# Package 'metapone'

July 2, 2025

**Type** Package

**Title** Conducts pathway test of metabolomics data using a weighted permutation test

**Version** 1.15.0

**Date** 2022-8-20

**Description** The package conducts pathway testing from untargetted metabolomics data. It requires the user to supply feature-level test results, from case-control testing, regression, or other suitable feature-level tests for the study design. Weights are given to metabolic features based on how many metabolites they could potentially match to. The package can combine positive and negative mode results in pathway tests.

**Depends** R (>= 4.1.0), BiocParallel, fields, markdown, fdrtool, fgsea, ggplot2, ggrepel

**Imports** methods

**biocViews** Technology, MassSpectrometry, Metabolomics, Pathways

**License** Artistic-2.0

**LazyLoad** yes

**NeedsCompilation** no

**Suggests** rmarkdown, knitr

**VignetteBuilder** knitr

**git_url** https://git.bioconductor.org/packages/metapone

**git_branch** devel

**git_last_commit** d8c1c3c

**git_last_commit_date** 2025-04-15

**Repository** Bioconductor 3.22

**Date/Publication** 2025-07-01

**Author** Leqi Tian [aut],
Tianwei Yu [aut],
Tianwei Yu [cre]

**Maintainer** Tianwei Yu <yutianwei@cuhk.edu.cn>

# Contents



---

`metapone-package` *Conducts pathway test of metabolomics data using a weighted permutation test*

---

## Description

The package conducts pathway testing from untargetted metabolomics data. It requires the user to supply feature-level test results, from case-control testing, regression, or other suitable feature-level tests for the study design. Weights are given to metabolic features based on how many metabolites they could potentially match to. The package can combine positive and negative mode results in pathway tests. The package contains two types of statistical testing that considers matching uncertainty - (1) a permutation test that is based on the hypergeometric test and (2) a GSEA type test with weighted features/metabolites.

## Details

The package conducts (1) a weighted hypergeometric test using permutations on metabolomics data. The weights are assigned based on how many metabolites each data feature can match to, (2) a GSEA type test based on an estimation of importance of metabolites/features. The importance is evluated by the size of matching for each metabolite/feature and the p-value of features.

The user can tune a parameter to change the penalty for multiple-matched features and choose the type of pathway testing.

## Author(s)

Tianwei Yu (<yutianwei@cuhk.edu.cn>)

bbplot1d *Plot of metapone result.*

## Description

The function bbplot1d() select important pathways with their P-value less than a threshold and returns ranked bubble plot showing important pathways names and their corresponding -log10(Pvalue).

## Usage

```
bbplot1d(res, p_thres = 0.05, sig_metab_thres = 1, low.color = "MidnightBlue", high.color = "LightSk
```

## Arguments

| | |
|---|---|
| res | The result matrix obtained from metapone with columns: "p_value", "n_significant metabolites", "n_mapped_metabolites", "n_metabolites", "significant metabolites", "mapped_metabolites", "fdr". |
| p_thres | The threshold of P-value for pathways to be shown in the bubble plot. The default threshold is 0.05. |
| sig_metab_thres | The threshold of fractional matched significant metabolite count for pathways to be shown in the bubble plot. The default is 1. |
| low.color | The GRB color of the lowest ldfr value to be shown in the bubble plot. |
| high.color | The GRB color of the highest ldfr value to be shown in the bubble plot. |

## Author(s)

Leqi Tian (<leqitian@link.cuhk.edu.cn>)

## See Also

metapone

## Examples

```
data(hmdbCompMZ.metapone)
data(pa)
data(pos)
data(neg)
dat <- list(pos, neg)
type <- list("pos", "neg")
r<-metapone(dat, type, pa, hmdbCompMZ=hmdbCompMZ.metapone, p.threshold=0.05,
   n.permu=100,fractional.count.power=0.5, max.match.count=10)
bbplot1d(ptable(r)) # p_thres = 0.05
```

bbplot2d *Plot of metapone result.*

## Description

The function bbplot2d() select important pathways with their P-value less than a threshold and returns a 2-D bubble plot with -log10(Pvalue) and the number of significant metabolites as coordinate axes.

## Usage

```
bbplot2d(res, p_thres = 0.05, sig_metab_thres = 1, low.color = "MidnightBlue", high.color = "LightSk
```

## Arguments

| | |
|---|---|
| res | The result matrix obtained from metapone with columns: "p_value", "n_significant metabolites", "n_mapped_metabolites", "n_metabolites", "significant metabolites", "mapped_metabolites", "fdr". |
| p_thres | The threshold of P-value for pathways to be shown in the bubble plot. The default threshold is 0.05. |
| sig_metab_thres | The threshold of fractional matched significant metabolite count for pathways to be shown in the bubble plot. The default is 1. |
| low.color | The GRB color of the lowest ldfr value to be shown in the bubble plot. |
| high.color | The GRB color of the highest ldfr value to be shown in the bubble plot. |

## Author(s)

Leqi Tian (<leqitian@link.cuhk.edu.cn>)

## See Also

metapone

## Examples

```
data(hmdbCompMZ.metapone)
data(pa)
data(pos)
data(neg)
dat <- list(pos, neg)
type <- list("pos", "neg")
r<-metapone(dat, type, pa, hmdbCompMZ=hmdbCompMZ.metapone, p.threshold=0.05,
   n.permu=100,fractional.count.power=0.5, max.match.count=10)
bbplot2d(ptable(r)) # p_thres = 0.05
```

ftable *Acessor functions for the feature mapping table in a metaponeResult object.*

## Description

Returns a `list` containing the mapped features in each pathway.

## Usage

```
## S4 method for signature 'metaponeResult'
ftable(object)
```

## Arguments

| | |
|---|---|
| `object` | A metaponeResult object. |

## Details

Each pathway is represented by a `data.frame` as an item in the list object. The dataframe include information of m.z, retention.time, p.value, statistic, HMDB_ID, theoretical m.z, ion.type, fractional counts.

## Value

The method returns a list. Each item is for a pathway. Matched significant metabolites are included.

## Author(s)

Tianwei Yu <yutianwei@cuhk.edu.cn>

## See Also

ptable

## Examples

```
data(hmdbCompMZ.metapone)
data(pa)
data(pos)
data(neg)
dat <- list(pos, neg)
type <- list("pos", "neg")
r<-metapone(dat, type, pa, hmdbCompMZ=hmdbCompMZ.metapone, p.threshold=0.05,
   n.permu=100,fractional.count.power=0.5, max.match.count=10)
ftable(r)[1:6]
```

## hmdbCompMZ *the m/z values of common adduct ions of HMDB metaboites*

### Description

Monoisotopic mass of common adduct ions.

### Usage

```
data("hmdbCompMZ")
```

### Format

A data frame with 5704350 observations on the following 3 variables.

`HMDB_ID` HMDB ID.

`ion.type` Adduct ion type.

`m.z` the m/z of the adduct ion.

### Source

https://hmdb.ca/

### References

https://hmdb.ca/

### Examples

```
data(hmdbCompMZ)
```

## hmdbCompMZ.metapone *the m/z values of common adduct ions of metapone metaboites*

### Description

Monoisotopic mass of common adduct ions, limited to those included in the pathways in metapone.

### Usage

```
data("hmdbCompMZ.metapone")
```

### Format

A data frame with 79350 observations on the following 3 variables.

`HMDB_ID` HMDB ID.

`ion.type` Adduct ion type.

`m.z` the m/z of the adduct ion.

### Details

The main difference of using this dataset vs using hmdbCompMZ, is the metabolite universe in testing is limited to those metabolites matched to metapone pathways, not all HMDB metabolites.

### Source

The Human Metabolome Database

### References

The Human Metabolome Database

### Examples

```
data(hmdbCompMZ)
```

---

## metapone *METAbolic pathway testing using both POsitive and NEgative mode data*

---

### Description

Metapone conducts pathway tests for untargeted metabolomics data. It has three main characteristics: (1) expanded database combining SMPDB and Mummichog databases, with manual cleaning to remove redundancies; (2) A new weighted testing scheme to address the issue of metabolite-feature matching uncertainties; (3) Can consider positive mode and negative mode data in a single analysis.

### Usage

```
metapone(dat=NULL, type=NULL, pa, hmdbCompMZ, pos.adductlist = c("M+H",
"M+NH4", "M+Na", "M+ACN+H","M+ACN+Na", "M+2ACN+H", "2M+H", "2M+Na",
"2M+ACN+H"), neg.adductlist = c("M-H","M-2H","M-2H+Na","M-2H+K",
"M-2H+NH4","M-H2O-H","M-H+Cl", "M+Cl", "M+2Cl"),
use.fractional.count=TRUE, match.tol.ppm=5, p.threshold=0.05,
n.permu=200, fractional.count.power=0.5, max.match.count=10,
use.fgsea = FALSE, use.meta = FALSE)
```

### Arguments

| | |
|---|---|
| dat | The list of test results. An element in the list should be postive ion mode test results or negative ion mode test results with four columns: m/z, retention time, p-value, test statistic. The package doesn't require both pos and neg to be present. One ion mode result is sufficient. Multiple ion mode results are allowed. |
| type | The list of corresponding ion mode of each element in dat. Each element in the list should be "pos" or "neg". The size of type should be consistent with the size of dat. |
| pa | Pathway information. A data frame with five columns: database pathway ID, pathway name, HMDB ID, KEGG ID, category of pathway. |
| hmdbCompMZ | the m/z values of common adduct ions of HMDB metaboites. See the help file of hmdbCompMZ for details. |

`pos.adductlist` The vector of positive adduct ions to be considered.

`neg.adductlist` The vector of negative adduct ions to be considered.

`use.fractional.count`
A lot of features match to multiple metabolites by m/z. Whether to discount such matches by using fractional counts.

`match.tol.ppm` The ppm level when conducting m/z match.

`p.threshold` The threshold of p-values of metabolic features to be considered significant.

`n.permu` The number of permutations in permutation test.

`fractional.count.power`
The fractional counts are taken to this power to transform the weights.

`max.match.count`
When calculating fractional counts, some features might be matched to too many. In that case the number of matches is capped by the value of max.match.count.

`use.fgsea` Whether to use a GSEA type test when performing pathway testing. When it is FALSE, a permutation-based weighted hypergeometric test is performed.

`use.meta` Whether to perform a GSEA type test with weighted metabolites. When it is FALSE, a GSEA type test is performed on weighted features.

## Value

The method returns a generic S4 object of class "metapone.result":

`@test.results` A matrix with 8 columns: "p_value", "n_significant metabolites", "n_mapped_metabolites", "n_metabolites", "significant metabolites", "mapped_metabolites", "lfdr", "adjust.p". Each row is for a pathway. When using GSEA test, "ES", "NES", "nMoreExtreme" are returned additionally.

`@mapped.features`
A list. Each item is for a pathway. The item lists matched significant metabolites.

The columns in test.result are the following:

`p_value` The p-value for each enrichment.

`n_significant metabolites`
The number of weighted significant metabolites associated with the enrichment.

`n_mapped_metabolites`
The number of weighted metabolites associated with the enrichment.

`n_metabolites` The number of metabolites associated with the enrichment.

`significant metabolites`
A string with the names of significant metabolites that drive the enrichment.

`mapped_metabolites`
A string with the names of metabolites that drive the enrichment.

`lfdr` The local fdr value for each enrichment.

`adjust.p` The enrichment BH-adjusted p-value for each enrichment.

`ES` The enrichment score (Avaliable in GSEA test).

`NES` The enrichment score normalized to mean enrichment of random samples of the same size (Avaliable in GSEA test).

`nMoreExtreme` The number of times a random metabolite set had a more extreme enrichment score value (Avaliable in GSEA test).

## Author(s)

Tianwei Yu (<yutianwei@cuhk.edu.cn>) Leqi Tian (<leqitian@link.cuhk.edu.cn>)

## References

Small Molecule Pathway Database

Mummichog

## See Also

pa, hmdbCompMZ

## Examples

```
data(hmdbCompMZ.metapone)
data(pa)
data(pos)
data(neg)
dat <- list(pos, neg)
type <- list("pos", "neg")
# Permutation-based weighted hypergeometric test
r<-metapone(dat, type, pa, hmdbCompMZ=hmdbCompMZ.metapone, p.threshold=0.05,
   n.permu=100,fractional.count.power=0.5, max.match.count=10)
hist(ptable(r)[,1])

# Metabolites based GSEA test
r<-metapone(dat, type, pa, hmdbCompMZ=hmdbCompMZ.metapone, p.threshold=0.05,
  n.permu=100,fractional.count.power=0.5, max.match.count=10, use.fgsea = TRUE, use.meta = TRUE)
hist(ptable(r)[,1])

# Features based GSEA test
r<-metapone(dat, type, pa, hmdbCompMZ=hmdbCompMZ.metapone, p.threshold=0.05,
  n.permu=100,fractional.count.power=0.5, max.match.count=10, use.fgsea = FALSE, use.meta = FALSE)
hist(ptable(r)[,1])
```

---

# metaponeResult-class *Class "metaponeResult"*

---

## Description

This class represents the results of pathway testing. The testing result contain two major components: the significant level of each pathway, and the features matched to each pathway.

## Objects from the Class

Objects can be created by calls of the form `new("metaponeResult", ...)`.

## Slots

`test.result`: a dataframe containing p_value, n_significant metabolites, n_mapped_metabolites, n_metabolites, significant metabolites, mapped_metabolite IDs, lfdr and pathway name.

`mapped.features`: A list containing n entries, where n is the number of pathways. Each entry is a data frame, containing the features mapped to this pathway. The information include m.z, retention.time, p.value, statistic, HMDB_ID, theoretical m.z, ion.type, fractional counts.

## Methods

**ptable** `signature(object = "metaponeResult")`: return the `data.frame` of test statistics for each pathway, including p_value, n_significant metabolites, n_mapped_metabolites, n_metabolites, significant metabolites, mapped_metabolite IDs lfdr and and pathway name.

**ftable** `signature(object = "metaponeResult")`: Returns a `list` containing the mapped features in each pathway. Each pathway is represented by a `data.frame` as an item in the list object. The dataframe include information of m.z, retention.time, p.value, statistic, HMDB_ID, theoretical m.z, ion.type, fractional counts.

## Author(s)

Tianwei Yu

---

# neg *Example negative mode data from the Metabolome Atlas of the Aging Mouse Brain*

---

## Description

The data is generated from the hypocampus data of the Metabolome Atlas of the Aging Mouse Brain (ST001888) dataset. The p-values and test statistics were obtained by contrasting mouse hypocampus metabolome between prime-age mice and aging mice.

## Usage

```
data("neg")
```

## Format

A data frame with 6947 observations on the following 4 variables.

`m.z` a numeric vector. The mass-to-charge ratio of the features.

`retention.time` a numeric vector. The retention time of the features.

`p.value` a numeric vector. The p-values of the features.

`statistic` a numeric vector. The test statistics of the features.

## References

https://www.metabolomicsworkbench.org/data/DRCCMetadata.php?Mode=Study&DataMode=FactorsData&StudyID=

## Examples

```
data(neg)
```

## pa *Pathway-metabolite match file.*

### Description

mapps pathways with metabolites.

### Usage

```
data("pa")
```

### Format

A data frame with 5395 observations on the following 5 variables.

`database` a character vector

`pathway.name` a character vector

`HMDB.ID` a character vector

`KEGG.ID` a character vector

`category` a character vector

### Source

Small Molecule Pathway Database

Mummichog

### Examples

```
data(pa)
```

## pos *Example positive mode data from the Metabolome Atlas of the Aging Mouse Brain*

### Description

The data is generated from the hypocampus data of the Metabolome Atlas of the Aging Mouse Brain (ST001888) dataset. The p-values and test statistics were obtained by contrasting mouse hypocampus metabolome between prime-age mice and aging mice.

### Usage

```
data("pos")
```

## Format

A data frame with 10085 observations on the following 4 variables.

`m.z` a numeric vector. The mass-to-charge ratio of the features.

`retention.time` a numeric vector. The retention time of the features.

`p.value` a numeric vector. The p-values of the features.

`statistic` a numeric vector. The test statistics of the features.

## References

https://www.metabolomicsworkbench.org/data/DRCCMetadata.php?Mode=Study&DataMode=FactorsData&StudyID=

## Examples

```
data(pos)
```

---

`ptable` *Acessor functions for the test result table in a metaponeResult object.*

---

## Description

return the `data.frame` of test statistics for each pathway.

## Usage

```
## S4 method for signature 'metaponeResult'
ptable(object)
```

## Arguments

| | |
|---|---|
| `object` | A metaponeResult object. |

## Details

Includes p_value, n_significant metabolites, n_mapped_metabolites, n_metabolites, significant metabolites, mapped_metabolite IDs and pathway name.

## Value

The method returns a data frame with 6 columns: "p_value", "n_significant metabolites", "n_mapped_metabolites", "n_metabolites", "significant metabolites", "mapped_metabolites".

## Author(s)

Tianwei Yu <yutianwei@cuhk.edu.cn>

## See Also

ftable

## Examples

```
data(hmdbCompMZ.metapone)
data(pa)
data(pos)
data(neg)
dat <- list(pos, neg)
type <- list("pos", "neg")
r<-metapone(dat, type, pa, hmdbCompMZ=hmdbCompMZ.metapone,
   p.threshold=0.05,n.permu=100,fractional.count.power=0.5, max.match.count=10)
head(ptable(r))
```

# Index