Package 'IFAA'

March 10, 2022

Title Robust Inference for Absolute Abundance in Microbiome Analysis

Version 1.0.5

Description A robust approach to make inference on the association of covariates with the absolute abundance (AA) of 'microbiome' in an ecosystem. It can be also directly applied to relative abundance (RA) data to make inference on AA (even if AA data is not available) because the ratio of two RA is equal ratio of their AA. This algorithm can estimate and test the associations of interest while adjusting for potential 'confounders'. The estimates of this method have easy interpretation like a typical regression analysis. High-dimensional covariates are handled with regularization and it is implemented by parallel computing. False discovery rate is automatically controlled by this approach.

License GNU General Public License version 2

Encoding UTF-8

URL https://github.com/gitlzg/IFAA,

https://arxiv.org/abs/1909.10101v3,

https://link.springer.com/article/10.1007/s12561-018-9219-2

LazyData true

RoxygenNote 7.1.2

Depends R (¿= 3.6.0),

Imports qlcMatrix (¿= 0.9.7), methods (¿= 3.3.0), mathjaxr (¿= 1.0-1), expm (¿= 0.999-3), foreach (¿= 1.4.3), rlecuyer (¿= 0.3-3), Matrix (¿= 1.4-0), HDCI (¿= 1.0-2), parallel (¿= 3.3.0), doParallel (¿= 1.0.11), future (¿= 1.12.0), glmnet, stats

RdMacros mathjaxr

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Quran Wu [aut],
Zhigang Li [aut, cre]

Maintainer Zhigang Li <zhigang.li@ufl.edu>

R topics documented:

dataC 2
dataM 2
IFAA 3
MZILN 6

Index **9**

`dataC` *Sample covariates data*

Description

A dataset ontains 3 covariates.

Usage

```
dataC
```

Format

A data frame with 40 rows and 4 variables:

`dataM` *Sample microbiome data*

Description

A dataset contains 60 taxa with absolute abundances and these are gut microbiome.

Usage

```
dataM
```

Format

A data frame with 40 rows and 61 variables:

IFAA *Robust association identification and inference for absolute abundance in microbiome analyses*

Description

Make inference on the association of microbiome with covariates

Usage

```
IFAA(
  MicrobData,
  CovData,
  linkIDname,
  testCov = NULL,
  ctrlCov = NULL,
  testMany = TRUE,
  ctrlMany = FALSE,
  nRef = 40,
  nRefMaxForEsti = 2,
  refTaxa = NULL,
  adjust_method = "BY",
  fdrRate = 0.15,
  paraJobs = NULL,
  bootB = 500,
  standardize = FALSE,
  sequentialRun = FALSE,
  refReadsThresh = 0.2,
  taxkeepThresh = 1,
  SDThresh = 0.05,
  SDquantilThresh = 0,
  balanceCut = 0.2,
  seed = 1
)
```

Arguments

`MicrobData` Microbiome data matrix containing microbiome absolute abundance or relative abundance with each row per sample and each column per taxon/OTU/ASV (or any other unit). It should contain an id variable to be linked with the id variable in the covariates data: `CovData`. This argument can take directory path. For example, `MicrobData="C://...//microbiomeData.tsv"`.

`CovData` Covariates data matrix containing covariates and confounders with each row per sample and each column per variable. It should also contain an id variable to be linked with the id variable in the microbiome data: `MicrobData`. This argument can take directory path. For example, `CovData = "C://...//covariatesData.tsv"`.

`linkIDname` The common variable name of the id variable in both `MicrobData` and `CovData`. The two data sets will be merged by this id variable.

`testCov` Covariates that are of primary interest for testing and estimating the associations. It corresponds to X_i in the equation. Default is `NULL` which means all covariates are `testCov`.

`ctrlCov` Potential confounders that will be adjusted in the model. It corresponds to W_i in the equation. Default is `NULL` which means all covariates except those in `testCov` are adjusted as confounders.

`testMany` This takes logical value `TRUE` or `FALSE`. If `TRUE`, the `testCov` will contain all the variables in `CovData` provided `testCov` is set to be `NULL`. The default value is `TRUE` which does not do anything if `testCov` is not `NULL`.

`ctrlMany` This takes logical value `TRUE` or `FALSE`. If `TRUE`, all variables except `testCov` are considered as control covariates provided `ctrlCov` is set to be `NULL`. The default value is `FALSE`.

`nRef` The number of randomly picked reference taxa used in phase 1. Default number is `40`.

`nRefMaxForEsti` The maximum number of final reference taxa used in phase 2. The default is `2`.

`refTaxa` A vector of taxa or OTU or ASV names. These are reference taxa specified by the user to be used in phase 1. If the number of reference taxa is less than 'nRef', the algorithm will randomly pick extra reference taxa to make up 'nRef'. The default is `NULL` since the algorithm will pick reference taxa randomly.

`adjust_method` The adjusting method for p value adjustment. Default is "BY" for dependent FDR adjustment. It can take any adjustment method for p.adjust function in R.

`fdrRate` The false discovery rate for identifying taxa/OTU/ASV associated with `testCov`. Default is `0.15`.

`paraJobs` If `sequentialRun` is `FALSE`, this specifies the number of parallel jobs that will be registered to run the algorithm. If specified as `NULL`, it will automatically detect the cores to decide the number of parallel jobs. Default is `NULL`.

`bootB` Number of bootstrap samples for obtaining confidence interval of estimates in phase 2 for the high dimensional regression. The default is `500`.

`standardize` This takes a logical value `TRUE` or `FALSE`. If `TRUE`, all design matrix X in phase 1 and phase 2 will be standardized in the analyses. Default is `FALSE`.

`sequentialRun` This takes a logical value `TRUE` or `FALSE`. Default is `FALSE`. This argument could be useful for debug.

`refReadsThresh` The threshold of proportion of non-zero sequencing reads for choosing the reference taxon in phase 2. The default is `0.2` which means at least 20% non-zero sequencing reads.

`taxkeepThresh` The threshold of number of non-zero sequencing reads for each taxon to be included into the analysis. The default is `0` which means taxon with at least `0` sequencing reads will be included into the analysis

`SDThresh` The threshold of standard deviations of sequencing reads for been chosen as the reference taxon in phase 2. The default is `0.05` which means the standard deviation of sequencing reads should be at least `0.05` in order to be chosen as reference taxon.

`SDquantilThresh`
The threshold of the quantile of standard deviation of sequencing reads, above which could be selected as reference taxon. The default is `0`.

`balanceCut`
The threshold of the proportion of non-zero sequencing reads in each group of a binary variable for choosing the final reference taxa in phase 2. The default number is `0.2` which means at least 20% non-zero sequencing reads in each group are needed to be eligible for being chosen as a final reference taxon.

`seed`
Random seed for reproducibility. Default is 1. It can be set to be NULL to remove seeding.

Details

Most of the time, users just need to feed the first five inputs to the function: `MicrobData`, `CovData`, `linkIDname`, `testCov` and `ctrlCov`. All other inputs can just take their default values. To model the association, the following equation is used:

$$\log(\mathcal{Y}_i^k)|\mathcal{Y}_i^k > 0 = \beta^{0k} + X_i^T\beta^k + W_i^T\gamma^k + Z_i^Tb_i + \epsilon_i^k, \ \ k = 1, ..., K+1$$

where

- $\mathcal{Y}_i^k$ is the AA of taxa k in subject i in the entire ecosystem.
- X_i is the covariate matrix.
- W_i is the confounder matrix.
- Z_i is the design matrix for random effects.
- β^k is the regression coefficients that will be estimated and tested with the `IFAA()` function.

The challenge in microbiome analysis is that $\mathcal{Y}_i^k$ can not be observed. What is observed is its small proportion: $Y_i^k = C_i\mathcal{Y}_i^k$, where C_i is an unknown number between 0 and 1 that denote the observed proportion.

The IFAA method can successfully addressed this challenge. The `IFAA()` will estimate the parameter β^k and their 95% confidence intervals. High-dimensional X_i is handled by regularization.

Value

A list containing the estimation results.

- `sig_results`: A list containing estimating results that are statistically significant.
- `full_results`: A list containing all estimating results. NA denotes unestimable.
- `covariatesData`: A dataset containing covariates and confounders used in the analyses.

References

Li et al.(2021) IFAA: Robust association identification and Inference For Absolute Abundance in microbiome analyses. Journal of the American Statistical Association

Zhang CH (2010) Nearly unbiased variable selection under minimax concave penalty. Annals of Statistics. 38(2):894-942.

Liu et al.(2020) A bootstrap lasso + partial ridge method to construct confidence intervals for parameters in high-dimensional sparse linear models. Statistica Sinica

Examples

```
data(dataM)
dim(dataM)
dataM[1:5, 1:8]
data(dataC)
dim(dataC)
dataC[1:5, ]

results <- IFAA(MicrobData = dataM,
                CovData = dataC,
                linkIDname = "id",
                testCov = c("v1", "v2"),
                ctrlCov = c("v3"),
                fdrRate = 0.15)
```

MZILN *Conditional regression for microbiome analysis based on multivariate zero-inflated logistic normal model*

Description

For estimating and testing the associations of abundance ratios with covariates.

Usage

```
MZILN(
  MicrobData,
  CovData,
  linkIDname,
  targetTaxa = NULL,
  refTaxa,
  allCov = NULL,
  adjust_method = "BY",
  fdrRate = 0.15,
  paraJobs = NULL,
  bootB = 500,
  taxkeepThresh = 1,
  standardize = FALSE,
  sequentialRun = TRUE,
  seed = 1
)
```

Arguments

`MicrobData` Microbiome data matrix containing microbiome absolute abundance or relative abundance with each row per sample and each column per taxon/OTU/ASV (or any other unit). It should contain an id variable to be linked with the id variable in the covariates data: `CovData`. This argument can take directory path. For example, `MicrobData="C://...//microbiomeData.tsv"`.

`CovData` Covariates data matrix containing covariates and confounders with each row per sample and each column per variable. It should also contain an id variable to be linked with the id variable in the microbiome data: `MicrobData`. This argument can take directory path. For example, `CovData="C://...//covar`

`linkIDname` The common variable name of the id variable in both `MicrobData` and `CovData`. The two data sets will be merged by this id variable.

`targetTaxa` The numerator taxa names specified by users for the targeted ratios. Default is NULL in which case all taxa are numerator taxa (except the taxa in the argument 'refTaxa').

`refTaxa` Denominator taxa names specified by the user for the targeted ratios. This could be a vector of names.

`allCov` All covariates of interest (including confounders) for estimating and testing their associations with the targeted ratios. Default is 'NULL' meaning that all covariates in covData are of interest.

`adjust_method` The adjusting method for p value adjustment. Default is "BY" for dependent FDR adjustment. It can take any adjustment method for p.adjust function in R.

`fdrRate` The false discovery rate for identifying taxa/OTU/ASV associated with `allCov`. Default is `0.15`.

`paraJobs` If `sequentialRun` is `FALSE`, this specifies the number of parallel jobs that will be registered to run the algorithm. If specified as `NULL`, it will automatically detect the cores to decide the number of parallel jobs. Default is `NULL`.

`bootB` Number of bootstrap samples for obtaining confidence interval of estimates for the high dimensional regression. The default is `500`.

`taxkeepThresh` The threshold of number of non-zero sequencing reads for each taxon to be included into the analysis. The default is `1` which means taxon with at least `1` sequencing reads will be included into the analysis.

`standardize` This takes a logical value `TRUE` or `FALSE`. If `TRUE`, all design matrix X in the analyses will be standardized. Default is `FALSE`.

`sequentialRun` This takes a logical value `TRUE` or `FALSE`. Default is `TRUE`. It can be set to be "FALSE" to increase speed if there are multiple taxa in the argument 'refTaxa'.

`seed` Random seed for reproducibility. Default is `1`. It can be set to be NULL to remove seeding.

Details

Most of the time, users just need to feed the first six inputs to the function: `MicrobData`, `CovData`, `linkIDname`, `targetTaxa`, `refTaxa` and `allCov`. All other inputs can just take their default values. The regression model for `MZILN()` can be expressed as follows:

$$\log\left(\frac{\mathcal{Y}_i^k}{\mathcal{Y}_i^{K+1}}\right)|\mathcal{Y}_i^k > 0, \mathcal{Y}_i^{K+1} > 0 = \alpha^{0k} + \mathcal{X}_i^T\alpha^k + \epsilon_i^k, \quad k = 1, ..., K$$

where

- $\mathcal{Y}_i^k$ is the AA of taxa k in subject i in the entire ecosystem.

- $\mathcal{Y}_i^{K+1}$ is the reference taxon (specified by user).
- $\mathcal{X}_i$ is the covariate matrix for all covariates including confounders.
- α^k is the regression coefficients along with their 95% confidence intervals that will be estimated by the `MZILN()` function.

High-dimensional X_i is handled by regularization.

Value

A list containing the estimation results.

##' - `targettaxa_result_list`: A list containing estimating results for the targeted ratios. Only available when targetTaxa is non-empty.

- `sig_results`: A list containing estimating results for all significant ratios
- `covariatesData`: A dataset containing all covariates used in the analyses.

References

Li et al.(2018) Conditional Regression Based on a Multivariate Zero-Inflated Logistic-Normal Model for Microbiome Relative Abundance Data. Statistics in Biosciences 10(3): 587-608

Zhang CH (2010) Nearly unbiased variable selection under minimax concave penalty. Annals of Statistics. 38(2):894-942.

Liu et al.(2020) A bootstrap lasso + partial ridge method to construct confidence intervals for parameters in high-dimensional sparse linear models. Statistica Sinica

Examples

```
data(dataM)
dim(dataM)
dataM[1:5, 1:8]
data(dataC)
dim(dataC)
dataC[1:5, ]

results <- MZILN(MicrobData = dataM,
                 CovData = dataC,
                 linkIDname = "id",
                 targetTaxa = "rawCount6",
                 refTaxa=c("rawCount11"),
                 allCov=c("v1","v2","v3"),
                 fdrRate=0.15)
```

Index

∗ **datasets**
  dataC, 2
  dataM, 2

dataC, 2
dataM, 2

IFAA, 3

MZILN, 6