# Package 'ANCOMBC'

June 5, 2023

**Type** Package

**Title** Microbiome differential abudance and correlation analyses with
bias correction

**Version** 2.3.0

**Description** ANCOMBC is a package containing differential abundance (DA) and
correlation analyses for microbiome data. Specifically, the package includes
Analysis of Compositions of Microbiomes with Bias Correction 2 (ANCOM-BC2),
Analysis of Compositions of Microbiomes with Bias Correction (ANCOM-BC), and
Analysis of Composition of Microbiomes (ANCOM) for DA analysis, and Sparse
Estimation of Correlations among Microbiomes (SECOM) for correlation
analysis. Microbiome data are typically subject to two sources of biases:
unequal sampling fractions (sample-specific biases) and differential
sequencing efficiencies (taxon-specific biases). Methodologies included in
the ANCOMBC package are designed to correct these biases and construct
statistically consistent estimators.

**Date** 2023-04-18

**License** Artistic-2.0

**Imports** mia, stats, CVXR, DescTools, Hmisc, MASS, Rdpack, S4Vectors,
SingleCellExperiment, SummarizedExperiment,
TreeSummarizedExperiment, doParallel, doRNG, dplyr, emmeans,
energy, foreach, lme4, lmerTest, magrittr, nloptr, parallel,
rlang, rngtools, tibble, tidyr, utils

**Suggests** knitr, rmarkdown, testthat, DT, caret, microbiome, tidyverse

**biocViews** DifferentialExpression, Microbiome, Normalization,
Sequencing, Software

**BugReports** https://github.com/FrederickHuangLin/ANCOMBC/issues

**URL** https://github.com/FrederickHuangLin/ANCOMBC

**VignetteBuilder** knitr

**RdMacros** Rdpack

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**Depends**  R (>= 4.2.0)

**LazyData**  false

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

**git_branch**  devel

**git_last_commit**  2c0d3dc

**git_last_commit_date**  2023-04-25

**Date/Publication**  2023-06-05

**Author**  Huang Lin [cre, aut] (<https://orcid.org/0000-0002-4892-7871>)

**Maintainer**  Huang Lin <huanglinfrederick@gmail.com>

# R topics documented:

---

ancom                                       *Analysis of Composition of Microbiomes (ANCOM)*

---

### Description

Determine taxa whose absolute abundances, per unit volume, of the ecosystem (e.g. gut) are significantly different with changes in the covariate of interest (e.g. group). The current version of `ancom` function implements ANCOM in cross-sectional and repeated measurements data while allowing for covariate adjustment.

### Usage

```
ancom(
  data = NULL,
  assay_name = "counts",
  tax_level = NULL,
  phyloseq = NULL,
  p_adj_method = "holm",
  prv_cut = 0.1,
  lib_cut = 0,
  main_var,
  adj_formula = NULL,
  rand_formula = NULL,
```

```
    lme_control = lme4::lmerControl(),
    struc_zero = FALSE,
    neg_lb = FALSE,
    alpha = 0.05,
    n_cl = 1
)
```

## Arguments

| | |
|---|---|
| data | the input data. A phyloseq, SummarizedExperiment, or TreeSummarizedExperiment object, which consists of a feature table (microbial count table), a sample metadata, a taxonomy table (optional), and a phylogenetic tree (optional). The row names of the metadata must match the sample names of the feature table, and the row names of the taxonomy table must match the taxon (feature) names of the feature table. See ?phyloseq::phyloseq, ?SummarizedExperiment::SummarizedExperiment, or ?TreeSummarizedExperiment::TreeSummarizedExperiment for more details. |
| assay_name | character. Name of the count table in the data object (only applicable if data object is a (Tree)SummarizedExperiment). Default is "counts". See ?SummarizedExperiment::assay for more details. |
| tax_level | character. The taxonomic level of interest. The input data can be agglomerated at different taxonomic levels based on your research interest. Default is NULL, i.e., do not perform agglomeration, and the ANCOM anlysis will be performed at the lowest taxonomic level of the input data. |
| phyloseq | a phyloseq object. Will be deprecated. |
| p_adj_method | character. method to adjust p-values. Default is "holm". Options include "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none". See ?stats::p.adjust for more details. |
| prv_cut | a numerical fraction between 0 and 1. Taxa with prevalences less than prv_cut will be excluded in the analysis. For instance, suppose there are 100 samples, if a taxon has nonzero counts presented in less than 10 samples, it will not be further analyzed. Default is 0.10. |
| lib_cut | a numerical threshold for filtering samples based on library sizes. Samples with library sizes less than lib_cut will be excluded in the analysis. Default is 0, i.e. do not discard any sample. |
| main_var | character. The name of the main variable of interest. |
| adj_formula | character string representing the formula for covariate adjustment. Default is NULL. |
| rand_formula | the character string expresses how the microbial absolute abundances for each taxon depend on the random effects in metadata. ANCOM follows the lmerTest package in formulating the random effects. See ?lmerTest::lmer for more details. Default is NULL. |
| lme_control | a list of control parameters for mixed model fitting. See ?lme4::lmerControl for details. |
| struc_zero | logical. whether to detect structural zeros based on main_var. main_var should be discrete. Default is FALSE. |

neg_lb              logical. whether to classify a taxon as a structural zero using its asymptotic
                    lower bound. Default is FALSE.

alpha               numeric. level of significance. Default is 0.05.

n_cl                numeric. The number of nodes to be forked. For details, see ?parallel::makeCluster.
                    Default is 1 (no parallel computing).

## Details

A taxon is considered to have structural zeros in some (>=1) groups if it is completely (or nearly
completely) missing in these groups. For instance, suppose there are three groups: g1, g2, and g3.
If the counts of taxon A in g1 are 0 but nonzero in g2 and g3, then taxon A will be considered to
contain structural zeros in g1. In this example, taxon A is declared to be differentially abundant
between g1 and g2, g1 and g3, and consequently, it is globally differentially abundant with respect
to this group variable. Such taxa are not further analyzed using ANCOM-BC, but the results are
summarized in the overall summary. For more details about the structural zeros, please go to the
ANCOM-II paper. Setting neg_lb = TRUE indicates that you are using both criteria stated in section
3.2 of ANCOM-II to detect structural zeros; otherwise, the algorithm will only use the equation 1 in
section 3.2 for declaring structural zeros. Generally, it is recommended to set neg_lb = TRUE when
the sample size per group is relatively large (e.g. > 30).

## Value

a list with components:

- res, a data.frame containing ANCOM result for the variable specified in main_var, each
  column is:
    - W, test statistics.
    - detected_0.9, detected_0.8, detected_0.7, detected_0.6, logical vectors repre-
      senting whether a taxon is differentially abundant under a series of cutoffs. For example,
      TRUE in detected_0.7 means the number of ALR transformed models where the taxon
      is differentially abundant with regard to the main variable outnumbers 0.7 * (n_tax -
      1). detected_0.7 is commonly used. Choose detected_0.8 or detected_0.9 for
      more conservative results, or choose detected_0.6 for more liberal results.

- zero_ind, a logical data.frame with TRUE indicating the taxon is detected to contain struc-
  tural zeros in some specific groups.

- beta_data, a numeric matrix containing pairwise coefficients for the main variable of interest
  in ALR transformed regression models.

- p_data, a numeric matrix containing pairwise p-values for the main variable of interest in
  ALR transformed regression models.

- q_data, a numeric matrix containing adjusted p-values by applying the p_adj_method to the
  p_data matrix.

## Author(s)

Huang Lin

## References

Mandal S, Van Treuren W, White RA, Eggesbo M, Knight R, Peddada SD (2015). "Analysis of composition of microbiomes: a novel method for studying microbial composition." *Microbial ecology in health and disease*, **26**(1), 27663.

Kaul A, Mandal S, Davidov O, Peddada SD (2017). "Analysis of microbiome data in the presence of excess zeros." *Frontiers in microbiology*, **8**, 2114.

## See Also

ancombc ancombc2

## Examples

```
library(ANCOMBC)
library(mia)
data(atlas1006, package = "microbiome")
tse = mia::makeTreeSummarizedExperimentFromPhyloseq(atlas1006)

# subset to baseline
tse = tse[, tse$time == 0]

# run ancom function
set.seed(123)
out = ancom(data = tse, assay_name = "counts",
            tax_level = "Family", phyloseq = NULL,
            p_adj_method = "holm", prv_cut = 0.10, lib_cut = 1000,
            main_var = "bmi_group", adj_formula = "age + nationality",
            rand_formula = NULL, lme_control = NULL,
            struc_zero = TRUE, neg_lb = TRUE, alpha = 0.05, n_cl = 1)

res = out$res

# to run ancom using the phyloseq object
tse_alt = agglomerateByRank(tse, "Family")
pseq = makePhyloseqFromTreeSummarizedExperiment(tse_alt)
set.seed(123)
out = ancom(data = NULL, assay_name = NULL,
            tax_level = "Family", phyloseq = pseq,
            p_adj_method = "holm", prv_cut = 0.10, lib_cut = 1000,
            main_var = "bmi_group", adj_formula = "age + nationality",
            rand_formula = NULL, lme_control = NULL,
            struc_zero = TRUE, neg_lb = TRUE, alpha = 0.05, n_cl = 1)
```

---

| ancombc | *Analysis of Compositions of Microbiomes with Bias Correction (ANCOM-BC)* |
|---|---|

---

## Description

Determine taxa whose absolute abundances, per unit volume, of the ecosystem (e.g., gut) are significantly different with changes in the covariate of interest (e.g., group). The current version of ancombc function implements Analysis of Compositions of Microbiomes with Bias Correction (ANCOM-BC) in cross-sectional data while allowing for covariate adjustment.

## Usage

```
ancombc(
  data = NULL,
  assay_name = "counts",
  tax_level = NULL,
  phyloseq = NULL,
  formula,
  p_adj_method = "holm",
  prv_cut = 0.1,
  lib_cut = 0,
  group = NULL,
  struc_zero = FALSE,
  neg_lb = FALSE,
  tol = 1e-05,
  max_iter = 100,
  conserve = FALSE,
  alpha = 0.05,
  global = FALSE,
  n_cl = 1,
  verbose = FALSE
)
```

## Arguments

| | |
|---|---|
| data | the input data. A phyloseq, SummarizedExperiment, or TreeSummarizedExperiment object, which consists of a feature table (microbial count table), a sample metadata, a taxonomy table (optional), and a phylogenetic tree (optional). The row names of the metadata must match the sample names of the feature table, and the row names of the taxonomy table must match the taxon (feature) names of the feature table. See ?phyloseq::phyloseq, ?SummarizedExperiment::SummarizedExperiment, or ?TreeSummarizedExperiment::TreeSummarizedExperiment for more details. |
| assay_name | character. Name of the count table in the data object (only applicable if data object is a (Tree)SummarizedExperiment). Default is "counts". See ?SummarizedExperiment::assay for more details. |
| tax_level | character. The taxonomic level of interest. The input data can be agglomerated at different taxonomic levels based on your research interest. Default is NULL, i.e., do not perform agglomeration, and the ANCOM-BC anlysis will be performed at the lowest taxonomic level of the input data. |
| phyloseq | a phyloseq object. Will be deprecated. |

| | |
|---|---|
| formula | the character string expresses how microbial absolute abundances for each taxon depend on the variables in metadata. |
| p_adj_method | character. method to adjust p-values. Default is "holm". Options include "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none". See `?stats::p.adjust` for more details. |
| prv_cut | a numerical fraction between 0 and 1. Taxa with prevalences less than `prv_cut` will be excluded in the analysis. For instance, suppose there are 100 samples, if a taxon has nonzero counts presented in less than 10 samples, it will not be further analyzed. Default is 0.10. |
| lib_cut | a numerical threshold for filtering samples based on library sizes. Samples with library sizes less than `lib_cut` will be excluded in the analysis. Default is 0, i.e. do not discard any sample. |
| group | character. the name of the group variable in metadata. `group` should be discrete. Specifying `group` is required for detecting structural zeros and performing global test. Default is NULL. If the `group` of interest contains only two categories, leave it as NULL. |
| struc_zero | logical. whether to detect structural zeros based on `group`. Default is FALSE. |
| neg_lb | logical. whether to classify a taxon as a structural zero using its asymptotic lower bound. Default is FALSE. |
| tol | numeric. the iteration convergence tolerance for the E-M algorithm. Default is 1e-05. |
| max_iter | numeric. the maximum number of iterations for the E-M algorithm. Default is 100. |
| conserve | logical. whether to use a conservative variance estimator for the test statistic. It is recommended if the sample size is small and/or the number of differentially abundant taxa is believed to be large. Default is FALSE. |
| alpha | numeric. level of significance. Default is 0.05. |
| global | logical. whether to perform the global test. Default is FALSE. |
| n_cl | numeric. The number of nodes to be forked. For details, see `?parallel::makeCluster`. Default is 1 (no parallel computing). |
| verbose | logical. Whether to generate verbose output during the ANCOM-BC fitting process. Default is FALSE. |

## Details

A taxon is considered to have structural zeros in some (>=1) groups if it is completely (or nearly completely) missing in these groups. For instance, suppose there are three groups: g1, g2, and g3. If the counts of taxon A in g1 are 0 but nonzero in g2 and g3, then taxon A will be considered to contain structural zeros in g1. In this example, taxon A is declared to be differentially abundant between g1 and g2, g1 and g3, and consequently, it is globally differentially abundant with respect to this group variable. Such taxa are not further analyzed using ANCOM-BC, but the results are summarized in the overall summary. For more details about the structural zeros, please go to the [ANCOM-II](#) paper. Setting neg_lb = TRUE indicates that you are using both criteria stated in section 3.2 of [ANCOM-II](#) to detect structural zeros; otherwise, the algorithm will only use the equation 1 in section 3.2 for declaring structural zeros. Generally, it is recommended to set neg_lb = TRUE when the sample size per group is relatively large (e.g. > 30).

**Value**

a `list` with components:

- `feature_table`, a `data.frame` of pre-processed (based on `prv_cut` and `lib_cut`) microbial count table.

- `zero_ind`, a logical `data.frame` with TRUE indicating the taxon is detected to contain structural zeros in some specific groups.

- `samp_frac`, a numeric vector of estimated sampling fractions in log scale (natural log).

- `delta_em`, estimated sample-specific biases through E-M algorithm.

- `delta_wls`, estimated sample-specific biases through weighted least squares (WLS) algorithm.

- `res`, a `list` containing ANCOM-BC primary result, which consists of:

  - `lfc`, a `data.frame` of log fold changes obtained from the ANCOM-BC log-linear (natural log) model.

  - `se`, a `data.frame` of standard errors (SEs) of `lfc`.

  - `W`, a `data.frame` of test statistics. `W = lfc/se`.

  - `p_val`, a `data.frame` of p-values. P-values are obtained from two-sided Z-test using the test statistic `W`.

  - `q_val`, a `data.frame` of adjusted p-values. Adjusted p-values are obtained by applying `p_adj_method` to `p_val`.

  - `diff_abn`, a logical `data.frame`. TRUE if the taxon has `q_val` less than `alpha`.

- `res_global`, a `data.frame` containing ANCOM-BC global test result for the variable specified in `group`, each column is:

  - `W`, test statistics.

  - `p_val`, p-values, which are obtained from two-sided Chi-square test using `W`.

  - `q_val`, adjusted p-values. Adjusted p-values are obtained by applying `p_adj_method` to `p_val`.

  - `diff_abn`, A logical vector. TRUE if the taxon has `q_val` less than `alpha`.

**Author(s)**

Huang Lin

**References**

Kaul A, Mandal S, Davidov O, Peddada SD (2017). "Analysis of microbiome data in the presence of excess zeros." *Frontiers in microbiology*, **8**, 2114.

Lin H, Peddada SD (2020). "Analysis of compositions of microbiomes with bias correction." *Nature communications*, **11**(1), 1–11.

**See Also**

[ancom](#) [ancombc2](#)

## Examples

```
#==========Build a TreeSummarizedExperiment Object from Scratch============
library(mia)

# microbial count table
otu_mat = matrix(sample(1:100, 100, replace = TRUE), nrow = 10, ncol = 10)
rownames(otu_mat) = paste0("taxon", 1:nrow(otu_mat))
colnames(otu_mat) = paste0("sample", 1:ncol(otu_mat))
assays = SimpleList(counts = otu_mat)

# sample metadata
smd = data.frame(group = sample(LETTERS[1:4], size = 10, replace = TRUE),
                 row.names = paste0("sample", 1:ncol(otu_mat)),
                 stringsAsFactors = FALSE)
smd = DataFrame(smd)

# taxonomy table
tax_tab = matrix(sample(letters, 70, replace = TRUE),
                 nrow = nrow(otu_mat), ncol = 7)
rownames(tax_tab) = rownames(otu_mat)
colnames(tax_tab) = c("Kingdom", "Phylum", "Class", "Order",
                      "Family", "Genus", "Species")
tax_tab = DataFrame(tax_tab)

# create TSE
tse = TreeSummarizedExperiment(assays = assays,
                               colData = smd,
                               rowData = tax_tab)

# convert TSE to phyloseq
pseq = makePhyloseqFromTreeSummarizedExperiment(tse)

#======================Run ANCOMBC Using a Real Data=====================
library(ANCOMBC)
data(atlas1006, package = "microbiome")
tse = mia::makeTreeSummarizedExperimentFromPhyloseq(atlas1006)

# subset to baseline
tse = tse[, tse$time == 0]

# run ancombc function
set.seed(123)
out = ancombc(data = tse, assay_name = "counts",
              tax_level = "Family", phyloseq = NULL,
              formula = "age + nationality + bmi_group",
              p_adj_method = "holm", prv_cut = 0.10, lib_cut = 1000,
              group = "bmi_group", struc_zero = TRUE, neg_lb = FALSE,
              tol = 1e-5, max_iter = 100, conserve = TRUE,
              alpha = 0.05, global = TRUE, n_cl = 1, verbose = TRUE)

res_prim = out$res
res_global = out$res_global
```

```
# to run ancombc using the phyloseq object
tse_alt = agglomerateByRank(tse, "Family")
pseq = makePhyloseqFromTreeSummarizedExperiment(tse_alt)
set.seed(123)
out = ancombc(data = NULL, assay_name = NULL,
              tax_level = "Family", phyloseq = pseq,
              formula = "age + nationality + bmi_group",
              p_adj_method = "holm", prv_cut = 0.10, lib_cut = 1000,
              group = "bmi_group", struc_zero = TRUE, neg_lb = FALSE,
              tol = 1e-5, max_iter = 100, conserve = TRUE,
              alpha = 0.05, global = TRUE, n_cl = 1, verbose = TRUE)
```

---

ancombc2                          *Analysis of Compositions of Microbiomes with Bias Correction 2*
                                  *(ANCOM-BC2)*

---

### Description

Determine taxa whose absolute abundances, per unit volume, of the ecosystem (e.g., gut) are significantly different with changes in the covariate of interest (e.g., group). The current version of ancombc2 function implements Analysis of Compositions of Microbiomes with Bias Correction (ANCOM-BC2) in cross-sectional and repeated measurements data. In addition to the two-group comparison, ANCOM-BC2 also supports testing for continuous covariates and multi-group comparisons, including the global test, pairwise directional test, Dunnett's type of test, and trend test.

### Usage

```
ancombc2(
  data,
  assay_name = "counts",
  tax_level = NULL,
  fix_formula,
  rand_formula = NULL,
  p_adj_method = "holm",
  pseudo = 0,
  pseudo_sens = TRUE,
  prv_cut = 0.1,
  lib_cut = 0,
  s0_perc = 0.05,
  group = NULL,
  struc_zero = FALSE,
  neg_lb = FALSE,
  alpha = 0.05,
  n_cl = 1,
  verbose = FALSE,
  global = FALSE,
```

```
    pairwise = FALSE,
    dunnet = FALSE,
    trend = FALSE,
    iter_control = list(tol = 0.01, max_iter = 20, verbose = FALSE),
    em_control = list(tol = 1e-05, max_iter = 100),
    lme_control = lme4::lmerControl(),
    mdfdr_control = list(fwer_ctrl_method = "holm", B = 100),
    trend_control = list(contrast = NULL, node = NULL, solver = "ECOS", B = 100)
)
```

## Arguments

| | |
|---|---|
| data | the input data. A phyloseq, SummarizedExperiment, or TreeSummarizedExperiment object, which consists of a feature table (microbial count table), a sample meta-data, a taxonomy table (optional), and a phylogenetic tree (optional). The row names of the metadata must match the sample names of the feature table, and the row names of the taxonomy table must match the taxon (feature) names of the feature table. See ?phyloseq::phyloseq, ?SummarizedExperiment::SummarizedExperiment, or ?TreeSummarizedExperiment::TreeSummarizedExperiment for more de-tails. It is highly recommended that the input data are in low taxonomic levels, such as OTU or species level, as the estimation of sampling fractions requires a large number of taxa. |
| assay_name | character. Name of the count table in the data object (only applicable if data ob-ject is a (Tree)SummarizedExperiment). Default is "counts". See ?SummarizedExperiment::assay for more details. |
| tax_level | character. The taxonomic level of interest. The input data can be analyzed at any taxonomic level without prior agglomeration. Note that tax_level must be a value from taxonomyRanks, which includes "Kingdom", "Phylum" "Class", "Order", "Family" "Genus" or "Species". See ?mia::taxonomyRanks for more details. Default is NULL, i.e., do not perform agglomeration, and the ANCOM-BC2 anlysis will be performed at the lowest taxonomic level of the input data. |
| fix_formula | the character string expresses how the microbial absolute abundances for each taxon depend on the fixed effects in metadata. |
| rand_formula | the character string expresses how the microbial absolute abundances for each taxon depend on the random effects in metadata. ANCOM-BC2 follows the lmerTest package in formulating the random effects. See ?lmerTest::lmer for more details. Default is NULL. |
| p_adj_method | character. method to adjust p-values. Default is "holm". Options include "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none". See ?stats::p.adjust for more details. |
| pseudo | numeric. Add pseudo-counts to the data. Default is 0 (no pseudo-count addi-tion). |
| pseudo_sens | logical. Whether to perform the sensitivity analysis to the pseudo-count addi-tion. Default is TRUE. See Details for a more comprehensive discussion on this sensitivity analysis. |
| prv_cut | a numerical fraction between 0 and 1. Taxa with prevalences less than prv_cut will be excluded in the analysis. For instance, suppose there are 100 samples, |

|                | if a taxon has nonzero counts presented in less than 10 samples, it will not be further analyzed. Default is 0.10. |
| --- | --- |
| lib_cut | a numerical threshold for filtering samples based on library sizes. Samples with library sizes less than lib_cut will be excluded in the analysis. Default is 0, i.e. do not discard any sample. |
| s0_perc | a numerical fraction between 0 and 1. Inspired by Significance Analysis of Microarrays (SAM) methodology, a small positive constant is added to the denominator of ANCOM-BC2 test statistic corresponding to each taxon to avoid the significance due to extremely small standard errors, especially for rare taxa. This small positive constant is chosen as s0_perc-th percentile of standard error values for each fixed effect. Default is 0.05 (5th percentile). |
| group | character. The name of the group variable in metadata. group should be discrete. Specifying group is required for detecting structural zeros and performing multi-group comparisons (global test, pairwise directional test, Dunnett's type of test, and trend test). Default is NULL. If the group of interest contains only two categories, leave it as NULL. |
| struc_zero | logical. Whether to detect structural zeros based on group. Default is FALSE. See Details for a more comprehensive discussion on structural zeros. |
| neg_lb | logical. Whether to classify a taxon as a structural zero using its asymptotic lower bound. Default is FALSE. |
| alpha | numeric. Level of significance. Default is 0.05. |
| n_cl | numeric. The number of nodes to be forked. For details, see ?parallel::makeCluster. Default is 1 (no parallel computing). |
| verbose | logical. Whether to generate verbose output during the ANCOM-BC2 fitting process. Default is FALSE. |
| global | logical. Whether to perform the global test. Default is FALSE. |
| pairwise | logical. Whether to perform the pairwise directional test. Default is FALSE. |
| dunnet | logical. Whether to perform the Dunnett's type of test. Default is FALSE. |
| trend | logical. Whether to perform trend test. Default is FALSE. |
| iter_control | a named list of control parameters for the iterative MLE or RMEL algorithm, including 1) tol: the iteration convergence tolerance (default is 1e-02), 2) max_iter: the maximum number of iterations (default is 20), and 3)verbose: whether to show the verbose output (default is FALSE). |
| em_control | a named list of control parameters for the E-M algorithm, including 1) tol: the iteration convergence tolerance (default is 1e-05) and 2) max_iter: the maximum number of iterations (default is 100). |
| lme_control | a list of control parameters for mixed model fitting. See ?lme4::lmerControl for details. |
| mdfdr_control | a named list of control parameters for mixed directional false discover rate (mdFDR), including 1) fwer_ctrl_method: family wise error (FWER) controlling procedure, such as "holm", "hochberg", "bonferroni", etc (default is "holm") and 2) B: the number of bootstrap samples (default is 100). Increase B will lead to a more accurate p-values. See Details for a more comprehensive discussion on mdFDR. |

trend_control    a named list of control parameters for the trend test, including 1) contrast: the list of contrast matrices for constructing inequalities, 2) node: the list of positions for the nodal parameter, 3) solver: a string indicating the solver to use (default is "ECOS"), and 4) B: the number of bootstrap samples (default is 100). Increase B will lead to a more accurate p-values. See vignette for the corresponding trend test examples.

### Details

A taxon is considered to have structural zeros in some (>=1) groups if it is completely (or nearly completely) missing in these groups. For instance, suppose there are three groups: g1, g2, and g3. If the counts of taxon A in g1 are 0 but nonzero in g2 and g3, then taxon A will be considered to contain structural zeros in g1. In this example, taxon A is declared to be differentially abundant between g1 and g2, g1 and g3, and consequently, it is globally differentially abundant with respect to this group variable. Such taxa are not further analyzed using ANCOM-BC2, but the results are summarized in the overall summary. For more details about the structural zeros, please go to the [ANCOM-II](#) paper. Setting neg_lb = TRUE indicates that you are using both criteria stated in section 3.2 of [ANCOM-II](#) to detect structural zeros; otherwise, the algorithm will only use the equation 1 in section 3.2 for declaring structural zeros. Generally, it is recommended to set neg_lb = TRUE when the sample size per group is relatively large (e.g. > 30).

Like other differential abundance analysis methods, ANCOM-BC2 log transforms the observed counts. However, to deal with zero counts, a pseudo-count is added before the log transformation. Several studies have shown that differential abundance results could be sensitive to the choice of pseudo-count ([Costea et al. (2014)](#); [Paulson, Bravo, and Pop (2014)](#)), resulting in an inflated false positive rate. To avoid such false positives, we conduct a sensitivity analysis and provide a sensitivity score for each taxon to determine if a particular taxon is sensitive to the choice of pseudo-count. The larger the score, the more likely the significant result is a false positive.

When performning pairwise directional (or Dunnett's type of) test, the mixed directional false discover rate (mdFDR) should be taken into account. The mdFDR is the combination of false discovery rate due to multiple testing, multiple pairwise comparisons, and directional tests within each pairwise comparison. For example, suppose we have five taxa and three experimental groups: g1, g2, and g3. Thus, we are performing five tests corresponding to five taxa. For each taxon, we are also conducting three pairwise comparisons (g1 vs. g2, g2 vs. g3, and g1 vs. g3). Within each pairwise comparison, we wish to determine if the abundance has increased or decreased or did not change (direction of the effect size). Errors could occur in each step. The overall false discovery rate is controlled by the mdFDR methodology we adopted from [Guo, Sarkar, and Peddada (2010)](#) and [Grandhi, Guo, and Peddada (2016)](#).

### Value

a list with components:

- feature_table, a data.frame of pre-processed (based on prv_cut and lib_cut) microbial count table.
- zero_ind, a logical data.frame with TRUE indicating the taxon is detected to contain structural zeros in some specific groups.
- samp_frac, a numeric vector of estimated sampling fractions in log scale (natural log).
- delta_em, estimated sample-specific biases through E-M algorithm.

- `delta_wls`, estimated sample-specific biases through weighted least squares (WLS) algorithm.
- `pseudo_sens_tab`, the results of sensitivity analysis for the pseudo-count addition.
- `res`, a `data.frame` containing ANCOM-BC2 primary result:
    - columns started with `lfc`: log fold changes obtained from the ANCOM-BC2 log-linear (natural log) model.
    - columns started with `se`: standard errors (SEs) of `lfc`.
    - columns started with `W`: test statistics. `W = lfc/se`.
    - columns started with `p`: p-values. P-values are obtained from two-sided Z-test using the test statistic `W`.
    - columns started with `q`: adjusted p-values. Adjusted p-values are obtained by applying `p_adj_method` to `p`.
    - columns started with `diff`: TRUE if the taxon is significant (has q less than `alpha`).
- `res_global`, a `data.frame` containing ANCOM-BC2 global test result for the variable specified in `group`, each column is:
    - `W`, test statistics.
    - `p_val`, p-values, which are obtained from two-sided Chi-square test using `W`.
    - `q_val`, adjusted p-values. Adjusted p-values are obtained by applying `p_adj_method` to `p_val`.
    - `diff_abn`, A logical vector. TRUE if the taxon has `q_val` less than `alpha`.
- `res_pair`, a `data.frame` containing ANCOM-BC2 pairwise directional test result for the variable specified in `group`:
    - columns started with `lfc`: log fold changes.
    - columns started with `se`: standard errors (SEs).
    - columns started with `W`: test statistics.
    - columns started with `p`: p-values.
    - columns started with `q`: adjusted p-values.
    - columns started with `diff`: TRUE if the taxon is significant (has q less than `alpha`).
- `res_dunn`, a `data.frame` containing ANCOM-BC2 Dunnett's type of test result for the variable specified in `group`:
    - columns started with `lfc`: log fold changes.
    - columns started with `se`: standard errors (SEs).
    - columns started with `W`: test statistics.
    - columns started with `p`: p-values.
    - columns started with `q`: adjusted p-values.
    - columns started with `diff`: TRUE if the taxon is significant (has q less than `alpha`).
- `res_trend`, a `data.frame` containing ANCOM-BC2 trend test result for the variable specified in `group`:
    - columns started with `lfc`: log fold changes.
    - columns started with `se`: standard errors (SEs).
    - `W`: test statistics.
    - `p_val`: p-values.
    - `q_val`: adjusted p-values.
    - `diff_abn`: TRUE if the taxon is significant (has q less than `alpha`).

## Author(s)

Huang Lin

## References

Kaul A, Mandal S, Davidov O, Peddada SD (2017). "Analysis of microbiome data in the presence of excess zeros." *Frontiers in microbiology*, **8**, 2114.

Lin H, Peddada SD (2020). "Analysis of compositions of microbiomes with bias correction." *Nature communications*, **11**(1), 1–11.

Tusher VG, Tibshirani R, Chu G (2001). "Significance analysis of microarrays applied to the ionizing radiation response." *Proceedings of the National Academy of Sciences*, **98**(9), 5116–5121.

Costea PI, Zeller G, Sunagawa S, Bork P (2014). "A fair comparison." *Nature methods*, **11**(4), 359–359.

Paulson JN, Bravo HC, Pop M (2014). "Reply to:" a fair comparison"." *Nature methods*, **11**(4), 359–360.

Guo W, Sarkar SK, Peddada SD (2010). "Controlling false discoveries in multidimensional directional decisions, with applications to gene expression data on ordered categories." *Biometrics*, **66**(2), 485–492.

Grandhi A, Guo W, Peddada SD (2016). "A multiple testing procedure for multi-dimensional pairwise comparisons with application to gene expression studies." *BMC bioinformatics*, **17**(1), 1–12.

## See Also

ancom ancombc

## Examples

```
#===========Build a TreeSummarizedExperiment Object from Scratch=============
library(mia)

# microbial count table
otu_mat = matrix(sample(1:100, 100, replace = TRUE), nrow = 10, ncol = 10)
rownames(otu_mat) = paste0("taxon", 1:nrow(otu_mat))
colnames(otu_mat) = paste0("sample", 1:ncol(otu_mat))
assays = SimpleList(counts = otu_mat)

# sample metadata
smd = data.frame(group = sample(LETTERS[1:4], size = 10, replace = TRUE),
                 row.names = paste0("sample", 1:ncol(otu_mat)),
                 stringsAsFactors = FALSE)
smd = DataFrame(smd)

# taxonomy table
tax_tab = matrix(sample(letters, 70, replace = TRUE),
                 nrow = nrow(otu_mat), ncol = 7)
rownames(tax_tab) = rownames(otu_mat)
colnames(tax_tab) = c("Kingdom", "Phylum", "Class", "Order",
                      "Family", "Genus", "Species")
```

```
       tax_tab = DataFrame(tax_tab)

       # create TSE
       tse = TreeSummarizedExperiment(assays = assays,
                                      colData = smd,
                                      rowData = tax_tab)

       # convert TSE to phyloseq
       pseq = makePhyloseqFromTreeSummarizedExperiment(tse)

       #=======================Run ANCOMBC2 Using a Real Data=====================
       library(ANCOMBC)
       data(dietswap, package = "microbiome")
       tse = mia::makeTreeSummarizedExperimentFromPhyloseq(dietswap)

       colData(tse)$bmi_group = factor(colData(tse)$bmi_group,
                                       levels = c("obese",
                                                  "overweight",
                                                  "lean"))

       set.seed(123)
       # Note that setting pseudo_sens = FALSE, max_iter = 1, and B = 1 is
       # only for the sake of speed
       # Set pseudo_sens = TRUE, and use default or larger values for max_iter and B
       # for better performance
       out = ancombc2(data = tse, assay_name = "counts", tax_level = "Phylum",
                      fix_formula = "nationality + timepoint + bmi_group",
                      rand_formula = "(timepoint | subject)",
                      p_adj_method = "holm", pseudo = 0, pseudo_sens = FALSE,
                      prv_cut = 0.10, lib_cut = 1000, s0_perc = 0.05,
                      group = "bmi_group", struc_zero = TRUE, neg_lb = TRUE,
                      alpha = 0.05, n_cl = 1, verbose = TRUE,
                      global = TRUE, pairwise = TRUE, dunnet = TRUE, trend = TRUE,
                      iter_control = list(tol = 1e-2, max_iter = 1, verbose = TRUE),
                      em_control = list(tol = 1e-5, max_iter = 1),
                      lme_control = lme4::lmerControl(),
                      mdfdr_control = list(fwer_ctrl_method = "holm", B = 1),
                      trend_control = list(contrast =
                                                     list(matrix(c(1, 0, -1, 1),
                                                                 nrow = 2,
                                                                 byrow = TRUE)),
                                           node = list(2),
                                           solver = "ECOS",
                                           B = 1))
       res_prim = out$res
       res_global = out$res_global
       res_pair = out$res_pair
       res_dunn = out$res_dunn
       res_trend = out$res_trend
```

---

| QMP | *Quantitative Microbiome Project data* |
|---|---|

---

### Description

The data containing quantitative microbiome count data of dimension 106 samples/subjects (in rows) and 91 OTUs (in columns). The raw dataset is pruned the taxa present less than 30 final dataset contains only healthy subjects from two cohorts: Study cohort and Disease cohort. For details, see https://doi.org/10.1038/nature24460.

### Usage

```
data(QMP)
```

### Format

The dataset in `matrix` format.

### Details

The dataset is also available via the SPRING R package https://github.com/GraceYoon/SPRING in `matrix` format.

### Value

Loads the dataset in R.

### Author(s)

Huang Lin <huanglinfrederick@gmail.com>

### References

Vanderputte et al. Nature. 551: 507-511, 2017. https://doi.org/10.1038/nature24460

---

| secom_dist | *Sparse estimation of distance correlations among microbiomes* |
|---|---|

---

### Description

Obtain the sparse correlation matrix for distance correlations between taxa.

## Usage

```
secom_dist(
  data,
  assay_name = "counts",
  tax_level = NULL,
  pseudo = 0,
  prv_cut = 0.5,
  lib_cut = 1000,
  corr_cut = 0.5,
  wins_quant = c(0.05, 0.95),
  R = 1000,
  thresh_hard = 0,
  max_p = 0.005,
  n_cl = 1
)
```

## Arguments

| | |
|---|---|
| data | a list of the input data. Each element of the list can be a phyloseq, SummarizedExperiment, or TreeSummarizedExperiment object, which consists of a feature table (microbial count table), a sample metadata, a taxonomy table (optional), and a phylogenetic tree (optional). The row names of the metadata must match the sample names of the feature table, and the row names of the taxonomy table must match the taxon (feature) names of the feature table. See ?phyloseq::phyloseq, ?SummarizedExperiment::SummarizedExperiment, or ?TreeSummarizedExperiment::TreeSummari for more details. It is highly recommended that the input data are in low taxonomic levels, such as OTU or species level, as the estimation of sampling fractions requires a large number of taxa. For multiple ecosystems, simply stack the data. For example, for two ecosystems, such as gut and tongue, specify the list of input data as data = list(gut = tse1, tongue = tse2). |
| assay_name | character. Name of the count table in the data object (only applicable if data object is a (Tree)SummarizedExperiment). Default is "counts". See ?SummarizedExperiment::assay for more details. |
| tax_level | character. The taxonomic level of interest. The input data can be agglomerated at different taxonomic levels based on your research interest. Default is NULL, i.e., do not perform agglomeration, and the SECOM anlysis will be performed at the lowest taxonomic level of the input data. |
| pseudo | numeric. Add pseudo-counts to the data. Default is 0 (no pseudo-counts). |
| prv_cut | a numerical fraction between 0 and 1. Taxa with prevalences less than prv_cut will be excluded in the analysis. For instance, suppose there are 100 samples, if a taxon has nonzero counts presented in less than 10 samples, it will not be further analyzed. Default is 0.50. |
| lib_cut | a numerical threshold for filtering samples based on library sizes. Samples with library sizes less than lib_cut will be excluded in the analysis. Default is 1000. |
| corr_cut | numeric. To prevent false positives due to taxa with small variances, taxa with Pearson correlation coefficients greater than corr_cut with the estimated sample- |

|              | specific bias will be flagged. The pairwise correlation coefficient between flagged taxa will be set to 0s. Default is 0.5. |
| :--- | :--- |
| wins_quant   | a numeric vector of probabilities with values between 0 and 1. Replace extreme values in the abundance data with less extreme values. Default is `c(0.05, 0.95)`. For details, see `?DescTools::Winsorize`. |
| R            | numeric. The number of replicates in calculating the p-value for distance correlation. For details, see `?energy::dcor.test`. Default is 1000. |
| thresh_hard  | Numeric. Set a hard threshold for the correlation matrix. Pairwise distance correlation less than or equal to `thresh_hard` will be set to 0. Default is 0 (No ad-hoc hard thresholding). |
| max_p        | numeric. Obtain the sparse correlation matrix by p-value filtering. Pairwise correlation coefficient with p-value greater than `max_p` will be set to 0. Default is 0.005. |
| n_cl         | numeric. The number of nodes to be forked. For details, see `?parallel::makeCluster`. Default is 1 (no parallel computing). |

## Details

The distance correlation, which is a measure of dependence between two random variables, can be used to quantify any dependence, whether linear, monotonic, non-monotonic or nonlinear relationships.

## Value

a `list` with components:

- `s_diff_hat`, a numeric vector of estimated sample-specific biases.

- `y_hat`, a matrix of bias-corrected abundances

- `mat_cooccur`, a matrix of taxon-taxon co-occurrence pattern. The number in each cell represents the number of complete (nonzero) samples for the corresponding pair of taxa.

- `dcorr`, the sample distance correlation matrix computed using the bias-corrected abundances `y_hat`.

- `dcorr_p`, the p-value matrix corresponding to the sample distance correlation matrix `dcorr`.

- `dcorr_fl`, the sparse correlation matrix obtained by p-value filtering based on the cutoff specified in `max_p`.

## Author(s)

Huang Lin

## See Also

[secom_linear](#)

## Examples

```
library(ANCOMBC)
data(dietswap, package = "microbiome")
tse = mia::makeTreeSummarizedExperimentFromPhyloseq(dietswap)

# subset to baseline
tse = tse[, tse$timepoint == 1]

set.seed(123)
res_dist = secom_dist(data = list(tse), assay_name = "counts",
                      tax_level = "Phylum", pseudo = 0,
                      prv_cut = 0.5, lib_cut = 1000, corr_cut = 0.5,
                      wins_quant = c(0.05, 0.95), R = 1000,
                      thresh_hard = 0.3, max_p = 0.005, n_cl = 2)

dcorr_fl = res_dist$dcorr_fl
```

---

secom_linear                 *Sparse estimation of linear correlations among microbiomes*

---

## Description

Obtain the sparse correlation matrix for linear correlations between taxa. The current version of
secom_linear function supports either of the three correlation coefficients: Pearson, Spearman,
and Kendall's $\tau$.

## Usage

```
secom_linear(
  data,
  assay_name = "counts",
  tax_level = NULL,
  pseudo = 0,
  prv_cut = 0.5,
  lib_cut = 1000,
  corr_cut = 0.5,
  wins_quant = c(0.05, 0.95),
  method = c("pearson", "kendall", "spearman"),
  soft = FALSE,
  thresh_len = 100,
  n_cv = 10,
  thresh_hard = 0,
  max_p = 0.005,
  n_cl = 1
)
```

**Arguments**

| | |
|---|---|
| data | a list of the input data. Each element of the list can be a phyloseq, SummarizedExperiment, or TreeSummarizedExperiment object, which consists of a feature table (microbial count table), a sample metadata, a taxonomy table (optional), and a phylogenetic tree (optional). The row names of the metadata must match the sample names of the feature table, and the row names of the taxonomy table must match the taxon (feature) names of the feature table. See ?phyloseq::phyloseq, ?SummarizedExperiment::SummarizedExperiment, or ?TreeSummarizedExperiment::TreeSummari for more details. It is highly recommended that the input data are in low taxonomic levels, such as OTU or species level, as the estimation of sampling fractions requires a large number of taxa. For multiple ecosystems, simply stack the data. For example, for two ecosystems, such as gut and tongue, specify the list of input data as data = list(gut = tse1, tongue = tse2). |
| assay_name | character. Name of the count table in the data object (only applicable if data object is a (Tree)SummarizedExperiment). Default is "counts". See ?SummarizedExperiment::assay for more details. |
| tax_level | character. The taxonomic level of interest. The input data can be agglomerated at different taxonomic levels based on your research interest. Default is NULL, i.e., do not perform agglomeration, and the SECOM anlysis will be performed at the lowest taxonomic level of the input data. |
| pseudo | numeric. Add pseudo-counts to the data. Default is 0 (no pseudo-counts). |
| prv_cut | a numerical fraction between 0 and 1. Taxa with prevalences less than prv_cut will be excluded in the analysis. For instance, suppose there are 100 samples, if a taxon has nonzero counts presented in less than 10 samples, it will not be further analyzed. Default is 0.50. |
| lib_cut | a numerical threshold for filtering samples based on library sizes. Samples with library sizes less than lib_cut will be excluded in the analysis. Default is 1000. |
| corr_cut | numeric. To prevent false positives due to taxa with small variances, taxa with Pearson correlation coefficients greater than corr_cut with the estimated sample-specific bias will be flagged. The pairwise correlation coefficient between flagged taxa will be set to 0s. Default is 0.5. |
| wins_quant | a numeric vector of probabilities with values between 0 and 1. Replace extreme values in the abundance data with less extreme values. Default is c(0.05, 0.95). For details, see ?DescTools::Winsorize. |
| method | character. It indicates which correlation coefficient is to be computed. It can be either "pearson" or "spearman". |
| soft | logical. TRUE indicates that soft thresholding is applied to achieve the sparsity of the correlation matrix. FALSE indicates that hard thresholding is applied to achieve the sparsity of the correlation matrix. Default is FALSE. |
| thresh_len | numeric. Grid-search is implemented to find the optimal values over thresh_len thresholds for the thresholding operator. Default is 100. |
| n_cv | numeric. The fold number in cross validation. Default is 10 (10-fold cross validation). |

| thresh_hard | Numeric. Set a hard threshold for the correlation matrix. Pairwise correlation coefficient (in its absolute value) less than or equal to thresh_hard will be set to 0. Default is 0 (No ad-hoc hard thresholding). |
|---|---|
| max_p | numeric. Obtain the sparse correlation matrix by p-value filtering. Pairwise correlation coefficient with p-value greater than max_p will be set to 0. Default is 0.005. |
| n_cl | numeric. The number of nodes to be forked. For details, see ?parallel::makeCluster. Default is 1 (no parallel computing). |

### Value

a list with components:

- s_diff_hat, a numeric vector of estimated sample-specific biases.

- y_hat, a matrix of bias-corrected abundances

- cv_error, a numeric vector of cross-validation error estimates, which are the Frobenius norm differences between correlation matrices using training set and validation set, respectively.

- thresh_grid, a numeric vector of thresholds in the cross-validation.

- thresh_opt, numeric. The optimal threshold through cross-validation.

- mat_cooccur, a matrix of taxon-taxon co-occurrence pattern. The number in each cell represents the number of complete (nonzero) samples for the corresponding pair of taxa.

- corr, the sample correlation matrix (using the measure specified in method) computed using the bias-corrected abundances y_hat.

- corr_p, the p-value matrix corresponding to the sample correlation matrix corr.

- corr_th, the sparse correlation matrix obtained by thresholding based on the method specified in soft.

- corr_fl, the sparse correlation matrix obtained by p-value filtering based on the cutoff specified in max_p.

### Author(s)

Huang Lin

### See Also

[secom_dist](secom_dist)

### Examples

```
library(ANCOMBC)
data(dietswap, package = "microbiome")
tse = mia::makeTreeSummarizedExperimentFromPhyloseq(dietswap)

# subset to baseline
tse = tse[, tse$timepoint == 1]

set.seed(123)
```

```
res_linear = secom_linear(data = list(tse), assay_name = "counts",
                          tax_level = "Phylum", pseudo = 0,
                          prv_cut = 0.5, lib_cut = 1000, corr_cut = 0.5,
                          wins_quant = c(0.05, 0.95), method = "pearson",
                          soft = FALSE, thresh_len = 20, n_cv = 10,
                          thresh_hard = 0.3, max_p = 0.005, n_cl = 2)

corr_th = res_linear$corr_th
corr_fl = res_linear$corr_fl
```

---

| | |
|---|---|
| sim_plnm | *Simulate Microbial Absolute Abundance Data by Poisson lognormal (PLN) model Based on a Real Dataset* |

---

## Description

Generate microbial absolute abundances using the Poisson lognormal (PLN) model based on the mechanism described in the LDM paper (supplementary text S2).

## Usage

```
sim_plnm(abn_table, taxa_are_rows = TRUE, prv_cut = 0.1, n, lib_mean, disp)
```

## Arguments

| | |
|---|---|
| abn_table | the input microbial count table. It is used to obtain the estimated variance-covariance matrix, can be in either `matrix` or `data.frame` format. |
| taxa_are_rows | logical. TRUE if the input dataset has rows represent taxa. Default is TRUE. |
| prv_cut | a numerical fraction between 0 and 1. Taxa with prevalences less than `prv_cut` will be excluded in the analysis. For instance, suppose there are 100 samples, if a taxon has nonzero counts presented in less than 10 samples, it will not be further analyzed. Default is 0.10. |
| n | numeric. The desired sample size for the simulated data. |
| lib_mean | numeric. Mean of the library size. Library sizes are generated from the negative binomial distribution with parameters `lib_mean` and `disp`. For details, see `?rnbinom`. |
| disp | numeric. The dispersion parameter for the library size. For details, see `?rnbinom`. |

## Details

The PLN model relates the abundance vector with a Gaussian latent vector. Because of the presence of a latent layer, the PLN model displays a larger variance than the Poisson model (over-dispersion). Also, the covariance (correlation) between abundances has the same sign as the covariance (correlation) between the corresponding latent variables. This property gives enormous flexibility in modeling the variance-covariance structure of microbial abundances since it is easy to specify different variance-covariance matrices in the multivariate Gaussian distribution.

However, instead of manually specifying the variance-covariance matrix, we choose to estimate the variance-covariance matrix from a real dataset, which will make the simulated data more resemble real data.

## Value

a `matrix` of microbial absolute abundances, where taxa are in rows and samples are in columns.

## Author(s)

Huang Lin

## References

Hu Y, Satten GA (2020). "Testing hypotheses about the microbiome using the linear decomposition model (LDM)." *Bioinformatics*, **36**(14), 4106–4115.

## Examples

```
library(ANCOMBC)
data(QMP)
abn_data = sim_plnm(abn_table = QMP, taxa_are_rows = FALSE, prv_cut = 0.05,
                    n = 100, lib_mean = 1e8, disp = 0.5)
rownames(abn_data) = paste0("Taxon", seq_len(nrow(abn_data)))
colnames(abn_data) = paste0("Sample", seq_len(ncol(abn_data)))
```

# Index