An Introduction to iNEXT.meta via Examples

2025-06-21

An Introduction to iNEXT.meta via Examples

`iNEXT.meta` (Meta-analysis of the difference between two treatments in interpolation and extrapolation for beta diversity across three dimensions)) is an R package that extends the concepts of iNEXT.beta3D (Chao et al., 2023) to meta analysis (fixed- or random-effects model). The measures are demonstrated using two datasets: spider data and bat data. The spider dataset contains species abundance data for 199 spider species across 11 sites (B04, B05, B06, B07, H09, L11, P08, S10, U01, U02, and U03), each assigned to one of two treatments: Control or Enhanced. The bat dataset consists of species-by-sampling-unit incidence data for 17 bat species collected from 6 sites (B04, B05, B06, B07, H09, and P08), also under the two treatments: Control and Enhanced.

For each of the three diversity dimensions—taxonomic (TD), phylogenetic (PD), and functional (FD)—iNEXT.meta estimates the difference in standardized diversity between two treatments, based on a common sample coverage, across various diversity types including alpha, beta, and gamma diversity, as well as four dissimilarity measures. The package then applies fixed- or random-effects models to perform meta-analysis across studies or sites. It also includes visualization tools such as forest plots to effectively present the analysis results.

SOFTWARE NEEDED TO RUN iNEXT.meta IN R

- Required: R
- Suggested: RStudio IDE

HOW TO RUN iNEXT.meta:

The `iNEXT.meta` package can be downloaded from Anne Chao's Github iNEXT.meta_github using the following commands. For a first-time installation, an additional visualization extension package (`forestplot` from CRAN) and (`iNEXT.beta3D` from Anne Chao's github) must be installed and loaded.

```
# install_github('AnneChao/iNEXT.beta3D')
# library(iNEXT.beta3D)

## install the latest version from github
#install.packages('devtools')
library(devtools)
# install_github('AnneChao/iNEXT.meta')
## import packages
library(iNEXT.meta)
```

There are three main functions in this package:

- **DataInfobeta3Dmeta**: Provides basic data information in each combination of site and treatment for (1) the gamma reference sample in the pooled assemblage, and (2) the alpha reference sample in the joint assemblage.
- **iNEXTbeta3Dmeta**: Estimates the difference of standardized 3D diversity with common sample coverage (for alpha, beta, gamma diversity, and four classes of dissimilarity measures) between two treatments, and fit a fixed- or random-effects model to perform meta-analysis.
- **ggiNEXTmeta**: Visualizes the output from the function iNEXTbeta3Dmeta by providing forest plot of the difference between two treatments of standardized 3D diversity in each study/site and meta-analysis (fixed- or random-effects model).

DATA INFORMATION: FUNCTION DataInfobeta3Dmeta()

The funciton `DataInfobeta3Dmeta()` provides basic data information in each combination of site and treatment for (1) the gamma reference sample in the pooled assemblage, and (2) the alpha reference sample in the joint assemblage. The function `DataInfobeta3Dmeta()` with default arguments is shown below:

```
iNEXTbeta3Dmeta(data, diversity = "TD", datatype = "abundance",
                PDtree = NULL, PDreftime = NULL,
                FDdistM = NULL, FDtype = "AUC", FDtau = NULL)
```

The arguments of this function are briefly described below, and will be explained in more details by illustrative examples in later text.

Argument	Description
`data`	a. For `datatype = "abundance"`, data can be input as a data.frame (species by assemblages). The data frame has study/site and treatment as the first two columns, followed by columns for species names. Here an assemblage refers to a combination of study/site and treatment. b. For `datatype = "incidence_raw"`, data can be input as a data.frame. The data frame has study/site, treatment, patch as the first three columns, followed by columns for species names.
`diversity`	Selection of diversity type: "TD" = Taxonomic diversity, "PD" = Phylogenetic diversity, and "FD" = Functional diversity.
`datatype`	Data type of input data: individual-based abundance data (`datatype = "abundance"`) or species by sampling-units incidence data (`datatype = "incidence_raw"`) with all entries being 0 (non-detection) or 1 (detection).
`PDtree`	(required only when `diversity = "PD"`), a phylogenetic tree in Newick format for all observed species in the pooled data.
`PDreftime`	(required only when `diversity = "PD"`), a numerical value specifying reference times for PD. Default is `NULL` (i.e., the age of the root of PDtree).
`FDdistM`	(required only when `diversity = "FD"`), a species pairwise distance matrix for all species in the pooled data.
`FDtype`	(required only when `diversity = "FD"`), select FD type: `FDtype = "tau_value"` for FD under a specified threshold value, or `FDtype = "AUC"` (area under the curve of tau-profile) for an overall FD which integrates all threshold values between zero and one. Default is `FDtype = "AUC"`.

Argument	Description
FDtau	(required only when diversity = "FD" and FDtype = "tau_value"), a numerical value between 0 and 1 specifying the tau value (threshold level) that will be used to compute FD. If FDtau = NULL (default), then threshold is set to be the mean distance between any two individuals randomly selected from the pooled data (i.e., quadratic entropy).

Running the DataInfobeta3Dmeta() function returns basic data information including sample size, observed species richness, and sample coverage. For abundance data, it provides both SC(n) and SC(2n); for incidence data, it provides SC(T) and SC(2T). The output also includes other relevant information across the three dimensions of diversity. We demonstrate the function using the Bat_incidence_data dataset for each dimension.

```
## Data information for taxonomic diversity
data("Bat_incidence_data")
infooutput1 <- DataInfobeta3Dmeta(data = Bat_incidence_data, diversity = "TD", datatype =
        "incidence_raw")


infooutput1
```

```
#> # A tibble: 15 × 13
#>    Site  Treatment Assemblage          T     U S.obs `SC(T)` `SC(2T)`    Q1    Q2
#>    <chr> <chr>     <chr>           <dbl> <dbl> <dbl>   <dbl>    <dbl> <dbl> <dbl>
#>  1 B04   Control   Pooled assemb…      4    22    12   0.811    0.957      6     4
#>  2 B04   Control   Joint assembl…      4    63    36   0.673    0.789     23     4
#>  3 B04   Enhanced  Pooled assemb…      4    36    14   0.979    1.00       2     5
#>  4 B04   Enhanced  Joint assembl…      4   100    57   0.782    0.940     30    17
#>  5 B05   Control   Pooled assemb…      4    19     8   0.947    0.997      2     3
#>  6 B05   Control   Joint assembl…      4    77    37   0.849    0.958     16     9
#>  7 B05   Enhanced  Pooled assemb…      4    24    11   0.889    0.978      4     3
#>  8 B05   Enhanced  Joint assembl…      4    60    31   0.787    0.913     16     6
#>  9 B06   Control   Pooled assemb…      4    24    10   0.964    0.999      2     4
#> 10 B06   Control   Joint assembl…      4    51    33   0.698    0.894     20     9
#> 11 B06   Enhanced  Pooled assemb…      4    31    12   0.993    1.00       1     5
#> 12 B06   Enhanced  Joint assembl…      4    74    42   0.709    0.839     25     6
#> 13 B07   Control   Pooled assemb…      4    22     7   0.973    0.996      1     1
#> 14 B07   Control   Joint assembl…      4    48    29   0.75     0.921     16     8
#> 15 B07   Enhanced  Pooled assemb…      4    27    13   0.889    0.986      5     5
#> # ℹ 3 more variables: Q3 <dbl>, Q4 <dbl>, Q5 <dbl>
```

Output description:

- Site = the input sites.
- Treatment = the input two treatment.
- Assemblage = individual assemblages, “Pooled assemblage” (for gamma) or “Joint assemblage” (for alpha).
- T = the number of sampling units in the reference sample (sample size for incidence data).
- U = total number of incidences in the reference sample.
- S.obs = number of observed species in the reference sample.
- SC(T) = sample coverage estimate of the reference sample.
- SC(2T) = sample coverage estimate of twice the reference sample size.
- Q1-Q5 = the first five species incidence frequency counts in the reference sample.

```
## Data information for phylogenetic diversity
data("Bat_incidence_data")
data("Bat_tree")
infooutput2 <- DataInfobeta3Dmeta(data = Bat_incidence_data, diversity = "PD", datatype =
        "incidence_raw",
                                  PDtree = Bat_tree, PDreftime = NULL)

infooutput2
```

```
#> # A tibble: 15 × 14
#>    Site  Treatment Assemblage        T     U S.obs `SC(T)` `SC(2T)` PD.obs `Q1*`
#>    <chr> <chr>     <chr>         <int> <dbl> <dbl>   <dbl>    <dbl>  <dbl> <dbl>
#>  1 B04   Control   Pooled assem…     4    22    12   0.811    0.957  139.      7
#>  2 B04   Control   Joint assemb…     4    63    36   0.673    0.789  525.     28
#>  3 B04   Enhanced  Pooled assem…     4    36    14   0.979    1.00   147.      2
#>  4 B04   Enhanced  Joint assemb…     4   100    57   0.782    0.940  815.     32
#>  5 B05   Control   Pooled assem…     4    19     8   0.947    0.997   91.0     2
#>  6 B05   Control   Joint assemb…     4    77    37   0.849    0.958  533.     18
#>  7 B05   Enhanced  Pooled assem…     4    24    11   0.889    0.978  119.      4
#>  8 B05   Enhanced  Joint assemb…     4    60    31   0.787    0.913  438.     16
#>  9 B06   Control   Pooled assem…     4    24    10   0.964    0.999  123.      2
#> 10 B06   Control   Joint assemb…     4    51    33   0.698    0.894  525.     23
#> 11 B06   Enhanced  Pooled assem…     4    31    12   0.993    1.00   144.      1
#> 12 B06   Enhanced  Joint assemb…     4    74    42   0.709    0.839  589.     25
#> 13 B07   Control   Pooled assem…     4    22     7   0.973    0.996   88.8     1
#> 14 B07   Control   Joint assemb…     4    48    29   0.75     0.921  450.     19
#> 15 B07   Enhanced  Pooled assem…     4    27    13   0.889    0.986  156.      6
#> # ℹ 4 more variables: `Q2*` <dbl>, R1 <dbl>, R2 <dbl>, Reftime <dbl>
```

Information description:

- `Site`, `Treatment`, `Assemblage`, `T`, `S.obs`, `SC(T)`, `SC(2T)` = definitions are the same as in the TD output.
- `PD.obs` = the observed total branch length in the phylogenetic tree spanned by all observed species.
- `f1*`, `f2*` = the number of singletons and doubletons in the node/branch abundance set.
- `g1`. `g2` =the total branch length of those singletons/doubletons in the node/branch abundance set.
- `Reftime` = reference time for phylogenetic diversity (the age of the root of phylogenetic tree).

```
## Data information for functional diversity (under a specified threshold level, FDtype =
        "tau_value")
data("Bat_incidence_data")
data("Bat_distM")
infooutput3 <- DataInfobeta3Dmeta(data = Bat_incidence_data, diversity = "FD", datatype =
        "incidence_raw",
                                  FDdistM = Bat_distM, FDtype = "tau_value", FDtau = NULL)

infooutput3
```

```
#> # A tibble: 15 × 13
#>    Site  Treatment Assemblage         T     U S.obs `SC(T)` `SC(2T)` `a1*` `a2*`
#>    <chr> <chr>     <chr>          <dbl> <dbl> <int>   <dbl>    <dbl> <int> <int>
#>  1 B04   Control   Pooled assemb…     4    22    12   0.811    0.957     1     3
#>  2 B04   Control   Joint assembl…     4    63    36   0.673    0.789    20     9
#>  3 B04   Enhanced  Pooled assemb…     4    36    14   0.979    1.00      0     1
```

```
#>  4 B04   Enhanced  Joint assembl…     4   100    57   0.782    0.940    27    19
#>  5 B05   Control   Pooled assemb…     4    19     8   0.947    0.997     1     2
#>  6 B05   Control   Joint assembl…     4    77    37   0.849    0.958    13    13
#>  7 B05   Enhanced  Pooled assemb…     4    24    11   0.889    0.978     1     4
#>  8 B05   Enhanced  Joint assembl…     4    60    31   0.787    0.913    12     9
#>  9 B06   Control   Pooled assemb…     4    24    10   0.964    0.999     1     2
#> 10 B06   Control   Joint assembl…     4    51    33   0.698    0.894    24     9
#> 11 B06   Enhanced  Pooled assemb…     4    31    12   0.993    1.00      0     0
#> 12 B06   Enhanced  Joint assembl…     4    74    42   0.709    0.839    19    11
#> 13 B07   Control   Pooled assemb…     4    22     7   0.973    0.996     0     0
#> 14 B07   Control   Joint assembl…     4    48    29   0.75     0.921    21     9
#> 15 B07   Enhanced  Pooled assemb…     4    27    13   0.889    0.986     0     2
#> # ℹ 3 more variables: h1 <dbl>, h2 <dbl>, Tau <dbl>
```

Information description:

- `Site`, `Treatment`, `Assemblage`, `T`, `S.obs`, `SC(T)`, `SC(2T)` = definitions are the same as in the TD output.
- `a1*`, `a2*` = the number of singletons (`a1*`) and of doubletons (`a2*`) among the functionally indistinct set at the specified threshold level `"Tau"`.
- `h1`, `h2` = the total contribution of singletons (`h1`) and of doubletons (`h2`) at the specified threshold level `"Tau"`.
- `Tau` = the specified threshold level of distinctiveness. Default is dmean (the mean distance between any two individuals randomly selected from the pooled data over all datasets).

```
## Data information for functional diversity (FDtype = "AUC")
data("Bat_incidence_data")
data("Bat_distM")
infooutput4 <- DataInfobeta3Dmeta(data = Bat_incidence_data, diversity = "FD", datatype =
        "incidence_raw",
                                  FDdistM = Bat_distM, FDtype = "AUC")

infooutput4
```

```
#> # A tibble: 15 × 11
#>    Site  Treatment Assemblage        T     U S.obs `SC(T)` `SC(2T)`   dmin dmean
#>    <chr> <chr>     <chr>         <dbl> <dbl> <dbl>   <dbl>    <dbl>  <dbl> <dbl>
#>  1 B04   Control   Pooled assem…     4    22    12   0.811    0.957 0.0621 0.283
#>  2 B04   Control   Joint assemb…     4    63    36   0.673    0.789 0.0621 0.283
#>  3 B04   Enhanced  Pooled assem…     4    36    14   0.979    1.00  0.0621 0.309
#>  4 B04   Enhanced  Joint assemb…     4   100    57   0.782    0.940 0.0621 0.309
#>  5 B05   Control   Pooled assem…     4    19     8   0.947    0.997 0.0621 0.203
#>  6 B05   Control   Joint assemb…     4    77    37   0.849    0.958 0.0621 0.203
#>  7 B05   Enhanced  Pooled assem…     4    24    11   0.889    0.978 0.0621 0.239
#>  8 B05   Enhanced  Joint assemb…     4    60    31   0.787    0.913 0.0621 0.239
#>  9 B06   Control   Pooled assem…     4    24    10   0.964    0.999 0.0621 0.260
#> 10 B06   Control   Joint assemb…     4    51    33   0.698    0.894 0.0621 0.260
#> 11 B06   Enhanced  Pooled assem…     4    31    12   0.993    1.00  0.0621 0.302
#> 12 B06   Enhanced  Joint assemb…     4    74    42   0.709    0.839 0.0621 0.302
#> 13 B07   Control   Pooled assem…     4    22     7   0.973    0.996 0.0621 0.206
#> 14 B07   Control   Joint assemb…     4    48    29   0.75     0.921 0.0621 0.206
#> 15 B07   Enhanced  Pooled assem…     4    27    13   0.889    0.986 0.0621 0.298
#> # ℹ 1 more variable: dmax <dbl>
```

Information description:

- `Site`, `Treatment`, `Assemblage`, `T`, `S.obs`, `SC(T)`, `SC(2T)` = definitions are the same as in the TD output.
- `dmin` = the minimum distance among all non-diagonal elements in the distance matrix.

-`dmean` = the mean distance between any two individuals randomly selected from each assemblage.

-`dmax` = the maximum distance among all elements in the distance matrix.

MAIN FUNCTION: iNEXTbeta3Dmeta()

We first describe the main function `iNEXTbeta3Dmeta()` with default arguments:

```
iNEXTbeta3Dmeta(data, model = "RE", diversity = "TD", order.q = c(0, 1, 2), datatype =
        "abundance",
                level = NULL, nboot = 10, treatment_order = c("Enhanced", "Control"), conf =
        0.95,
                PDtree, PDreftime = NULL, PDtype = "meanPD", FDdistM, FDtype = "AUC", FDtau =
        NULL, FDcut_number = 30)
```

The arguments of this function are briefly described below, and will be explained in more details by illustrative examples in later text.

Argument	Description
`data`	a. For `datatype = "abundance"`, data can be input as a data.frame (species by assemblages). The data frame has study/site and treatment as the first two columns, followed by columns for species names. Here an assemblage refers to a combination of study/site and treatment. b. For `datatype = "incidence_raw"`, data can be input as a data.frame. The data frame has study/site, treatment, patch as the first three columns, followed by columns for species names.
`model`	Selection of model type: “FE” = Fixed-effects model, “RE” = Random-effects model.
`diversity`	Selection of diversity type: “TD” = Taxonomic diversity, “PD” = Phylogenetic diversity, and “FD” = Functional diversity.
`order.q`	A numerical value specifying the diversity order, Default is `q = 0, 1, 2`.
`datatype`	Data type of input data: individual-based abundance data (`datatype = "abundance"`) or species by sampling-units incidence data (`datatype = "incidence_raw"`) with all entries being 0 (non-detection) or 1 (detection).
`level`	A numerical value between 0 and 1 specifying the sample coverage level used for computing standardized diversity and dissimilarity. By default(level = NULL), the function automatically calculates standardized 3D gamma, alpha, and beta diversities, along with four dissimilarity indices, up to the minimum coverage achieved by doubling the reference sample size across all site and treatment combinations.
`nboot`	A positive integer specifying the number of bootstrap replications when assessing sampling uncertainty for estimating standardized beta3D diversity and the associated confidence intervals. Default is 10. If more accurate results are required, set `nboot = 100` (or `nboot = 200`).
`treatment_order`	A character vector for the names of treatment. The difference of standardized beta3D diversity will be computed as diversity of the first treatment minus the diversity of second treatment.

Argument	Description
`conf`	A positive number < 1 specifying the level of confidence interval. Default is `conf = 0.95`.
`PDtree`	(required only when `diversity = "PD"`), a phylogenetic tree in Newick format for all observed species in the pooled data.
`PDreftime`	(required only when `diversity = "PD"`), a numerical value specifying reference times for PD. Default is `NULL` (i.e., the age of the root of PDtree).
`PDtype`	(required only when `diversity = "PD"`), select PD type: `PDtype = "PD"` (effective total branch length) or `PDtype = "meanPD"` (effective number of equally divergent lineages). Default is `PDtype = "meanPD"`, where `meanPD` = PD/tree depth.
`FDdistM`	(required only when `diversity = "FD"`), a species pairwise distance matrix for all species in the pooled data.
`FDtype`	(required only when `diversity = "FD"`), select FD type: `FDtype = "tau_value"` for FD under a specified threshold value, or `FDtype = "AUC"` (area under the curve of tau-profile) for an overall FD which integrates all threshold values between zero and one. Default is `FDtype = "AUC"`.
`FDtau`	(required only when `diversity = "FD"` and `FDtype = "tau_value"`), a numerical value between 0 and 1 specifying the tau value (threshold level) that will be used to compute FD. If `FDtau = NULL` (default), then threshold is set to be the mean distance between any two individuals randomly selected from the pooled data (i.e., quadratic entropy).
`FDcut_number`	(required only when `diversity = "FD"` and `FDtype = "AUC"`), a numeric number to cut [0, 1] interval into equal-spaced sub-intervals to obtain the AUC value by integrating the tau-profile. Equivalently, the number of tau values that will be considered to compute the integrated AUC value. Default is 30. A larger value can be set to obtain more accurate AUC value.

This function returns an `"iNEXTbeta3Dmeta"` object which can be further used to make plots using the function `ggiNEXTmeta()` to be described below.

DATA INPUT FORMAT

To perform meta-analysis across studies or sites, the input dataset must follow specific formatting rules depending on the data type:

Abundance Data (`datatype = "abundance"`)

- The input must be a `data.frame` with:
  - **First two columns**:
    - `Site`: study or site name
    - `Treatment`: treatment condition (e.g., Control, Enhanced)
  - **Remaining columns**:
    - Species names as column headers
    - Cell values indicate the **abundance** of each species in the assemblage

Incidence Data (`datatype = "incidence_raw"`)

- The input must be a `data.frame` with:

- **First three columns**:
  - `Site`: study or site name
  - `Treatment`: treatment condition
  - `Patch`: sampling unit ID
- **Remaining columns**:
  - Species names as column headers
  - Entries should be **0 or 1**, indicating absence or presence in each patch

---

Each **row** in the dataset represents a unique **assemblage**, defined by the combination of `Site` and `Treatment`.

There is **no need to manually concatenate** these columns into a single string (e.g., `"Site_B04_Control"`). The functions in the `iNEXT.meta` package will handle grouping and labeling automatically.

These assemblages are used to compute standardized diversity metrics (taxonomic, phylogenetic, or functional) and to compare treatment effects across multiple studies/sites for meta-analysis.

Bat Species Incidence Data Example

We use bat species incidence data collected from **six sites** under **two treatments** (Enhanced and Control). The dataset is named `Bat_incidence_data` and is included in the `iNEXT.meta` package.

The dataset is formatted as a **single data frame** suitable for `datatype = "incidence_raw"`, where each **row represents one sampling unit (patch)**. The data includes:

- `Site`: the study site (e.g., “B04”, “P08”, “U03”)
- `Treatment`: the treatment type (`"Enhanced"` or `"Control"`)
- `Patch`: the sampling unit within each site-treatment combination
- Species columns: presence (1) or absence (0) of each species in that sampling unit

This format allows the functions in the `iNEXT.meta` package to automatically recognize and group assemblages by site and treatment for further analysis.

You can load and view the dataset using the following code:

```
data("Bat_incidence_data")
Bat_incidence_data
```

```
#>    Site Treatment Patch Barbastella_barbastellus Eptesicus_nilssonii
#> 1   B04  Enhanced   100                         1                    0
#> 2   B04  Enhanced   100                         0                    0
#> 3   B04  Enhanced   100                         1                    1
#> 4   B04  Enhanced   100                         0                    0
#> 5   B04  Enhanced   101                         0                    1
#> 6   B04  Enhanced   101                         0                    0
#> 7   B04  Enhanced   101                         0                    1
#> 8   B04  Enhanced   101                         0                    0
#> 9   B04  Enhanced   102                         1                    1
#> 10  B04  Enhanced   102                         0                    0
#> 11  B04  Enhanced   102                         0                    1
#> 12  B04  Enhanced   102                         0                    0
#> 13  B04  Enhanced   103                         0                    1
#> 14  B04  Enhanced   103                         0                    0
#> 15  B04  Enhanced   103                         0                    1
```

Phylogenetic tree format for PD

To perform PD analysis, the phylogenetic tree (in Newick format) spanned by species observed in all datasets must be stored in a txt file. For example, the phylogenetic tree for all observed species is stored in a data file named “Bat_tree” for demonstration purpose. A partial list of the tip labels are shown below.

```
data("Bat_tree")
Bat_tree
#>
#> Phylogenetic tree with 17 tips and 16 internal nodes.
#>
#> Tip labels:
#>   Myotis_dasycneme, Myotis_myotis, Myotis_nattereri, Myotis_bechsteinii, Myotis_daubentonii,
        Myotis_mystacinus, ...
#>
#> Rooted; includes branch lengths.
```

Species pairwise distance matrix format for FD

To perform FD analysis, the species-pairwise distance matrix (Gower distance computed from species traits) for species observed in all datasets must be stored in a matrix/data.frame format. Typically, the distance between any two species is computed from species traits using the Gower distance. In our demo data, the distance matrix for all species is stored in a csv file named “Bat_distM” for demonstration purpose. Here we only show the first three rows and three columns of the distance matrix.

```
data("Bat_distM")
Bat_distM
```

```
#>                          Barbastella_barbastellus Eptesicus_nilssonii
#> Barbastella_barbastellus                    0.000                0.351
#> Eptesicus_nilssonii                         0.351                0.000
#> Eptesicus_serotinus                         0.346                0.219
#>                          Eptesicus_serotinus
#> Barbastella_barbastellus               0.346
#> Eptesicus_nilssonii                    0.219
#> Eptesicus_serotinus                    0.000
```

Output of the main function iNEXTbetameta()

The `iNEXTbeta3Dmeta()` function returns a **list with seven components**: `Gamma`, `Alpha`, `Beta`, `1-C`, `1-U`, `1-V`, and `1-S`.

Each component is itself a list containing:

1. A `data.frame` of site-level estimates

Each row corresponds to a unique study or site, and includes the following columns:

- `Site`: the name of the study/site (automatically taken from the `Site` column in the input)
- `Order.q`: the order of diversity (q = 0, 1, or 2)
- `Diversity`: the dimension of diversity (TD = Taxonomic, PD = Phylogenetic, FD = Functional)
- `Difference`: the difference in diversity between two treatments (calculated as *first treatment − second treatment*, based on `treatment_order`)

- `SE`: the standard error of the difference
- `LCL`, `UCL`: the lower and upper 95% confidence limits of the difference
- Two columns named after the treatments (e.g., `Enhanced`, `Control`), showing the estimated diversity values for each treatment in the corresponding site
- `Weight`: the weight assigned to each site for meta-analysis (depending on whether a fixed- or random-effects model is used)

2. A summary table

This includes the following meta-analytic statistics:

- `Q_val`: Cochran's Q statistic
- `df_val`: degrees of freedom
- `p_val`: p-value of the heterogeneity test
- `I2_val`: percentage of heterogeneity (I^2)
- `tau2_val`: estimated between-site variance (τ^2)

This structured output enables both per-site interpretation and global inference through meta-analysis.

Taxonomic diversity

First, we run the `iNEXTbeta3Dmeta()` function with `Bat_incidence_data` to compute the difference of taxonomic diversity between two treatments across all sites and perform meta analysis by running the following code:

```
#>        Site Difference   SE   LCL  UCL Order.q Diversity Enhanced Control Weight
#> 1       B04       1.16 0.97 -0.73 3.06       0        TD     6.35    5.19  13.72
#> 2       B05       1.26 0.65 -0.02 2.55       0        TD     4.70    3.44  29.92
#> 3       B06       1.19 0.60  0.03 2.36       0        TD     5.68    4.49  36.15
#> 4       B07       2.47 1.22  0.08 4.86       0        TD     5.43    2.96   8.61
#> 5       H09      -0.83 2.72 -6.15 4.50       0        TD     6.96    7.78   1.74
#> 6       P08      -0.60 1.14 -2.84 1.63       0        TD     5.04    5.64   9.86
#> 7 RE Model       1.11 0.36  0.41 1.81       0        TD       NA      NA 100.00


#> [[1]]
#> # A tibble: 1 × 6
#>   Order.q Q_val df_val p_val I2_val tau2_val
#>     <dbl> <dbl>  <dbl> <chr>  <dbl>    <dbl>
#> 1       0  4.08      5 0.538      0        0
```

Phylogenetic diversity

As with taxonomic diversity, `iNEXT.meta` computes the difference of phylogenetic diversity between two treatments across all sites and perform meta analysis.

The required argument for performing PD analysis is `PDtree`. For example, the phylogenetic tree for all observed species is stored in a txt file named "Bat_tree". Then we enter the argument `PDtree = Bat_tree`. Two optional arguments are: `PDtype` and `PDreftime`. There are two options for `PDtype`: `"PD"` (effective total branch length) or `"meanPD"` (effective number of equally divergent lineages, meanPD = PD/tree depth). Default is `PDtype = "meanPD"`. `PDreftime` is a numerical value specifying a reference time for computing phylogenetic diversity. By default (`PDreftime = NULL`), the reference time is set to the tree depth, i.e., age of the root of the phylogenetic tree.

Run the following code to perform PD analysis. The output is similar to the taxonomic diversity and thus is omitted; see later graphical display of the output.

```
## Phylogenetic diversity

data("Bat_incidence_data")
data("Bat_tree")
output2 <- iNEXTbeta3Dmeta(data = Bat_incidence_data, model = "RE", diversity = "PD",
                           order.q = 0, datatype = "incidence_raw", level = NULL,
                           nboot = 10, treatment_order = c("Enhanced", "Control"), conf =
        0.95,
                           PDtree = Bat_tree, PDreftime = NULL, PDtype = "meanPD")

output2
```

Functional diversity

As with taxonomic and phylogenetic diversity, `iNEXT.meta` computes the difference of functional diversity between two treatments across all sites and perform meta analysis.

The required argument for performing FD analysis is `FDdistM`. For example, the distance matrix for all species is stored in a csv file named “Bat_distM”. Then we enter the argument `FDdistM = Bat_distM`. Three optional arguments are (1) `FDtype`: `FDtype = "AUC"` means FD is computed from the area under the curve of a tau-profile by integrating all plausible threshold values between zero and one; `FDtype = "tau-value"` means FD is computed under a specific threshold value to be specified in the argument `FD_tau`. (2) `FD_tau`: a numerical value specifying the tau value (threshold level) that will be used to compute FD. If `FDtype = "tau-value"` and `FD_tau = NULL`, then the threshold level is set to be the mean distance between any two individuals randomly selected from the pooled data over all datasets (i.e., quadratic entropy). (3) `FDcut_number` is a numeric number to cut [0, 1] interval into equal-spaced sub-intervals to obtain the AUC value. Default is `FDcut_number = 30`. If more accurate integration is desired, then use a larger integer.

Run the following code to perform FD analysis. The output is similar to the taxonomic diversity and thus is omitted; see later graphical display of the output.

```
## Functional diversity

data("Bat_incidence_data")
data("Bat_distM")
output3 <- iNEXTbeta3Dmeta(data = Bat_incidence_data, model = "RE", diversity = "FD",
                           order.q = 0, datatype = "incidence_raw", level = NULL,
                           nboot = 10, treatment_order = c("Enhanced", "Control"), conf =
        0.95,
                           FDdistM = Bat_distM, FDtype = "AUC", FDcut_number = 30)

output3
```

GRAPHIC DISPLAYS: FUNCTION ggiNEXmeta()

The function `ggiNEXTmeta()` with default arguments is described as follows:

```
ggiNEXTmeta(output, order.q, num_round = 3, range, type, level = NULL)
```

Argument	Description
`output`	The output of the `iNEXTbeta3Dmeta function.`

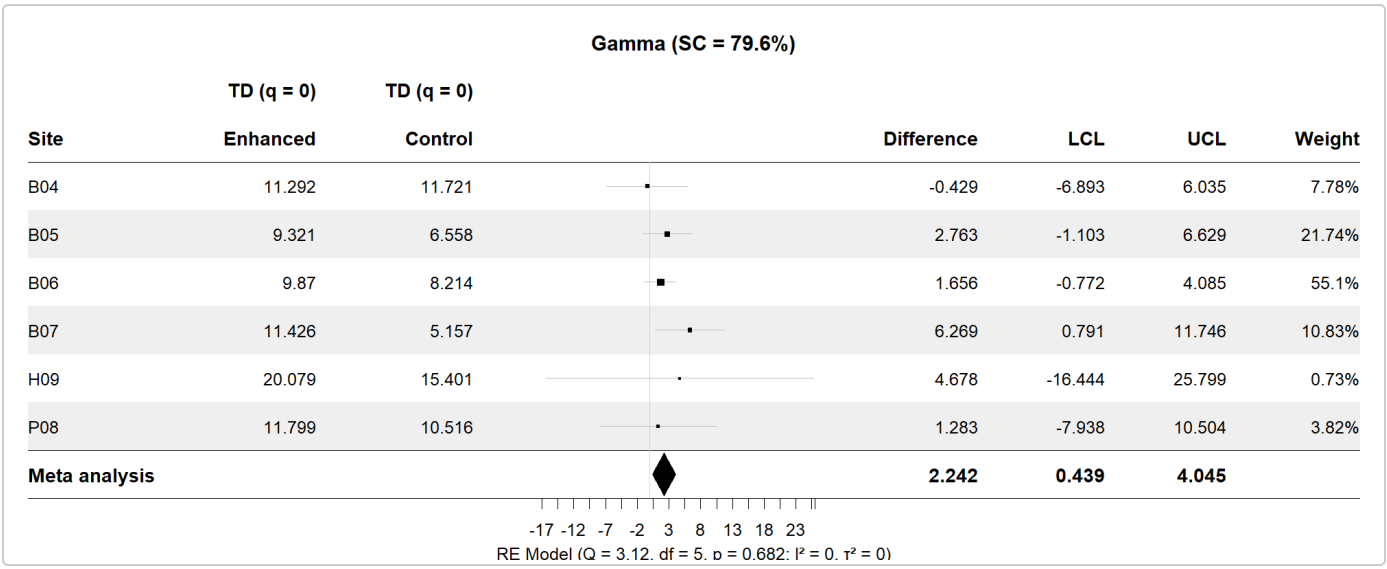
Argument	Description
`order.q`	A previously appeared “Order.q” value in “output”
`num_round`	A numerical value that the values show on the plot are rounded to the specified value of decimal places. Default is 3.
`range`	Lower and upper limits for clipping confidence intervals to arrows.
`type`	Specify diversity type (“Gamma”, “Alpha”, “Beta”), or dissimilarity type (“1-C”, “1-U”, “1-V”, “1-S”).
`level`	An optional sample coverage value (between 0 and 100 percent) to be annotated on the forest plot, indicating the fixed sample coverage used; if `level = NULL`, the annotation will be omitted.

The `ggiNEXTmeta()` function is a wrapper around the `forestplot` package. `ggiNEXTmeta()` provides forest plot for visualizing the output of `iNEXTbeta3Dmeta()`. Run the following code to display the plot for the output of `iNEXTbeta3Dmeta()` with taxonomic, phylogenetic and functional diversity, respectively:

```
## Taxonomic diversity

data("Bat_incidence_data")
infooutput <- DataInfobeta3Dmeta(data = Bat_incidence_data, diversity = "TD", datatype =
        "incidence_raw")
level1 <- infooutput |> filter(Assemblage == "Joint assemblage") |> pull(`SC(2T)`) |>
        quantile(0.25)
output1 <- iNEXTbeta3Dmeta(data = Bat_incidence_data, model = "RE", diversity = "TD",
                           order.q = 0, datatype = "incidence_raw", level = level1c,
                           nboot = 100, treatment_order = c("Enhanced", "Control"), conf =
        0.95)

ggiNEXTmeta(output1, order.q = 0, num_round = 3, range = c(-20, 15), type = "Gamma", level =
        round(level1, 3) * 100)
```

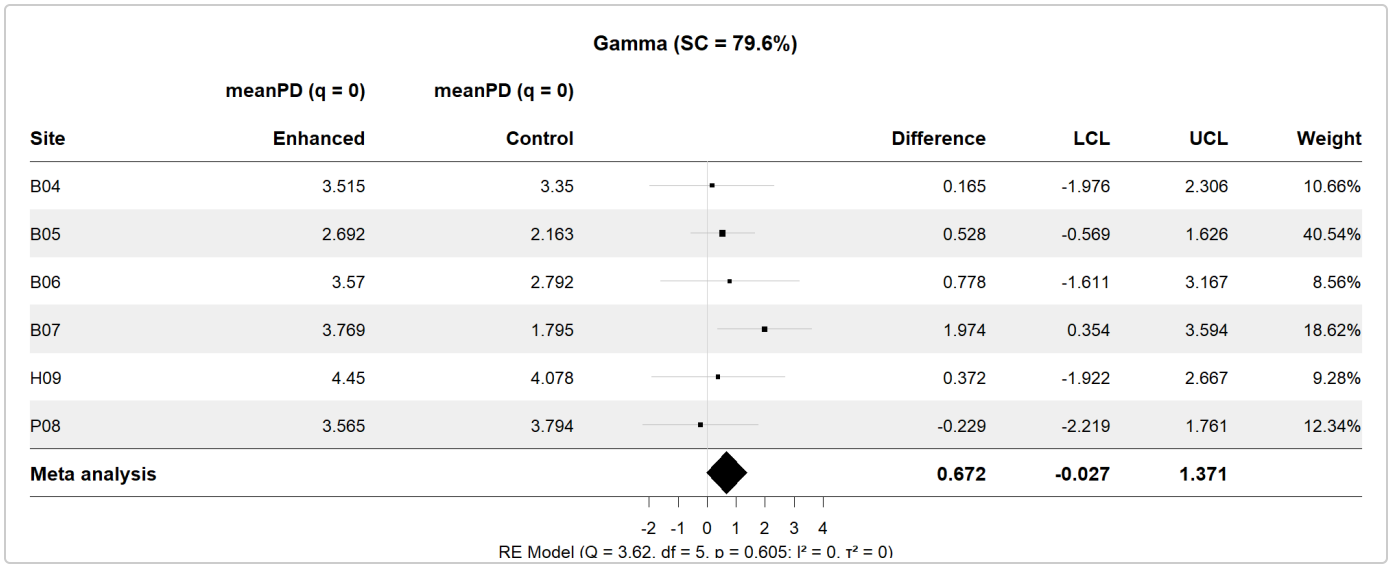



```
## Phylogenetic diversity

data("Bat_incidence_data")
data("Bat_tree")
infooutput <- DataInfobeta3Dmeta(data = Bat_incidence_data, diversity = "TD", datatype =
        "incidence_raw")
```

```
level1 <- infooutput |> filter(Assemblage == "Joint assemblage") |> pull(`SC(2T)`) |>
        quantile(0.25)
output2 <- iNEXTbeta3Dmeta(data = Bat_incidence_data, model = "RE", diversity = "PD",
                           order.q = 0, datatype = "incidence_raw", level = NULL,
                           nboot = 10, treatment_order = c("Enhanced", "Control"), conf =
        0.95,
                           PDtree = Bat_tree, PDreftime = NULL, PDtype = "meanPD")

ggiNEXTmeta(output2, order.q = 0, num_round = 3, range = c(-20, 15), type = "Gamma", level =
        round(level1, 3) * 100)
```



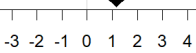

```
## Functional diversity

data("Bat_incidence_data")
data("Bat_distM")
infooutput <- DataInfobeta3Dmeta(data = Bat_incidence_data, diversity = "TD", datatype =
        "incidence_raw")
level1 <- infooutput |> filter(Assemblage == "Joint assemblage") |> pull(`SC(2T)`) |>
        quantile(0.25)
output3 <- iNEXTbeta3Dmeta(data = Bat_incidence_data, model = "RE", diversity = "FD",
                           order.q = 0, datatype = "incidence_raw", level = NULL,
                           nboot = 10, treatment_order = c("Enhanced", "Control"), conf =
        0.95,
                           FDdistM = Bat_distM, FDtype = "AUC", FDcut_number = 30)

ggiNEXTmeta(output3, order.q = 0, num_round = 3, range = c(-20, 15), type = "Gamma", level =
        round(level1, 3) * 100)
```

Gamma (SC = 79.6%)

Site	FD_AUC (q = 0) Enhanced	FD_AUC (q = 0) Control	Difference	LCL	UCL	Weight
B04	6.347	5.185	1.162	-1.141	3.465	15.08%
B05	4.701	3.438	1.264	-0.597	3.125	23.09%
B06	5.68	4.488	1.192	-0.893	3.277	18.4%
B07	5.433	2.963	2.47	0.474	4.466	20.07%
H09	6.044	5.403	0.641	-2.242	3.524	9.62%
P08	5.038	5.638	-0.601	-3.013	1.812	13.74%
Meta analysis			**1.161**	**0.267**	**2.056**	



RE Model (Q = 3.84, df = 5, p = 0.573; I^2 = 0, τ^2 = 0)