`Decomp_Factor_Matx()` can be used if the input has multiple rows of data to decompose, whereas `Decomp_Factor()` only takes in vectors as inputs.

Usage

```
Decomp_Factors(vec_x, vec_y, return_dt = TRUE, equality_check = TRUE,
  ...)

Decomp_Factors_Matx(mat_x, mat_y, return_dt = TRUE, use_cpp = TRUE,
  parallel = 1, equality_check = TRUE, ...)
```

Arguments

`vec_x`	First input vector (represents t-1)
`vec_y`	Second input vector (represents t)
`return_dt`	A boolean on whether to return a data.table or a vector
`equality_check`	Check whether the decomp values align with inputs. Default: TRUE. Highly recommended to set to TRUE.
`...`	extra parameters to be passed to `all.equal()`, for e.g. the tolerance.
`mat_x`	First input matrix (only for `Decomp_Factor_Matx()`)
`mat_y`	Second input matrix (only for `Decomp_Factor_Matx()`)
`use_cpp`	A boolean on whether to use the C++ compiled code or not. Default: TRUE
`parallel`	Number of threads. Default : 1

Value

A data.table or vector of decomposed effects for each factors, which is already multiplied by the change values

Decomp_on_DT *Apply DG Decomposition to data.table columns*

Description

Apply DG Decomposition to data.table columns

Usage

```
Decomp_on_DT(input_data, factor_names, time_col, bycol, use_cpp = TRUE,
  parallel = 1, ...)
```

Arguments

input_data	A data.table with the factors, which must already be sorted
factor_names	A vector of column names for the factor
time_col	A string for the column name
bycol	The 'by' slicer which must make sure that the data is reduced to just 2 rows per group after slicing
use_cpp	A boolean on whether to use the C++ compiled code for the factor for-loop or not (passes to Decomp_Factor_Matx()). Default: TRUE
parallel	Number of threads. Default : 1
...	extra parameters to be passed through Decomp_Factors() to all.equal()

Value

A data.table of the same size as input, but instead with the additive decomposition results (first row will be NA as being the starting period)

Func_Create_Combn *Compute the combination of all positions of the given factor segmented into two pieces*

Description

Compute the combination of all positions of the given factor segmented into two pieces

Usage

```
Func_Create_Combn(Pfac, size1, size2)
```

Arguments

Pfac	Number of factors minus 1
size1	Number of elements to take from vec_x
size2	Number of elements to take from vec_y

Value

A vector of positions made from the unique combinations of size1 and size2

Func_Cross — *Compute the combination of all the elements of the given vectors corresponding to the given sizes (using Func_Create_Combn)*

Description

Compute the combination of all the elements of the given vectors corresponding to the given sizes (using Func_Create_Combn)

Usage

```
Func_Cross(vec_x, vec_y, size1, size2)

Func_Cross_Matx(vec_x, vec_y, size1, size2)
```

Arguments

vec_x	First input vector
vec_y	Second input vector
size1	Number of elements to take from vec_x
size2	Number of elements to take from vec_y

Value

A vector of column products made from the unique combinations of the *data*

Func_Dem — *Compute the denomiator of the inner sums in the DG Decomposition*

Description

Compute the denomiator of the inner sums in the DG Decomposition

Usage

```
Func_Dem(P, r)
```

Arguments

P	Number of factors
r	The summing indicator

Value

A numeric with value of `P * choose((P -1),(r -1))`

Func_Inner — *Compute the fraction for the inner sum in the DG Decomposition where all the params gets passed to Func_Num() and Func_Den()*

Description

Compute the fraction for the inner sum in the DG Decomposition where all the params gets passed to Func_Num() and Func_Den()

Usage

```
Func_Inner(P, r, vec_x, vec_y)

Func_Inner_Matx(P, r, vec_x, vec_y)
```

Arguments

P	Number of factors
r	The summing indicator
vec_x	First input vector
vec_y	Second input vector

Value

The fraction of the results of Func_Num() and Func_Den

Func_Inner_Sum — *Sum over all inner sums for the DG Decomposition*

Description

Sum over all inner sums for the DG Decomposition

Usage

```
Func_Inner_Sum(P, vec_x, vec_y)

Func_Inner_Sum_Matx(P, vec_x, vec_y)
```

Arguments

P	Number of factors
vec_x	First input vector
vec_y	Second input vector
threads	Number of OpenMP threads to use. Default: 1

Value

A numeric value with the full inner sum for the given effect

Func_Num — *Compute the numerator of the inner sums in the DG Decomposition*

Description

Compute the numerator of the inner sums in the DG Decomposition

Usage

```
Func_Num(P, r, vec_x, vec_y)

Func_Num_Matx(P, r, vec_x, vec_y)
```

Arguments

P	Number of factors
r	The summing indicator
vec_x	First input vector
vec_y	Second input vector

Value

A single numeric from the sums of Func_Cross()

simulate_decomp_data — *Simulate simple random decomp data for P factors such that the product of P factors equal a measure for 2 time periods*

Description

Simulate simple random decomp data for P factors such that the product of P factors equal a measure for 2 time periods

Usage

```
simulate_decomp_data(num_fac)
```

Arguments

num_fac	Number of factors to simulate

Value

A named list with the vector of P factors for 2 time periods, and 2 numeric measures for each time period, which are just the product of each of the two factor vectors

`simulate_decomp_data_fullmat` *Simulate grouped data for decomp analysis (T by IDI by factors)*

Description

Simulate grouped data for decomp analysis (T by IDI by factors)

Usage

```
simulate_decomp_data_fullmat(T_term, num_factors, id_grps)
```

Arguments

`T_term`	Number of time periods
`num_factors`	Number of factors (gets slower for large numbers)
`id_grps`	Number of groups

Value

A data.table with T_term*id_grps rows and num_factors+1 columns where X_1, ... , X_p are the factors, and Y is the rowwise product of the factors

Index