

Package ‘MDMR’

July 21, 2025

Type Package

Title Multivariate Distance Matrix Regression

Version 0.5.2

Date 2025-03-27

Description Allows users to conduct multivariate distance matrix regression using analytic p-values and compute measures of effect size. For details on the method, see McArtor, Lubke, & Bergeman (2017) <[doi:10.1007/s11336-016-9527-8](https://doi.org/10.1007/s11336-016-9527-8)>.

License GPL (>= 2)

LazyData TRUE

Depends R (>= 3.0.0), CompQuadForm, lme4,

Imports parallel, car

Suggests knitr, rmarkdown

VignetteBuilder knitr

URL <https://github.com/dmcartor/mdmr>

BugReports <https://github.com/dmcartor/mdmr/issues>

RoxygenNote 7.3.2

Encoding UTF-8

NeedsCompilation no

Author Dan McArtor [aut, cre] (dmcartor@gmail.com)

Maintainer Dan McArtor <dmcartor@gmail.com>

Repository CRAN

Date/Publication 2025-03-28 14:40:01 UTC

Contents

MDMR-package	2
.hat	3
.mdmr.permstats	4
.pmdmr	4

.pseudo.r2	5
delta	6
gower	9
mdmr	9
mixed.mdmr	12
print.mdmr	15
print.mixed.mdmr	15
summary.mdmr	16
summary.mixed.mdmr	17
X.clust	19
X.mdmr	19
Y.clust	20
Y.mdmr	20

Index	21
--------------	-----------

MDMR-package

Multivariate Distance Matrix Regression

Description

MDMR allows a user to conduct multivariate distance matrix regression using analytic p-values and measures of effect size described by McArtor et al. (2017). Analytic p-values are computed using the R package CompQuadForm (Duchesne & De Micheaux, 2010). It also facilitates the use of MDMR on samples consisting of (hierarchically) clustered observations.

Usage

To access this package's tutorial, type the following line into the console:

```
vignette('mdmr-vignette')
```

There are three primary functions that comprise this package: `mdmr`, which regresses a distance matrix onto a set of predictors, and `delta`, which computes measures of univariate effect size in the context of multivariate distance matrix regression. The third function `mixed.mdmr` facilitates the use of MDMR on (hierarchically) clustered samples using an approach analogous to the linear mixed-effects model for univariate outcomes. The help files of all all three functions provide more general information than the package vignette.

Author(s)

Maintainer: Dan McArtor <dmcartor@gmail.com> (dmcartor@gmail.com)

References

- Davies, R. B. (1980). The Distribution of a Linear Combination of chi-square Random Variables. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 29(3), 323-333.
- Duchesne, P., & De Micheaux, P.L. (2010). Computing the distribution of quadratic forms: Further comparisons between the Liu-Tang-Zhang approximation and exact methods. *Computational Statistics and Data Analysis*, 54(4), 858-862.

McArtor, D. B., Lubke, G. H., & Bergeman, C. S. (2017). Extending multivariate distance matrix regression with an effect size measure and the distribution of the test statistic. *Psychometrika*, 82, 1052-1077.

McArtor, D. B. (2017). Extending a distance-based approach to multivariate multiple regression (Doctoral Dissertation).

See Also

Useful links:

- <https://github.com/dmcartor/mdmr>
- Report bugs at <https://github.com/dmcartor/mdmr/issues>

Examples

```
#####
## Conducting MDMR on data comprised of independent observations
#####
# Source data
data(mdmrdata)

# Get distance matrix
D <- dist(Y.mdmr, method = 'euclidean')

# Conduct MDMR
mdmr.res <- mdmr(X = X.mdmr, D = D)
summary(mdmr.res)

#####
## Conducting MDMR on data comprised of dependent observations
#####
# Source data
data("clustmdmrdata")

# Get distance matrix
D <- dist(Y.clust)

# Conduct mixed-MDMR
mixed.res <- mixed.mdmr(~ x1 + x2 + (x1 + x2 | grp),
                        data = X.clust, D = D)
summary(mixed.res)
```

<code>.hat</code>	<i>Function to the hat matrix</i>
-------------------	-----------------------------------

Description

Function to the hat matrix

Usage

```
.hat(X)
```

Arguments

X A $n \times p$ matrix or data frame of predictors.

```
.mdmr.permstats            Function to compute test statistics for a permuted design matrix
```

Description

Function to compute test statistics for a permuted design matrix

Usage

```
.mdmr.permstats(X, n, vg, trG, px, test.inds)
```

Arguments

X A $n \times p$ matrix or data frame of predictors.
n sample size
vg Computational details (internal function)
trG trace of G
px Computational details (internal function)
test.inds Computational details (internal function)

```
.pmdmr                    Function to compute analytic p-values using Davies (1980)
```

Description

Compute analytic MDMR p-values

Usage

```
.pmdmr(q, lambda, k, p, n = length(lambda), lim = 50000, acc = 1e-20)
```

Arguments

q	Test statistic
lambda	Centered distance matrix eigenvalues
k	Numerator degrees of freedom for this test statistic
p	Total number of predictors in the design matrix (except for the intercept)
n	Sample size
lim	Maximum number of integration terms. Realistic values for "lim" range from 1,000 if the procedure is to be called repeatedly up to 50,000 if it is to be called only occasionally
acc	Error bound. Suitable values for "acc" range from 0.001 to 0.00005 which should be adequate for most statistical purposes.

Value

Output of `CompQuadForm::davies()`

Author(s)

Daniel B. McArtor (dmcartor@gmail.com) [aut, cre]

References

Davies, R. B. (1980). The Distribution of a Linear Combination of chi-square Random Variables. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 29(3), 323-333.

Duchesne, P., & De Micheaux, P. L. (2010). Computing the distribution of quadratic forms: Further comparisons between the Liu-Tang-Zhang approximation and exact methods. *Computational Statistics and Data Analysis*, 54(4), 858-862.

McArtor, D. B., Lubke, G. H., & Bergeman, C. S. (2017). Extending multivariate distance matrix regression with an effect size measure and the distribution of the test statistic. *Psychometrika*, 82, 1052-1077.

.pseudo.r2

function to compute pseudo R-square

Description

function to compute pseudo R-square

Usage

`.pseudo.r2(G, vh, vhs, x.inds, unique.xnames = NULL)`

Arguments

G	Gower's centered similarity matrix computed from D. Either D or G must be passed to <code>mdmr</code> .
vh	Computational details (internal function)
vhs	Computational trick (internal function)
x.inds	Computational trick (internal function)
unique.xnames	Computational details (internal function)

delta	<i>Compute univariate MDMR effect sizes</i>
-------	---

Description

`delta` computes permutation-based effect sizes on individual items comprising the distance matrix outcome used in multivariate distance matrix regression. It returns the omnibus estimates of delta (i.e. effect size of the entire design matrix on each outcome) as well as estimates of each pair-wise effect size (i.e. the effect of each predictor on each outcome variable, conditional on the rest of the predictors).

Usage

```
delta(
  X,
  Y = NULL,
  dtype = NULL,
  niter = 10,
  x.inds = NULL,
  y.inds = NULL,
  G = NULL,
  G.list = NULL,
  ncores = 1,
  seed = NULL,
  plot.res = F,
  grayscale = F,
  cex = 1,
  y.las = 2
)
```

Arguments

X	A $n \times p$ matrix or data frame of predictors. Unordered factors will be tested with contrast-codes by default, and ordered factors will be tested with polynomial contrasts. For finer control of how categorical predictors are handled, or if higher-order effects are desired, the output from a call to <code>model.matrix()</code> can be supplied to this argument as well.
---	--

<code>Y</code>	Outcome data: $n \times q$ matrix of scores along the dependent variables.
<code>dtype</code>	Measure of dissimilarity that will be used by <code>dist</code> to compute the distance matrix based on <code>Y</code> . As is the case when calling <code>dist</code> directly, this must be one of "euclidean", "maximum", "manhattan", "canberra", "binary" or "minkowski", and unambiguous substring can be given.
<code>niter</code>	Number of times to permute each outcome item in the procedure to compute delta. The final result is the average of all <code>niter</code> iterations. Higher values of <code>niter</code> require more computation time, but result in more precise estimates.
<code>x.inds</code>	Vector indicating which columns of <code>X</code> should have their conditional effect sizes computed. Default value of <code>NULL</code> results in all effects being computed, and a value of \emptyset results in no conditional effects being computed, such that only the omnibus effect sizes will be reported.
<code>y.inds</code>	Vector indicating which columns of <code>Y</code> effect sizes should be computed on. Default value of <code>NULL</code> results in all columns being used.
<code>G</code>	Gower's centered similarity matrix computed from <code>D</code> . Either <code>D</code> or <code>G</code> must be passed to <code>mdmr()</code> .
<code>G.list</code>	List of length q where the i^{th} element contains the <code>G</code> matrix computed from distance a matrix that was computed on a version of <code>Y</code> where the i^{th} column has been randomly permuted.
<code>ncores</code>	Integer; if <code>ncores > 1</code> , the <code>parallel</code> package is used to speed computation. Note: Windows users must set <code>ncores = 1</code> because the <code>parallel</code> package relies on forking. See <code>mc.cores</code> in the <code>mclapply</code> function in the <code>parallel</code> package for more details.
<code>seed</code>	Integer; sets seed for the permutations of each variable comprising <code>Y</code> so that results can be replicated.
<code>plot.res</code>	Logical; Indicates whether or not a heat-map of the results should be plotted.
<code>grayscale</code>	Logical; Indicates whether or not the heat-map should be plotted in grayscale.
<code>cex</code>	Multiplier for <code>cex.axis</code> , <code>cex.lab</code> , <code>cex.main</code> , and <code>cex</code> that are passed to the plotted result.
<code>y.las</code>	Orientation of labels for the outcome items. Defaults to vertical (2). Value of 1 prints horizontal labels, and is only recommended if the multivariate outcome is comprised of few variables.

Details

See McArtor et al. (2017) for a detailed description of how delta is computed. Note that it is a relative measure of effect, quantifying which effects are strong (high values of delta) and weak (low values of delta) within a single analysis, but estimates of delta cannot be directly compared across different datasets.

There are two options for using this function. The first option is to specify the predictor matrix `X`, the outcome matrix `Y`, the distance type `dtype` (supported by "dist" in R), and number of iterations `niter`. This option conducts the permutation of each `Y`-item `niter` times (to average out random association in each permutation) and reports the median estimates of delta over the `niter` reps.

The second option is to specify `X`, `G`, and `G.list`, a list of `G` matrices where the permutation has already been done for each item comprising `Y`. The names of the elements in `G.list` should correspond to the names of the variables that were permuted. This option is implemented so that `delta` can be computed when MDMR is being used in conjunction with distance metrics not supported by `dist`.

Value

A data frame whose rows correspond to the omnibus effects and the effect of each individual predictor (conditional on the rest), and whose columns correspond to each outcome variable whose effect sizes are being quantified. If `plot.res = TRUE`, a heat-map is plotted of this data frame to easily identify the strongest effects. Note that the heatmap is partitioned into the omnibus effect (first row) and pair-wise effects (remaining rows), because otherwise the omnibus effect would dominate the heatmap.

Author(s)

Daniel B. McArtor (dmcartor@gmail.com) [aut, cre]

References

McArtor, D. B., Lubke, G. H., & Bergeman, C. S. (2017). Extending multivariate distance matrix regression with an effect size measure and the distribution of the test statistic. *Psychometrika*, 82, 1052-1077.

Examples

```
data(mdmrdata)
# --- Method 1 --- #
delta(X.mdmr, Y = Y.mdmr, dtype = "euclidean", niter = 1, seed = 12345)

# --- Method 2 --- #
D <- dist(Y.mdmr, method = "euclidean")
G <- gower(D)
q <- ncol(Y.mdmr)
G.list <- vector(mode = "list", length = q)
names(G.list) <- names(Y.mdmr)
for(i in 1:q) {
  Y.shuf <- Y.mdmr
  Y.shuf[,i] <- sample(Y.shuf[,i])
  G.list[[i]] <- gower(dist(Y.shuf, method = "euclidean"))
}
delta(X.mdmr, G = G, G.list = G.list)
```

gower	<i>Compute Gower's centered similarity matrix from a distance matrix</i>
-------	--

Description

Compute Gower's centered similarity matrix G , which is the matrix decomposed by the MDMR test statistic.

Usage

```
gower(d.mat)
```

Arguments

d.mat	Symmetric distance matrix (or R distance object) computed from the outcome data to be used in MDMR.
-------	---

Value

G Gower's centered dissimilarity matrix computed from D.

Author(s)

Daniel B. McArtor (dmcartor@gmail.com) [aut, cre]

References

Gower, J. C. (1966). Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika*, 53(3-4), 325-338.

mdmr	<i>Conduct MDMR with analytic p-values</i>
------	--

Description

mdmr (multivariate distance matrix regression) is used to regress a distance matrix onto a set of predictors. It returns the test statistic, pseudo R-square statistic, and analytic p-values for all predictors jointly and for each predictor individually, conditioned on the rest.

Usage

```
mdmr(
  X,
  D = NULL,
  G = NULL,
  lambda = NULL,
  return.lambda = F,
  start.acc = 1e-20,
  ncores = 1,
  perm.p = (nrow(as.matrix(X)) < 200),
  nperm = 500,
  seed = NULL
)
```

Arguments

X	A $n \times p$ matrix or data frame of predictors. Unordered factors will be tested with contrast-codes by default, and ordered factors will be tested with polynomial contrasts. For finer control of how categorical predictors are handled, or if higher-order effects are desired, the output from a call to <code>model.matrix()</code> can be supplied to this argument as well.
D	Distance matrix computed on the outcome data. Can be either a matrix or an R dist object. Either D or G must be passed to <code>mdmr()</code> .
G	Gower's centered similarity matrix computed from D. Either D or G must be passed to <code>mdmr</code> .
lambda	Optional argument: Eigenvalues of G. Eigendecomposition of large G matrices can be somewhat time consuming, and the theoretical p-values require the eigenvalues of G. If MDMR is to be conducted multiple times on one distance matrix, it is advised to conduct the eigendecomposition once and pass the eigenvalues to <code>mdmr()</code> directly each time.
return.lambda	Logical; indicates whether or not the eigenvalues of G should be returned, if calculated. Default is FALSE.
start.acc	Starting accuracy of the Davies (1980) algorithm implemented in the davies function in the <code>CompQuadForm</code> package (Duchesne & De Micheaux, 2010) that <code>mdmr()</code> uses to compute MDMR p-values.
ncores	Integer; if <code>ncores > 1</code> , the parallel package is used to speed computation. Note: Windows users must set <code>ncores = 1</code> because the <code>parallel</code> package relies on forking. See <code>mc.cores</code> in the mclapply function in the <code>parallel</code> package for more details.
perm.p	Logical: should permutation-based p-values be computed instead of analytic p-values? Default behavior is TRUE if $n < 200$ and FALSE otherwise because the analytic p-values depend on asymptotics. for $n > 200$ and "permutation" otherwise.
nperm	Number of permutations to use if permutation-based p-values are to be computed.
seed	Random seed to use to generate the permutation null distribution. Defaults to a random seed.

Details

This function is the fastest approach to conducting MDMR. It uses the fastest known computational strategy to compute the MDMR test statistic (see Appendix A of McArtor et al., 2017), and it uses fast, analytic p-values.

The slowest part of conducting MDMR is now the necessary eigendecomposition of the G matrix, whose computation time is a function of n^3 . If MDMR is to be conducted multiple times on the same distance matrix, it is recommended to compute eigenvalues of G in advance and pass them to the function rather than computing them every time `mdmr` is called, as is the case if the argument `lambda` is left NULL.

The distance matrix D can be passed to `mdmr` as either a distance object or a symmetric matrix.

Value

An object with six elements and a summary function. Calling `summary(mdmr.res)` produces a data frame comprised of:

Statistic	Value of the corresponding MDMR test statistic
Numer DF	Numerator degrees of freedom for the corresponding effect
Pseudo R2	Size of the corresponding effect on the distance matrix
p-value	The p-value for each effect.

In addition to the information in the three columns comprising `summary(res)`, the `res` object also contains:

<code>p.prec</code>	A data.frame reporting the precision of each p-value. If analytic p-values were computed, these are the maximum error bound of the p-values reported by the <code>davies</code> function in <code>CompQuadForm</code> . If permutation p-values were computed, it is the standard error of each permutation p-value.
<code>lambda</code>	A vector of the eigenvalues of G (if <code>return.lambda = T</code>).
<code>nperm</code>	Number of permutations used. Will read NA if analytic p-values were computed

Note that the printed output of `summary(res)` will truncate p-values to the smallest trustworthy values, but the object returned by `summary(res)` will contain the p-values as computed. The reason for this truncation differs for analytic and permutation p-values. For an analytic p-value, if the error bound of the Davies algorithm is larger than the p-value, the only conclusion that can be drawn with certainty is that the p-value is smaller than (or equal to) the error bound. For a permutation test, the estimated p-value will be zero if no permuted test statistics are greater than the observed statistic, but the zero p-value is only a product of the finite number of permutations conducted. The only conclusion that can be drawn is that the p-value is smaller than $1/nperm$.

Author(s)

Daniel B. McArtor (dmcartor@gmail.com) [aut, cre]

References

- Davies, R. B. (1980). The Distribution of a Linear Combination of chi-square Random Variables. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 29(3), 323-333.
- Duchesne, P., & De Micheaux, P. L. (2010). Computing the distribution of quadratic forms: Further comparisons between the Liu-Tang-Zhang approximation and exact methods. *Computational Statistics and Data Analysis*, 54(4), 858-862.
- McArtor, D. B., Lubke, G. H., & Bergeman, C. S. (2017). Extending multivariate distance matrix regression with an effect size measure and the distribution of the test statistic. *Psychometrika*, 82, 1052-1077.

Examples

```
# --- The following two approaches yield equivalent results --- #
# Approach 1
data(mdmrdata)
D <- dist(Y.mdmr, method = "euclidean")
res1 <- mdmr(X = X.mdmr, D = D)
summary(res1)

# Approach 2
data(mdmrdata)
D <- dist(Y.mdmr, method = "euclidean")
G <- gower(D)
res2 <- mdmr(X = X.mdmr, G = G)
summary(res2)
```

mixed.mdmr

Fit Mixed-MDMR models

Description

`mixed.mdmr` allows users to conduct multivariate distance matrix regression (MDMR) in the context of a (hierarchically) clustered sample without inflating Type-I error rates as a result of the violation of the independence assumption. This is done by invoking a mixed-effects modeling framework, in which clustering/grouping variables are specified as random effects and the covariate effects of interest are fixed effects. The input to `mixed.mdmr` largely reflects the input of the `lmer` function from the package `lme4` insofar as the specification of random and fixed effects are concerned (see Arguments for details). Note that this function simply controls for the random effects in order to test the fixed effects; it does not facilitate point estimation or inference on the random effects.

Usage

```
mixed.mdmr(
  fmla,
  data,
  D = NULL,
```

```

G = NULL,
use.ssd = 1,
start.acc = 1e-20,
ncores = 1
)

```

Arguments

fm1a	A one-sided linear formula object describing both the fixed-effects and random-effects part of the model, beginning with an <code>~</code> operator, which is followed by the terms to include in the model, separated by <code>+</code> operators. Random-effects terms are distinguished by vertical bars (<code> </code>) separating expressions for design matrices from grouping factors. Two vertical bars (<code> </code>) can be used to specify multiple uncorrelated random effects for the same grouping variable.
data	A mandatory data frame containing the variables named in formula.
D	Distance matrix computed on the outcome data. Can be either a matrix or an R dist object. Either D or G must be passed to <code>mdmr()</code> .
G	Gower's centered similarity matrix computed from D. Either D or G must be passed to <code>mdmr</code> .
use.ssd	The proportion of the total sum of squared distances (SSD) that will be targeted in the modeling process. In the case of non-Euclidean distances, specifying <code>use.ssd</code> to be slightly smaller than 1.00 (e.g., 0.99) can substantially lower the computational burden of <code>mixed.mdmr</code> while maintaining well-controlled Type-I error rates and only sacrificing a trivial amount of power. In the case of Euclidean distances the computational burden of <code>mixed.mdmr</code> is small, so <code>use.ssd</code> should be set to 1.00.
start.acc	Starting accuracy of the Davies (1980) algorithm implemented in the davies function in the <code>CompQuadForm</code> package (Duchesne & De Micheaux, 2010) that <code>mdmr()</code> uses to compute MDMR p-values.
ncores	Integer; if <code>ncores > 1</code> , the parallel package is used to speed computation. Note: Windows users must set <code>ncores = 1</code> because the <code>parallel</code> package relies on forking. See <code>mc.cores</code> in the mclapply function in the <code>parallel</code> package for more details.

Value

An object with six elements and a summary function. Calling `summary(mixed.mdmr.res)` produces a data frame comprised of:

Statistic	Value of the corresponding MDMR test statistic
Numer DF	Numerator degrees of freedom for the corresponding effect
p-value	The p-value for each effect.

In addition to the information in the three columns comprising `summary(res)`, the `res` object also contains:

`p.prec` A data.frame reporting the precision of each p-value. If analytic p-values were computed, these are the maximum error bound of the p-values reported by the `davies` function in `CompQuadForm`. If permutation p-values were computed, it is the standard error of each permutation p-value.

Note that the printed output of `summary(res)` will truncate p-values to the smallest trustworthy values, but the object returned by `summary(res)` will contain the p-values as computed. The reason for this truncation differs for analytic and permutation p-values. For an analytic p-value, if the error bound of the Davies algorithm is larger than the p-value, the only conclusion that can be drawn with certainty is that the p-value is smaller than (or equal to) the error bound.

Author(s)

Daniel B. McArtor (dmcartor@gmail.com) [aut, cre]

References

Davies, R. B. (1980). The Distribution of a Linear Combination of chi-square Random Variables. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 29(3), 323-333.

Duchesne, P., & De Micheaux, P. L. (2010). Computing the distribution of quadratic forms: Further comparisons between the Liu-Tang-Zhang approximation and exact methods. *Computational Statistics and Data Analysis*, 54(4), 858-862.

McArtor, D. B. (2017). Extending a distance-based approach to multivariate multiple regression (Doctoral Dissertation).

Examples

```
data("clustmdmrdata")

# Get distance matrix
D <- dist(Y.clust)

# Regular MDMR without the grouping variable
mdmr.res <- mdmr(X = X.clust[,1:2], D = D, perm.p = FALSE)

# Results look significant
summary(mdmr.res)

# Account for grouping variable
mixed.res <- mixed.mdmr(~ x1 + x2 + (x1 + x2 | grp),
                       data = X.clust, D = D)

# Significance was due to the grouping variable
summary(mixed.res)
```

print.mdmr	<i>Print MDMR Object</i>
------------	--------------------------

Description

print method for class mdmr

Usage

```
## S3 method for class 'mdmr'  
print(x, ...)
```

Arguments

x	Output from mdmr
...	Further arguments passed to or from other methods.

Value

p-value	Analytic p-values for the omnibus test and each predictor
---------	---

Author(s)

Daniel B. McArtor (dmcartor@gmail.com) [aut, cre]

print.mixed.mdmr	<i>Print Mixed MDMR Object</i>
------------------	--------------------------------

Description

print method for class mixed.mdmr

Usage

```
## S3 method for class 'mixed.mdmr'  
print(x, ...)
```

Arguments

x	Output from mixed.mdmr
...	Further arguments passed to or from other methods.

Value

p-value	Analytic p-values for the omnibus test and each predictor
---------	---

Author(s)

Daniel B. McArtor (dmcartor@gmail.com) [aut, cre]

summary.mdmr

Summarizing MDMR Results

Description

summary method for class mdmr

Usage

```
## S3 method for class 'mdmr'
summary(object, ...)
```

Arguments

object Output from mdmr
 ... Further arguments passed to or from other methods.

Value

Calling `summary(mdmr.res)` produces a data frame comprised of:

Statistic	Value of the corresponding MDMR test statistic
Pseudo R2	Size of the corresponding effect on the distance matrix
p-value	The p-value for each effect.

In addition to the information in the three columns comprising `summary(res)`, the `res` object also contains:

p.prec	A data.frame reporting the precision of each p-value. If analytic p-values were computed, these are the maximum error bound of the p-values reported by the <code>davies</code> function in <code>CompQuadForm</code> . If permutation p-values were computed, it is the standard error of each permutation p-value.
lambda	A vector of the eigenvalues of G (if <code>return.lambda = T</code>).
nperm	Number of permutations used. Will read NA if analytic p-values were computed

Note that the printed output of `summary(res)` will truncate p-values to the smallest trustworthy values, but the object returned by `summary(res)` will contain the p-values as computed. The reason for this truncation differs for analytic and permutation p-values. For an analytic p-value, if the error bound of the Davies algorithm is larger than the p-value, the only conclusion that can be drawn with certainty is that the p-value is smaller than (or equal to) the error bound. For a permutation test, the estimated p-value will be zero if no permuted test statistics are greater than the observed statistic, but the zero p-value is only a product of the finite number of permutations conducted. The only conclusion that can be drawn is that the p-value is smaller than $1/nperm$.

Author(s)

Daniel B. McArtor (dmcartor@gmail.com) [aut, cre]

References

Davies, R. B. (1980). The Distribution of a Linear Combination of chi-square Random Variables. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 29(3), 323-333.

Duchesne, P., & De Micheaux, P. L. (2010). Computing the distribution of quadratic forms: Further comparisons between the Liu-Tang-Zhang approximation and exact methods. *Computational Statistics and Data Analysis*, 54(4), 858-862.

McArtor, D. B., Lubke, G. H., & Bergeman, C. S. (2017). Extending multivariate distance matrix regression with an effect size measure and the distribution of the test statistic. *Psychometrika*, 82, 1052-1077.

Examples

```
# --- The following two approaches yield equivalent results --- #
# Approach 1
data(mdmrdata)
D <- dist(Y.mdmr, method = "euclidean")
mdmr.res <- mdmr(X = X.mdmr, D = D)
summary(mdmr.res)
```

summary.mixed.mdmr *Summarizing Mixed MDMR Results*

Description

summary method for class mixed.mdmr

Usage

```
## S3 method for class 'mixed.mdmr'
summary(object, ...)
```

Arguments

object Output from [mixed.mdmr](#)

... Further arguments passed to or from other methods.

Value

Calling `summary(mdmr.res)` produces a data frame comprised of:

Statistic	Value of the corresponding MDMR test statistic
p-value	The p-value for each effect.

In addition to the information in the three columns comprising `summary(res)`, the `res` object also contains:

p.prec	A data.frame reporting the precision of each p-value. If analytic p-values were computed, these are the maximum error bound of the p-values reported by the <code>davies</code> function in <code>CompQuadForm</code> . If permutation p-values were computed, it is the standard error of each permutation p-value.
--------	--

Note that the printed output of `summary(res)` will truncate p-values to the smallest trustworthy values, but the object returned by `summary(res)` will contain the p-values as computed. The reason for this truncation differs for analytic and permutation p-values. For an analytic p-value, if the error bound of the Davies algorithm is larger than the p-value, the only conclusion that can be drawn with certainty is that the p-value is smaller than (or equal to) the error bound.

Author(s)

Daniel B. McArtor (dmcartor@gmail.com) [aut, cre]

References

- Davies, R. B. (1980). The Distribution of a Linear Combination of chi-square Random Variables. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 29(3), 323-333.
- Duchesne, P., & De Micheaux, P. L. (2010). Computing the distribution of quadratic forms: Further comparisons between the Liu-Tang-Zhang approximation and exact methods. *Computational Statistics and Data Analysis*, 54(4), 858-862.
- McArtor, D. B. (2017). Extending a distance-based approach to multivariate multiple regression (Doctoral Dissertation).

Examples

```
data("clustmdmrdata")

# Get distance matrix
D <- dist(Y.clust)

# Regular MDMR without the grouping variable
mdmr.res <- mdmr(X = X.clust[,1:2], D = D, perm.p = FALSE)

# Results look significant
summary(mdmr.res)

# Account for grouping variable
mixed.res <- mixed.mdmr(~ x1 + x2 + (x1 + x2 | grp),
```

```
data = X.clust, D = D)

# Significance was due to the grouping variable
summary(mixed.res)
```

X.clust	<i>Simulated clustered predictor data to illustrate the Mixed-MDMR function</i>
---------	---

Description

See [mixed.mdmr](#).

Usage

```
X.clust
```

Format

An object of class `data.frame` with 250 rows and 3 columns.

X.mdmr	<i>Simulated predictor data to illustrate the MDMR package.</i>
--------	---

Description

See package vignette by calling `vignette("mdmr-vignette")`.

Usage

```
X.mdmr
```

Format

An object of class `matrix` (inherits from `array`) with 500 rows and 3 columns.

Y.clust	<i>Simulated clustered outcome data to illustrate the Mixed-MDMR function</i>
---------	---

Description

See [mixed.mdmr](#).

Usage

Y.clust

Format

An object of class `data.frame` with 250 rows and 12 columns.

Y.mdmr	<i>Simulated outcome data to illustrate the MDMR package.</i>
--------	---

Description

See package vignette by calling `vignette("mdmr-vignette")`.

Usage

Y.mdmr

Format

An object of class `matrix` (inherits from `array`) with 500 rows and 10 columns.

Index

* datasets

X.clust, 19

X.mdmr, 19

Y.clust, 20

Y.mdmr, 20

.hat, 3

.mdmr.permstats, 4

.pmdmr, 4

.pseudo.r2, 5

davies, 10, 13

delta, 2, 6

dist, 7, 10, 13

gower, 9

lme4, 12

lmer, 12

mclapply, 7, 10, 13

MDMR (MDMR-package), 2

mdmr, 2, 9

MDMR-package, 2

mixed.mdmr, 2, 12, 15, 17, 19, 20

parallel, 7, 10, 13

print.mdmr, 15

print.mixed.mdmr, 15

summary.mdmr, 16

summary.mixed.mdmr, 17

X.clust, 19

X.mdmr, 19

Y.clust, 20

Y.mdmr, 20