Package 'decontam'

November 12, 2026

Type Package

Title Identify Contaminants in Marker-gene and Metagenomics Sequencing Data

Version 1.33.0

Date 2021-02-19

Maintainer Benjamin Callahan <benjamin.j.callahan@gmail.com>

Description Simple statistical identification of contaminating sequence features in marker-gene or metagenomics data. Works on any kind of feature derived from environmental sequencing data (e.g. ASVs, OTUs, taxonomic groups, MAGs,...). Requires DNA quantitation data or sequenced negative control samples.

Depends R (>= 3.4.1), methods (>= 3.4.1)

Imports ggplot2 (>= 2.1.0), reshape2 (>= 1.4.1), stats

Suggests BiocStyle, knitr, rmarkdown, phyloseq

License Artistic-2.0

Encoding UTF-8

VignetteBuilder knitr

biocViews ImmunoOncology, Microbiome, Sequencing, Classification, Metagenomics

URL https://github.com/benjjneb/decontam

BugReports https://github.com/benjjneb/decontam/issues

LazyData true

RoxygenNote 7.1.1

git_url https://git.bioconductor.org/packages/decontam

git_branch devel

git_last_commit afdb4e8

git_last_commit_date 2026-04-28

Repository Bioconductor 3.24

Date/Publication 2026-11-12

Author Benjamin Callahan [aut, cre],
Nicole Marie Davis [aut],
Felix G.M. Ernst [ctb] (ORCID: <https://orcid.org/0000-0001-5064-0928>)

Contents

isContaminant 2
isNotContaminant 4
plot_condition 5
plot_frequency 6

Index **8**

`isContaminant` *Identify contaminant sequences.*

Description

The frequency of each sequence (or OTU) in the input feature table as a function of the concentration of amplified DNA in each sample is used to identify contaminant sequences.

Usage

```
isContaminant(seqtab, ...)

## S4 method for signature 'ANY'
isContaminant(
  seqtab,
  conc = NULL,
  neg = NULL,
  method = c("auto", "frequency", "prevalence", "combined", "minimum", "either",
    "both"),
  batch = NULL,
  batch.combine = c("minimum", "product", "fisher"),
  threshold = 0.1,
  normalize = TRUE,
  detailed = TRUE
)
```

Arguments

`seqtab`	(Required). `Integer matrix` or `phyloseq` object. A feature table recording the observed abundances of each sequence variant (or OTU) in each sample. Rows should correspond to samples, and columns to sequences (or OTUs). If a phyloseq object is provided, the otu-table component will be extracted.
`...`	Not used currently
`conc`	(Optional). `numeric`. Required if performing frequency-based testing. A quantitative measure of the concentration of amplified DNA in each sample prior to sequencing. All values must be greater than zero. Zero is assumed to represent the complete absence of DNA. If `seqtab` was prodivded as a phyloseq object, the name of the appropriate sample-variable in that phyloseq object can be provided.
`neg`	(Optional). `logical`. Required if performing prevalence-based testing. TRUE if sample is a negative control, and FALSE if not (NA entries are not included in the testing). Extraction controls give the best results. If `seqtab` was provided as

a phyloseq object, the name of the appropriate sample-variable in that phyloseq object can be provided.

`method` (Optional). `character`. The method used to test for contaminants.

auto (Default). frequency, prevalence or combined will be automatically selected based on whether just `conc`, just `neg`, or both were provided.

frequency Contaminants are identified by frequency that varies inversely with sample DNA concentration.

prevalence Contaminants are identified by increased prevalence in negative controls.

combined The frequency and prevalence probabilities are combined with Fisher's method and used to identify contaminants.

minimum The minimum of the frequency and prevalence probabilities is used to identify contaminants.

either Contaminants are called if identified by either the frequency or prevalance methods.

both Contaminants are called if identified by both the frequency and prevalance methods.

`batch` (Optional). `factor`, or any type coercible to a `factor`. Default NULL. If provided, should be a vector of length equal to the number of input samples which specifies which batch each sample belongs to (eg. sequencing run). Contaminants identification will be performed independently within each batch. If `seqtab` was provided as a phyloseq object, the name of the appropriate sample-variable in that phyloseq object can be provided.

`batch.combine` (Optional). Default "minimum". For each input sequence variant (or OTU) the probabilities calculated in each batch are combined into a single probability that is compared to 'codethreshold' to classify contaminants. Valid values: "minimum", "product", "fisher".

`threshold` (Optional). Default `0.1`. The probability threshold below which (strictly less than) the null-hypothesis (not a contaminant) should be rejected in favor of the alternate hypothesis (contaminant). A length-two vector can be provided when using the `either` or `both` methods: the first value is the threshold for the frequency test and the second for the prevalence test.

`normalize` (Optional). Default TRUE. If TRUE, the input `seqtab` is normalized so that each row sums to 1 (converted to frequency). If FALSE, no normalization is performed (the data should already be frequencies or counts from equal-depth samples).

`detailed` (Optional). Default TRUE. If TRUE, the return value is a `data.frame` containing diagnostic information on the contaminant decision. If FALSE, the return value is a `logical` vector containing the binary contaminant classifications.

Value

If `detailed=TRUE` a `data.frame` with classification information. If `detailed=FALSE` a `logical` vector is returned, with TRUE indicating contaminants.

Examples

```
st <- readRDS(system.file("extdata", "st.rds", package="decontam"))
# conc should be positive and non-zero
conc <- c(6413, 3581.0, 5375, 4107, 4291, 4260, 4171, 2765, 33, 48)
```

```
neg <- c(FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, TRUE, TRUE)
# Use frequency or frequency and prevalence to identify contaminants
isContaminant(st, conc=conc, method="frequency", threshold=0.2)
isContaminant(st, conc=conc, neg=neg, method="both", threshold=c(0.1,0.5))
```

`isNotContaminant` *Identify non-contaminant sequences.*

Description

The prevalence of each sequence (or OTU) in the input feature table across samples and negative controls is used to identify non-contaminant sequences. Note that the null hypothesis here is that sequences **are** contaminants. This function is intended for use on low-biomass samples in which a large proportion of the sequences are likely to be contaminants.

Usage

```
isNotContaminant(seqtab, ...)

## S4 method for signature 'ANY'
isNotContaminant(
  seqtab,
  neg = NULL,
  method = "prevalence",
  threshold = 0.5,
  normalize = TRUE,
  detailed = FALSE
)
```

Arguments

`seqtab`	(Required). Integer matrix. A feature table recording the observed abundances of each sequence (or OTU) in each sample. Rows should correspond to samples, and columns to sequences (or OTUs).
`...`	Not used currently
`neg`	(Required). `logical` The negative control samples. Extraction controls give the best results.
`method`	(Optional). Default "prevalence". The method used to test for contaminants. Currently the only method supported is prevalence. prevalence: Contaminants are identified by increased prevalence in negative controls.
`threshold`	(Optional). Default `0.5`. The probability threshold below which (strictly less than) the null-hypothesis (a contaminant) should be rejected in favor of the alternate hypothesis (not a contaminant).
`normalize`	(Optional). Default TRUE. If TRUE, the input `seqtab` is normalized so that each row sums to 1 (converted to frequency). If FALSE, no normalization is performed (the data should already be frequencies or counts from equal-depth samples).
`detailed`	(Optional). Default FALSE. If TRUE, the return value is a `data.frame` containing diagnostic information on the non-contaminant decision. If FALSE, the return value is a `logical` vector containing the non-contaminant decisions.

Value

If `detailed=FALSE` a `logical` vector is returned, with TRUE indicating non-contaminants. If `detailed=TRUE` a `data.frame` is returned instead.

Examples

```
st <- readRDS(system.file("extdata", "st.rds", package="decontam"))
samdf <- readRDS(system.file("extdata", "samdf.rds", package="decontam"))
isNotContaminant(st, samdf$quant_reading, threshold=0.05)
```

`plot_condition` *Plot DNA concentrations as a function of experimental conditions.*

Description

Plots DNA concentration as a function of experimental conditions. This function is intended as a convenient exploration of potential covariation between DNA concentrations and conditions that could influence the community composition, as this could lead to higher rates of false-positive contaminant identifications.

Usage

```
plot_condition(seqtab, condition, conc, batch = NULL, log = FALSE)
```

Arguments

`seqtab`	(Required). `Integer matrix` or `phyloseq` object. A feature table recording the observed abundances of each sequence feature (e.g. OTUs or ASVs or or genus or ortholog or...) in each sample. Rows should correspond to samples, and columns to sequences (or OTUs). If a phyloseq object is provided, the otu-table component will be extracted.
`condition`	(Required). `numeric` or any type coercible to a `factor`. Default NULL. If provided, should be a vector of length equal to the number of input samples which specifies the experimental condition of interest for each sample (e.g. pH). If `seqtab` was provided as a phyloseq object, the name of the appropriate sample-variable in that phyloseq object can be provided.
`conc`	(Required). `numeric`. A quantitative measure of the concentration of amplified DNA in each sample prior to sequencing. All values must be greater than zero. Zero is assumed to represent the complete absence of DNA. If `seqtab` was provided as a `phyloseq` object, the name of the sample variable in the `phyloseq` object can be provided.
`batch`	(Optional). `factor`, or any type coercible to a `factor`. Default NULL. If provided, should be a vector of length equal to the number of input samples which specifies which batch each sample belongs to (eg. sequencing run). Contaminants identification will be performed independently within each batch. If `seqtab` was provided as a phyloseq object, the name of the appropriate sample-variable in that phyloseq object can be provided.
`log`	(Optional). `logical`. Default TRUE. If TRUE, the axes are log10-scaled.

Examples

```
# MUC is a phyloseq object, MUC.conc is the vector of sample concentrations
MUC <- readRDS(system.file("extdata", "MUClite.rds", package="decontam"))
MUC.conc <- readRDS(system.file("extdata", "MUCconc.rds", package="decontam"))
plot_condition(MUC, "Habitat", MUC.conc)
# Plot against random quantitative variable
plot_condition(MUC, runif(length(MUC.conc)), MUC.conc, log=TRUE)
```

---

plot_frequency *Plot frequencies as a function of input DNA concentration*

Description

Plots the frequencies of selected sequence features vs. each sample's DNA concentration.

Usage

```
plot_frequency(
  seqtab,
  taxa,
  conc,
  neg = NULL,
  normalize = TRUE,
  showModels = TRUE,
  log = TRUE,
  facet = TRUE
)
```

Arguments

seqtab	(Required). Integer matrix or phyloseq object. A feature table recording the observed abundances of each sequence feature (e.g. OTUs or ASVs or or genus or ortholog or...) in each sample. Rows should correspond to samples, and columns to sequences (or OTUs). If a phyloseq object is provided, the otu-table component will be extracted.
taxa	(Required). character. The names of the sequence features to include in this plot. Should match colnames(setab) if a matrix was provided, or taxa_names(seqtab) if a phyloseq object was provided.
conc	(Required). numeric. A quantitative measure of the concentration of amplified DNA in each sample prior to sequencing. All values must be greater than zero. Zero is assumed to represent the complete absence of DNA. If seqtab was provided as a phyloseq object, the name of the sample variable in the phyloseq object can be provided.
neg	(Optional). logical. Default NULL. TRUE if sample is a negative control, and FALSE if not. If seqtab was provided as a phyloseq object, the name of the appropriate sample-variable in that phyloseq object can be provided. NULL indicates no samples should be condired negative controls.

normalize	(Optional). logical. Default TRUE. If TRUE, the input seqtab is normalized so that each row sums to 1 (converted to frequency). If FALSE, no normalization is performed (the data should already be frequencies or counts from equal-depth samples).
showModels	(Optional). logical. Default TRUE. If TRUE, the contaminant (red, dashed line) and non-contaminant (black, solid line) models are shown in the plot.
log	(Optional). logical. Default TRUE. If TRUE, the axes are log10-scaled.
facet	(Optional). logical. Default TRUE. If TRUE, multiple sequence features will be plotted in separate facets.

Value

A ggplot2 object. Will be rendered to default device if printed, or can be stored and further modified. See ggsave for additional options.

Examples

```
# MUC is a phyloseq object, MUC.conc is the vector of sample concentrations
MUC <- readRDS(system.file("extdata", "MUClite.rds", package="decontam"))
MUC.conc <- readRDS(system.file("extdata", "MUCconc.rds", package="decontam"))
plot_frequency(MUC, "Seq1", conc=MUC.conc)
# The concentration can also be reference directly as the quant_reading sample variable in MUC
plot_frequency(MUC, "Seq1", conc="quant_reading")
plot_frequency(MUC, c("Seq1", "Seq10", "Seq33"), conc="quant_reading", log=FALSE)
```

Index

ggplot, 7
ggsave, 7

isContaminant, 2
isContaminant,ANY-method
        (isContaminant), 2
isNotContaminant, 4
isNotContaminant,ANY-method
        (isNotContaminant), 4

plot_condition, 5
plot_frequency, 6
print, 7