

Package ‘SAFEMCN’

May 28, 2026

Type Package

Title Network Topology Parameter Analysis with Rarefaction

Version 1.0.0

Description Calculate network topology parameters from Operational Taxonomic Unit (OTU) tables with customizable correlation thresholds, parallel processing options, and visualization capabilities including trend fitting, prediction of future sample sizes, and lag-1 autocorrelation (AR1) analysis. Methods are based on co-occurrence network construction via correlation thresholds and graph-theoretic metrics computed with 'igraph'.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 3.5.0)

Imports igraph, Hmisc, parallel, ggplot2, grDevices, graphics, rlang, stats, utils

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

RoxygenNote 7.3.3

NeedsCompilation no

Author Xiaotong Xu [aut, cre],
Yue Zheng [cph]

Maintainer Xiaotong Xu <xxt19992014@163.com>

Repository CRAN

Date/Publication 2026-05-28 13:00:05 UTC

Contents

| | |
|------------------------------------|----------|
| analyze_network_topology | 2 |
| Index | 5 |

 analyze_network_topology

Analyze Network Topology Parameters with Rarefaction

Description

Calculate network topology parameters from OTU tables with customizable correlation thresholds, parallel processing options, and visualization capabilities including topology parameter plots, exponential formula fitting, prediction of future sample sizes, and AR1 analysis.

Usage

```
analyze_network_topology(
  otu_file,
  r_threshold = 0.6,
  p_threshold = 0.05,
  cor_method = "spearman",
  start_size = 5,
  end_size = NULL,
  step_size = 1,
  replicates = 50,
  use_parallel = FALSE,
  output_prefix = "network_parameters",
  plot_topology = FALSE,
  plot_start_size = 11,
  fit_formula = FALSE,
  fit_start_size = 11,
  predict_end_size = NULL,
  run_ar1 = FALSE,
  ar1_input_file = NULL,
  ar1_windows = c(5, 10, 15)
)
```

Arguments

| | |
|--------------|---|
| otu_file | OTU table file name (full path) |
| r_threshold | Correlation coefficient threshold (default: 0.6) |
| p_threshold | P-value threshold (default: 0.05) |
| cor_method | Correlation method: "spearman" or "pearson" (default: "spearman") |
| start_size | Starting sample size (default: 5) |
| end_size | Ending sample size (default: NULL, uses total samples) |
| step_size | Step size for sample sizes (default: 1) |
| replicates | Number of replicates (default: 50) |
| use_parallel | Use parallel processing (default: FALSE) |

```

output_prefix  Output file prefix (default: "network_parameters")
plot_topology  Whether to plot topology parameters vs sample size (default: FALSE)
plot_start_size
                Starting sample size for topology plots (default: 11)
fit_formula    Whether to perform exponential formula fitting (default: FALSE)
fit_start_size Starting sample size for formula fitting (default: 11)
predict_end_size
                End sample size for prediction (default: NULL, no prediction)
run_ar1        Whether to run AR1 analysis (default: FALSE)
ar1_input_file Input file name for AR1 analysis (default: NULL, auto-select)
ar1_windows    Sliding window sizes for AR1 (default: c(5, 10, 15))

```

Value

A list containing raw results, summary statistics, fit parameters, and predictions

Examples

```

# Create a small toy dataset (8 OTUs x 10 samples)
set.seed(42)
otu_data <- data.frame(
  matrix(rpois(80, lambda = 10), nrow = 8, ncol = 10)
)
rownames(otu_data) <- paste0("OTU", 1:8)
colnames(otu_data) <- paste0("Sample", 1:10)
tmp_file <- tempfile(fileext = ".csv")
write.csv(otu_data, tmp_file)

results <- analyze_network_topology(
  otu_file = tmp_file,
  r_threshold = 0.4,
  p_threshold = 0.05,
  start_size = 5,
  end_size = 7,
  step_size = 1,
  replicates = 2,
  use_parallel = FALSE
)
# Check structure of returned list
names(results)
head(results$summary)

# Larger run with fitting and prediction (takes longer)
set.seed(100)
otu_big <- data.frame(
  matrix(rpois(200, lambda = 10), nrow = 10, ncol = 20)
)
rownames(otu_big) <- paste0("OTU", 1:10)

```

```
colnames(otu_big) <- paste0("Sample", 1:20)
tmp_file2 <- tempfile(fileext = ".csv")
write.csv(otu_big, tmp_file2)

results2 <- analyze_network_topology(
  otu_file = tmp_file2,
  r_threshold = 0.4,
  start_size = 5,
  end_size = 18,
  step_size = 1,
  replicates = 5,
  fit_formula = TRUE,
  fit_start_size = 5,
  predict_end_size = 25
)
str(results2$fit_parameters)
```

Index

`analyze_network_topology`, [2](#)