

# Package ‘FinfoMDS’

April 7, 2026

**Type** Package

**Title** Multidimensional Scaling with F-ratio for microbiome visualization

**Version** 1.1.0

**Description** F-informed MDS is a new multidimensional scaling-based ordination method that configures data distribution based on the F-statistic (i.e., the ratio of dispersion between groups with shared or differing labels).

**License** GPL-3

**Encoding** UTF-8

**Suggests** testthat (>= 3.0.0), knitr, rmarkdown, BiocStyle

**VignetteBuilder** knitr

**biocViews** DimensionReduction, MultidimensionalScaling, Visualization, Microbiome

**URL** <https://github.com/soob-kim/FinfoMDS>

**BugReports** <https://github.com/soob-kim/FinfoMDS/issues>

**RoxygenNote** 7.3.2

**Imports** phyloseq

**Depends** R (>= 4.4.0)

**Config/testthat/edition** 3

**git\_url** <https://git.bioconductor.org/packages/FinfoMDS>

**git\_branch** devel

**git\_last\_commit** a94fee5

**git\_last\_commit\_date** 2025-10-29

**Repository** Bioconductor 3.23

**Date/Publication** 2026-04-07

**Author** Soobin Kim [aut, cre],  
Hyungseok Kim [aut]

**Maintainer** Soobin Kim <soob.kim00@gmail.com>

## Contents

fmds . . . . .	2
getDistMat . . . . .	3
getIndMat . . . . .	3
getP . . . . .	4
mdsObj . . . . .	5
microbiome . . . . .	5
pairByRank . . . . .	6
pseudoF . . . . .	7
<b>Index</b>	<b>8</b>

---

fmds	<i>FMDS calculation using MM algorithm</i>
------	--

---

### Description

FMDS calculation using MM algorithm

### Usage

```
fmds(D = NULL, y, X, nit = 100, lambda = 0.5, threshold_p = 0.05, z0 = NULL)
```

### Arguments

D	Square matrix of pairwise distance, size of N by N
y	Vector of label or group set, size of N
X	Object matrix; used to build distance matrix D; D is prioritized
nit	Number of iterations; 100 by default
lambda	Hyperparameter; 0.5 by default
threshold_p	Lower limit of p-value difference that allows iteration
z0	Initialization of configuration; NULL by default

### Value

2D representation vector, size of N by 2

### Examples

```
set.seed(100)
require(phyloseq)
data(microbiome)
D <- distance(microbiome, method = 'wunifrac') # requires phyloseq package
y <- sample_data(microbiome)$Treatment
z0 <- cmdscale(d = D)
fmds(z0 = z0, D = D, y = y)
```

---

getDistMat	<i>Distance between vectors</i>
------------	---------------------------------

---

**Description**

Distance between vectors

**Usage**

```
getDistMat(z)
```

**Arguments**

`z` Matrix or vector of observations

**Value**

Distance matrix

**Examples**

```
set.seed(100)
z <- rbind(matrix(rnorm(100), ncol=4),
matrix(rnorm(100, 2), ncol=4))
getDistMat(z)
```

---

getIndMat	<i>Get index matrix</i>
-----------	-------------------------

---

**Description**

Get index matrix

**Usage**

```
getIndMat(y)
```

**Arguments**

`y` a vector of treatments of length N

**Value**

A N by N matrix of indicators of equal treatment

**Examples**

```
require(phyloseq)
data(microbiome)
y <- sample_data(microbiome)$Treatment
getIndMat(y)
```

---

**getP***Compute p-value from PERMANOVA test*

---

**Description**

Compute p-value from PERMANOVA test

**Usage**

```
getP(z = NULL, D = NULL, y, n_iter = 999)
```

**Arguments**

z	Object matrix; used to build distance matrix d; d is prioritized
D	Distance matrix; if NULL, obtain from mat using Euclidean distance
y	Vector of treatments
n_iter	Number of iterations; defaults to 999

**Value**

list of ratio\_all: vector of obtained pseudo-F values from permutations, ratio: pseudo-F value, p: p-value from PERMANOVA

**Examples**

```
require(phyloseq)
data(microbiome)
D <- distance(microbiome, method = 'wunifrac') # requires phyloseq package
y <- sample_data(microbiome)$Treatment
getP(D = D, y = y)
```

---

mdsObj	<i>Objective term of MDS</i>
--------	------------------------------

---

**Description**

Objective term of MDS

**Usage**

```
mdsObj(D, z, N)
```

**Arguments**

D	Original distance matrix
z	Lower dimension representation
N	Number of observations–scaling factors

**Value**

Scalar of objective function value of MDS

**Examples**

```
require(phyloseq)
data(microbiome)
D <- distance(microbiome, method = 'wunifrac') # requires phyloseq package
z0 <- cmdscale(d = D)
N <- dim(z0)[1]
mdsObj(D = D, z = z0, N = N)
```

---

microbiome	<i>Microbiome dataset</i>
------------	---------------------------

---

**Description**

A phyloseq-class object representing an example microbial community dataset. The dataset consists of 36 microbiome samples, half of which were co-cultured with their diatom host (*Phaeodactylum tricorutum*). Each microbiome comprises 72 bacterial taxa identified by amplicon sequence variants (ASVs) of the 16S rRNA gene. ASV counts represent relative abundances after applying the cumulative sum scaling (CSS) method.

**Usage**

```
data(microbiome)
```

**Format**

A phyloseq-class object.

**Value**

A phyloseq-class object consisting of relative abundance table, taxonomy table, sample dataframe, and phylogenetic tree

**Source**

Data derived from an experimental study on host-microbe interactions.

**References**

Kim H., Kimbrel J.A., Vaiana C.A., Wollard J.R., Mayali X., Buie C.R. (2022). Bacterial response to spatial gradients of algal-derived nutrients in a porous microplate. *The ISME Journal*, 16(4), 1036–1045. doi:[10.1038/s41396021011634](https://doi.org/10.1038/s41396021011634)

---

pairByRank	<i>Get p-value matrix</i>
------------	---------------------------

---

**Description**

Get p-value matrix

**Usage**

```
pairByRank(z, D, y)
```

**Arguments**

z	Lower dimension representation
D	Original distance matrix
y	Treatment vector

**Value**

pseudo-F values matrix 1st col of original data, 2nd col of reduced dim

**Examples**

```
require(phyloseq)
data(microbiome)
D <- distance(microbiome, method = 'wunifrac') # requires phyloseq package
y <- sample_data(microbiome)$Treatment
z0 <- cmdscale(d = D)
pairByRank(z = z0, D = D, y = y)
```

---

pseudoF

*Compute pseudo-F statistic for PERMANOVA*

---

**Description**

Compute pseudo-F statistic for PERMANOVA

**Usage**

```
pseudoF(z = NULL, D = NULL, y)
```

**Arguments**

z	Object matrix; used to build distance matrix d; d is prioritized
D	Distance matrix; if NULL, obtain from mat using Euclidean distance
y	Vector of treatments

**Value**

pseudo-F value

**Examples**

```
require(phyloseq)
data(microbiome)
D <- distance(microbiome, method = 'wunifrac') # requires phyloseq package
y <- sample_data(microbiome)$Treatment
pseudoF(D = D, y = y)
```

# Index

\* **datasets**  
    microbiome, 5

fmds, 2

getDistMat, 3  
getIndMat, 3  
getP, 4

mdsObj, 5  
microbiome, 5

pairByRank, 6  
pseudoF, 7