

# Package ‘miRtest’

February 20, 2015

**Maintainer** Stephan Artmann <stephanartmann@gmx.net>

**License** GPL

**Title** combined miRNA- and mRNA-testing

**URL** <http://www.ncbi.nlm.nih.gov/pubmed/22723856>

**Type** Package

**LazyLoad** yes

**Author** Stephan Artmann, Klaus Jung, Tim Beissbarth

**Description** combined miRNA- and mRNA-testing

**Version** 1.8

**Date** 2014-11-25

**Depends** corpcor, MASS

**Imports** globaltest, GlobalAncova, limma, RepeatedHighDim

**Collate** 'miRtest.R'

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2014-11-27 01:46:34

## R topics documented:

A	2
contingency.table	2
fisher.combination	3
generate.A	3
gs.test	5
inverse.normal.combination	7
limma.one.sided	7
limma.test	8
m.combine	8
miR.test	9
miRtest	12
X	14
Y	15

**Index****16**

---

**A****A**

---

**Description**

Part of expression data from Nielsen et al.

**Author(s)**

Stephan Artmann taken from Nielsen et al.

---

`contingency.table`*Contingency table.*

---

**Description**

Contingency table. Necessary for Fisher test.

**Usage**

```
contingency.table(gene.set, p.val, sign=0.05)
```

**Arguments**

<code>gene.set</code>	Vector of gene sets.
<code>p.val</code>	Vector with p-values.
<code>sign</code>	Significance threshold.

**Author(s)**

Stephan Artmann

---

fisher.combination      *Fisher method of p value combination.*

---

**Description**

Fisher method of p value combination.

**Usage**

```
fisher.combination(p1, p2, check.range=FALSE)
```

**Arguments**

p1,p2                  one-sided p-values that shall be combined.  
check.range          If set to "TRUE" values above 1 will be set to 1.

**Value**

Combined p-value.

**Author(s)**

Stephan Artmann

---

generate.A                  *Turn a data...*

---

**Description**

Turn a data.frame indicating gene sets into the allocation matrix.

**Usage**

```
generate.A(df, X, Y, verbose=TRUE)
```

**Arguments**

df                      data.frame with mRNAs in its first and miRNAs in its second column.  
X                      Expression matrix of miRNAs whose row names will be used to generate the list of miRNAs.  
Y                      Expression matrix of mRNAs whose row names will be used to generate the list of mRNAs.  
verbose                Logical. Shall progress be printed?

**Value**

Allocation matrix A necessary for "miR.test" function.

**Author(s)**

Stephan Artmann

**Examples**

```
#####
### Generate random expression data ###
#####
# Generate random miRNA expression data of 3 miRNAs
# with 8 replicates
set.seed(1)
X = rnorm(24);
dim(X) = c(3,8);
rownames(X) = 1:3;
# Generate random mRNA expression data with 20 mRNAs
# and 10 replicates
Y = rnorm(200);
dim(Y) = c(20,10);
rownames(Y) = 1:20;
# Let's assume that we want to compare 2 miRNA groups, each of 4 replicates:
group.miRNA = factor(c(1,1,1,1,2,2,2,2));
# ... and that the corresponding mRNA experiments had 5 replicates in each group
group.mRNA = factor(c(1,1,1,1,1,2,2,2,2));
#####
### Perform Test ###
#####
library(miRtest)
#Let miRNA 1 attack mRNAs 1 to 9 and miRNA 2 attack mRNAs 10 to 17.
# mRNAs 18 to 20 are not attacked. miRNA 3 has no gene set.
miR = c(rep(1,9),c(rep(2,8)));
mRNAs = 1:17;
A = data.frame(mRNAs,miR); # Note that the miRNAs MUST be in the second column!
A
set.seed(1)
P = miR.test(X,Y,A,group.miRNA,group.mRNA)
P

#####
### For a faster result: use other gene set tests ###
#####
# Wilcoxon two-sample test is recommended for fast results
# Note that results may vary depending on how much genes correlate

P.gsWilcox = miR.test(X,Y,A,group.miRNA,group.mRNA,gene.set.tests="W")
P.gsWilcox
#####
### We can use an allocation matrix as A ###
```

```
#####
A = generate.A(A,X=X,Y=Y,verbose=FALSE);
A
# Now we can test as before
set.seed(1)
P = miR.test(X,Y,A,group.miRNA,group.mRNA,allocation.matrix=TRUE)
P

#####
### Other Designs ###
#####

# Some more complicated designs are implemented, check the vignette "miRtest" for details.
group.miRNA = 1:8
group.mRNA = 1:10
covariable.miRNA = factor(c(1,2,3,4,1,2,3,4))   ### A covariable in miRNAs.
covariable.mRNA = factor(c(1,2,3,4,5,1,2,3,4,5)) ### A covariable in mRNAs.

library(limma)
design.miRNA = model.matrix(~group.miRNA + covariable.miRNA)
design.mRNA = model.matrix(~group.mRNA + covariable.mRNA)

P = miR.test(X,Y,A,design.miRNA=design.miRNA,design.mRNA=design.mRNA,allocation.matrix=TRUE)
P
```

---

gs.test

*Internal function for gene set testing.*


---

## Description

Internal function for gene set testing.

## Usage

```
gs.test(A, X, Y, group, tests, permutation=FALSE, nrot=1000, design,
        allocation.matrix=FALSE, verbose=FALSE)
```

## Arguments

A	Allocation matrix as in "miR.test" function.
X	miRNA expression matrix as in 'miR.test' function. Only necessary when allocation.matrix=TRUE.
Y	mRNA expression matrix as in "miR.test" function.
group	group as in 'miR.test' function
tests	Test applied, see gene.set.tests
permutation	Shall permutation procedure for global tests be applied? Put 'FALSE' to use approximate results or give a number for the number of permutations.

nrot	Number of rotations of rotation tests. Defaults to 1000 to be able to show p-values as low as $10^{-3}$ .
design	If specified, group will be ignored. Design matrix as used in 'limma' package. Cannot be used with global tests.
allocation.matrix	Logical, is A an allocation matrix with mRNAs in its columns and miRNAs in its rows, or is it an allocation data.frame?
verbose	Defaults to FALSE. If TRUE, progress is printed.

**Value**

List of the following, for up- and for down-regulation: Matrix with testing results for every gene set in its rows and the applied gene set test in its columns.

**Author(s)**

Stephan Artmann

**References**

- Artmann, Stephan and Jung, Klaus and Bleckmann, Annalen and Beissbarth, Tim (2012). Detection of simultaneous group effects in microRNA expression and related functional gene sets. *PLoS ONE* 7(6):e38365, PMID: 22723856.
- Brunner, E. (2009) Repeated measures under non-sphericity. *Proceedings of the 6th St. Petersburg Workshop on Simulation*, 605-609.
- Jelle J. Goeman, Sara A. van de Geer, Floor de Kort, Hans C. van Houwelingen (2004) A global test for groups of genes: testing association with a clinical outcome. *Bioinformatics* 20, 93-99.
- Jung, Klaus and Becker, Benjamin and Brunner, Edgar and Beissbarth, Tim (2011). Comparison of Global Tests for Functional Gene Sets in Two-Group Designs and Selection of Potentially Effect-causing Genes. *Bioinformatics*, 27: 1377-1383.
- Majewski, IJ, Ritchie, ME, Phipson, B, Corbin, J, Pakusch, M, Ebert, A, Busslinger, M, Koseki, H, Hu, Y, Smyth, GK, Alexander, WS, Hilton, DJ, and Blewitt, ME (2010). Opposing roles of polycomb repressive complexes in hematopoietic stem and progenitor cells. *\_Blood\_*, published online 5 May 2010.
- Mansmann, U. and Meister, R., 2005, Testing differential gene expression in functional groups, *\_Methods Inf Med\_* 44 (3).
- Smyth, G. K. (2004). Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *\_Statistical Applications in Genetics and Molecular Biology\_*, Volume \*3\*, Article 3.
- Wu, D, Lim, E, Francois Vaillant, F, Asselin-Labat, M-L, Visvader, JE, and Smyth, GK (2010). ROAST: rotation gene set tests for complex microarray experiments. *\_Bioinformatics\_*, published online 7 July 2010.

---

`inverse.normal.combination`

*Inverse-normal method for p value combination.*

---

### **Description**

Inverse-normal method for p value combination.

### **Usage**

```
inverse.normal.combination(p1, p2)
```

### **Arguments**

`p1,p2`            one-sided p-values that shall be combined.

### **Value**

Two-sided combined p-value.

### **Author(s)**

Stephan Artmann

---

`limma.one.sided`

*Internal algorithm: Make limma test one-sided...*

---

### **Description**

Internal algorithm: Make limma test one-sided

### **Usage**

```
limma.one.sided(fit, lower=FALSE)
```

### **Arguments**

`fit`                Result of "lmFit" and "eBayes" functions in "limma" package.

`lower`             Shall one-sided p-value indicated down-regulation?

limma.test                    *internal algorithm for author's convenience.*

---

**Description**

internal algorithm for author's convenience. Create a linear model with the limma package.

**Usage**

```
limma.test(X, group, design)
```

**Arguments**

X	Expression matrix.
group	Group membership of replicates.
design	Design as specified in limma (design matrix, see model.matrix).

**Author(s)**

Stephan Artmann

---

m.combine                    *Internal function for author's convenience and more legible code.*

---

**Description**

Internal function for author's convenience and more legible code. Applies a function to every column vector of a matrix and a vector.

**Usage**

```
m.combine(M, v, FUN, ...)
```

**Arguments**

M	The matrix for whose column vectors mapply shall be used.
v	The vector.
FUN	The function.
...	Further arguments to be given to FUN.

**Author(s)**

Stephan Artmann



miR.test

*Main Function of miRtest package.***Description**

Main Function of miRtest package.

**Usage**

```
miR.test(X, Y, A, group.miRNA, group.mRNA, gene.set.tests="romer",
         design.miRNA, design.mRNA, adjust="none", permutation=FALSE,
         nrot=1000, allocation.matrix=FALSE, verbose=FALSE, errors=TRUE)
```

**Arguments**

X	miRNA expression matrix with genes in rows and replicates in columns
Y	mRNA expression matrix with genes in rows and replicates in columns
A	Allocation data.frame or Allocation matrix. An allocation data.frame contains the mRNAs in its first column and the miRNAs in its second column. See vignette 'miRtest' for information on Allocation matrices.
group.miRNA	Vector of miRNA group membership, being either numeric or a factor (**this makes a difference**). E. g. if you have four replicates in a control group and three replicates in a treated group, you may choose c(1,1,1,1,2,2,2)
design.miRNA	If specified, group.miRNA will be ignored. Here you can specify a design matrix as it is returned from the model.matrix 'limma' function.
design.mRNA	If specified, group.mRNA will be ignored. Here you can specify a design matrix as it is returned from the model.matrix 'limma' function.
group.mRNA	Vector of mRNA group membership, being either numeric or a factor (**this makes a difference**).E. g. if you have four replicates in a control group and three replicates in a treated group, you may choose c(1,1,1,1,2,2,2)
gene.set.tests	Test to be applied for gene set testing. Can be one or more of the following: 'globaltest', 'GA', 'RHD', 'KS', 'W', 'Fisher', 'roast', 'romer', or 'all' if you want to do all tests.
adjust	Multiple hypothesis testing adjustment. Same options as in "p.adjust" function.
permutation	Number of permutations for 'globaltest' or 'GlobalAncova' gene set tests. Put to "FALSE" to use the approximate p-values instead of permutation ones.
nrot	Number of rotations for rotation tests 'ROAST' and 'romer'
allocation.matrix	Logical, is A an allocation matrix with mRNAs in its columns and miRNAs in its rows, or is it an allocation data.frame?
verbose	Defaults to FALSE. If TRUE, output on progress is printed.
errors	Defaults to TRUE. If set to FALSE, some errors checking correct sizes of matrices are turned into warning messages.

**Value**

Matrix with testing results for every miRNA in its rows and the applied gene set test in its columns. Note that result will depend on whether multiple hypothesis testing correction was applied or not.

**Author(s)**

Stephan Artmann

**References**

- Artmann, Stephan and Jung, Klaus and Bleckmann, Annalen and Beissbarth, Tim (2012). Detection of simultaneous group effects in microRNA expression and related functional gene sets. *PLoS ONE* 7(6):e38365, PMID: 22723856.
- Brunner, E. (2009) Repeated measures under non-sphericity. *Proceedings of the 6th St. Petersburg Workshop on Simulation*, 605-609.
- Jelle J. Goeman, Sara A. van de Geer, Floor de Kort, Hans C. van Houwelingen (2004) A global test for groups of genes: testing association with a clinical outcome. *Bioinformatics* 20, 93-99.
- Jung, Klaus and Becker, Benjamin and Brunner, Edgar and Beissbarth, Tim (2011). Comparison of Global Tests for Functional Gene Sets in Two-Group Designs and Selection of Potentially Effect-causing Genes. *Bioinformatics*, 27: 1377-1383.
- Majewski, IJ, Ritchie, ME, Phipson, B, Corbin, J, Pakusch, M, Ebert, A, Busslinger, M, Koseki, H, Hu, Y, Smyth, GK, Alexander, WS, Hilton, DJ, and Blewitt, ME (2010). Opposing roles of polycomb repressive complexes in hematopoietic stem and progenitor cells. *\_Blood\_*, published online 5 May 2010.
- Mansmann, U. and Meister, R., 2005, Testing differential gene expression in functional groups, *\_Methods Inf Med\_* 44 (3).
- Smyth, G. K. (2004). Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *\_Statistical Applications in Genetics and Molecular Biology\_*, Volume \*3\*, Article 3.
- Wu, D, Lim, E, Francois Vaillant, F, Asselin-Labat, M-L, Visvader, JE, and Smyth, GK (2010). ROAST: rotation gene set tests for complex microarray experiments. *\_Bioinformatics\_*, published online 7 July 2010.

**Examples**

```
#####
### Generate random expression data ###
#####
# Generate random miRNA expression data of 3 miRNAs
# with 8 replicates
set.seed(1)
X = rnorm(24);
dim(X) = c(3,8);
rownames(X) = 1:3;
# Generate random mRNA expression data with 20 mRNAs
# and 10 replicates
Y = rnorm(200);
```

```

dim(Y) = c(20,10);
rownames(Y) = 1:20;
# Let's assume that we want to compare 2 miRNA groups, each of 4 replicates:
group.miRNA = factor(c(1,1,1,1,2,2,2,2));
# ... and that the corresponding mRNA experiments had 5 replicates in each group
group.mRNA = factor(c(1,1,1,1,1,2,2,2,2,2));
#####
### Perform Test ###
#####
library(miRtest)
#Let miRNA 1 attack mRNAs 1 to 9 and miRNA 2 attack mRNAs 10 to 17.
# mRNAs 18 to 20 are not attacked. miRNA 3 has no gene set.
miR = c(rep(1,9),c(rep(2,8)));
mRNAs = 1:17;
A = data.frame(mRNAs,miR); # Note that the miRNAs MUST be in the second column!
A
set.seed(1)
P = miR.test(X,Y,A,group.miRNA,group.mRNA)
P

#####
### For a faster result: use other gene set tests ###
#####
# Wilcoxon two-sample test is recommended for fast results
# Note that results may vary depending on how much genes correlate

P.gsWilcox = miR.test(X,Y,A,group.miRNA,group.mRNA,gene.set.tests="W")
P.gsWilcox
#####
### We can use an allocation matrix as A ###
#####
A = generate.A(A,X=X,Y=Y,verbose=FALSE);
A
# Now we can test as before
set.seed(1)
P = miR.test(X,Y,A,group.miRNA,group.mRNA,allocation.matrix=TRUE)
P

#####
### Other Designs ###
#####

# Some more complicated designs are implemented, check the vignette "miRtest" for details.
group.miRNA = 1:8
group.mRNA = 1:10
covariable.miRNA = factor(c(1,2,3,4,1,2,3,4))    ### A covariable in miRNAs.
covariable.mRNA = factor(c(1,2,3,4,5,1,2,3,4,5)) ### A covariable in mRNAs.

library(limma)
design.miRNA = model.matrix(~group.miRNA + covariable.miRNA)
design.mRNA = model.matrix(~group.mRNA + covariable.mRNA)

```

```
P = miR.test(X,Y,A,design.miRNA=design.miRNA,design.mRNA=design.mRNA,allocation.matrix=TRUE)
P
```

---

miRtest

*Package Description: Two-group combined miRNA- and mRNA- expression testing.*


---

## Description

Looking for differential expression in miRNA-data can have low power. Taking their respective mRNA-gene sets on the other hand can lead to too liberal results. In Artmann et al. we proposed a method to combine both information sources and generate p-values that can detect either miRNA- and target gene set expression differences.

## Details

Package: miRtest  
 Type: Package  
 Version: 1.8  
 Date: 2014-11-25  
 License: GPL  
 LazyLoad: yes  
 URL: <http://www.ncbi.nlm.nih.gov/pubmed/22723856>

For a detailed help check `vignette("miRtest")`

You can start the test with the "miR.test" function, which needs the expression matrix X of miRNAs, the expression matrix Y of mRNAs and the allocation matrix.

## Author(s)

Stephan Artmann <[stephanartmann@gmx.net](mailto:stephanartmann@gmx.net)>, Klaus Jung, Tim Beissbarth

Maintainer: Stephan Artmann <[stephanartmann@gmx.net](mailto:stephanartmann@gmx.net)>

## References

Artmann, Stephan and Jung, Klaus and Bleckmann, Annalen and Beissbarth, Tim (2012). Detection of simultaneous group effects in microRNA expression and related functional gene sets. Plos ONE, PMID: 22723856.

Brunner, E. (2009) Repeated measures under non-sphericity. Proceedings of the 6th St. Petersburg Workshop on Simulation, 605-609.

Jelle J. Goeman, Sara A. van de Geer, Floor de Kort, Hans C. van Houwelingen (2004) A global test for groups of genes: testing association with a clinical outcome. Bioinformatics 20, 93-99.

Jung, Klaus and Becker, Benjamin and Brunner, Edgar and Beissbarth, Tim (2011). Comparison of Global Tests for Functional Gene Sets in Two-Group Designs and Selection of Potentially Effect-causing Genes. *Bioinformatics*, 27: 1377-1383.

Majewski, IJ, Ritchie, ME, Phipson, B, Corbin, J, Pakusch, M, Ebert, A, Busslinger, M, Koseki, H, Hu, Y, Smyth, GK, Alexander, WS, Hilton, DJ, and Blewitt, ME (2010). Opposing roles of polycomb repressive complexes in hematopoietic stem and progenitor cells. *\_Blood\_*, published online 5 May 2010.

Mansmann, U. and Meister, R., 2005, Testing differential gene expression in functional groups, *\_Methods Inf Med\_* 44 (3).

Smyth, G. K. (2004). Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *\_Statistical Applications in Genetics and Molecular Biology\_*, Volume \*3\*, Article 3.

Wu, D, Lim, E, Francois Vaillant, F, Asselin-Labat, M-L, Visvader, JE, and Smyth, GK (2010). ROAST: rotation gene set tests for complex microarray experiments. *\_Bioinformatics\_*, published online 7 July 2010.

### See Also

Function "generate.A" as well as main function "miR.test"

### Examples

```
#####
### Generate random expression data ###
#####
# Generate random miRNA expression data of 3 miRNAs
# with 8 replicates
set.seed(1)
X = rnorm(24);
dim(X) = c(3,8);
rownames(X) = 1:3;
# Generate random mRNA expression data with 20 mRNAs
# and 10 replicates
Y = rnorm(200);
dim(Y) = c(20,10);
rownames(Y) = 1:20;
# Let's assume that we want to compare 2 miRNA groups, each of 4 replicates:
group.miRNA = factor(c(1,1,1,1,2,2,2,2));
# ... and that the corresponding mRNA experiments had 5 replicates in each group
group.mRNA = factor(c(1,1,1,1,1,2,2,2,2,2));
#####
### Perform Test ###
#####
library(miRtest)
#Let miRNA 1 attack mRNAs 1 to 9 and miRNA 2 attack mRNAs 10 to 17.
# mRNAs 18 to 20 are not attacked. miRNA 3 has no gene set.
miR = c(rep(1,9),c(rep(2,8)));
mRNAs = 1:17;
A = data.frame(mRNAs,miR); # Note that the miRNAs MUST be in the second column!
A
```

```

set.seed(1)
P = miR.test(X,Y,A,group.miRNA,group.mRNA)
P

#####
### For a faster result: use other gene set tests ###
#####
# Wilcoxon two-sample test is recommended for fast results
# Note that results may vary depending on how much genes correlate

P.gsWilcox = miR.test(X,Y,A,group.miRNA,group.mRNA,gene.set.tests="W")
P.gsWilcox
#####
### We can use an allocation matrix as A ###
#####
A = generate.A(A,X=X,Y=Y,verbose=FALSE);
A
# Now we can test as before
set.seed(1)
P = miR.test(X,Y,A,group.miRNA,group.mRNA,allocation.matrix=TRUE)
P

#####
### Other Designs ###
#####

# Some more complicated designs are implemented, check the vignette "miRtest" for details.
group.miRNA = 1:8
group.mRNA = 1:10
covariable.miRNA = factor(c(1,2,3,4,1,2,3,4))   ### A covariable in miRNAs.
covariable.mRNA = factor(c(1,2,3,4,5,1,2,3,4,5)) ### A covariable in mRNAs.

library(limma)
design.miRNA = model.matrix(~group.miRNA + covariable.miRNA)
design.mRNA = model.matrix(~group.mRNA + covariable.mRNA)

P = miR.test(X,Y,A,design.miRNA=design.miRNA,design.mRNA=design.mRNA,allocation.matrix=TRUE)
P

```

---

X

---

X

---

## Description

Part of expression data from Nielsen et al.

## Author(s)

Stephan Artmann taken from Nielsen et al.

---

Y

Y

---

**Description**

Part of expression data from Nielsen et al.

**Author(s)**

Stephan Artmann taken from Nielsen et al.

# Index

\*Topic **combined testing**,

miRtest, [12](#)

\*Topic **expression**

miRtest, [12](#)

\*Topic **package, miRNA**

miRtest, [12](#)

A, [2](#)

contingency.table, [2](#)

fisher.combination, [3](#)

generate.A, [3](#)

gs.test, [5](#)

inverse.normal.combination, [7](#)

limma.one.sided, [7](#)

limma.test, [8](#)

m.combine, [8](#)

miR.test, [9](#)

miRtest, [12](#)

X, [14](#)

Y, [15](#)