A Quick Introduction to iNEXT via Examples

T. C. Hsieh, K. H. Ma, and Anne Chao

Institute of Statistics, National Tsing Hua University, Hsin-Chu, Taiwan 30043

`iNEXT` (iNterpolation and EXTrapolation) is an R package modified from the original version, which was supplied in the Supplement of Chao et al. (2014). In the latest, updated version, we have added more user-friendly features and refined the graphic displays. In this document, we provide a quick introduction demonstrating how to run `iNEXT`. Detailed information about `iNEXT` functions is provided in the iNEXT Manual, also available in CRAN. See Chao & Jost (2012), Colwell et al. (2012) and Chao et al. (2014) for methodologies. A short review of theoretical background and methods relevant to the package are included in an application paper by Hsieh, Ma & Chao (2016). An online version of `iNEXT` (https://chao.shinyapps.io/iNEXT/) is also available for users without an R background.

`iNEXT` focuses on three measures of Hill numbers of order `q`: species richness (`q=0`), Shannon diversity (`q=1`, the exponential of Shannon entropy) and Simpson diversity (`q=2`, the inverse of Simpson concentration). For each diversity measure, `iNEXT` uses the observed sample of abundance or incidence data (called the "reference sample") to compute diversity estimates and the associated 95% (default) confidence intervals as well as plot the following two types of rarefaction and extrapolation (R/E) curves:

1. Sample-size-based R/E sampling curves: `iNEXT` computes diversity estimates for rarefied and extrapolated samples up to double the reference sample size (by default) or a user-specified size. This type of sampling curve plots the diversity estimates with respect to sample size. Sample size refers to the number of individuals in a sample for abundance data, whereas it refers to the number of sampling units for incidence data.

2. Coverage-based R/E sampling curves: `iNEXT` computes diversity estimates for rarefied and extrapolated samples with sample completeness (as measured by sample coverage) up to the coverage value of double the reference sample size (by default) or a user-specified coverage. This type of sampling curve plots the diversity estimates with respect to sample coverage.

In addition to the above two types of sampling curves, `iNEXT` also plots a sample completeness curve, which depicts how the sample coverage estimate varies as a

function of sample size. The sample completeness curve can be thought of as a bridge connecting the afore-mentioned two types of curves.

SOFTWARE NEEDED TO RUN INEXT IN R

Required: R
Suggested: RStudio IDE

HOW TO RUN INEXT

The `iNEXT` package is available on CRAN and can be downloaded with a standard R installation procedure using the following commands shown below or from the github. For a first-time installation, an additional visualization extension package (`ggplot2`) must be loaded.

```
## install iNEXT package from CRAN
install.packages("iNEXT")

## install the latest version from github
install.packages('devtools')
library(devtools)
install_github('JohnsonHsieh/iNEXT')

## import packages
library(iNEXT)
library(ggplot2)
```

Remark: In order to install the `devtools` package, you should update R to the latest version. Also, to get `install_github` to work properly, you should install the `httr` package.

MAIN FUNCTION: iNEXT()

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

```
iNEXT (x, q=0, datatype="abundance", size=NULL, endpoint=NULL, knots=40,
se=TRUE, conf=0.95, nboot=50)
```

The arguments of this function are briefly described below, and will be explained in more detail through illustrative examples later in the text. This main function computes diversity estimates of order `q`, the sample coverage estimates and related statistics for K (if `knots=K`) evenly-spaced `knots` (sample sizes) between size 1 and the `endpoint`, where the `endpoint` is as described below. Each knot represents a particular sample

size for which diversity estimates will be calculated. By default, `endpoint` is set to be double the reference sample size. For example, if `endpoint=10`, `knot=4`, then diversity estimates will be computed for a sequence of samples with sizes `(1,4,7,10)`.

Argument	Description
`x`	a `matrix`, `data.frame`, `lists` of species abundances/incidences, or `lists` of incidence frequencies (see data format/information below).
`q`	a number or vector specifying the diversity order(s) of Hill numbers;
`datatype`	type of input data, `"abundance"`, `"incidence_raw"`, or "`incidence_freq`";
`size`	an integer vector of sample sizes for which diversity estimates will be computed. If `NULL`, then diversity estimates will be calculated for those sample sizes determined by the specified/default `endpoint` and `knots`;
`endpoint`	an integer specifying the sample size that is the endpoint for R/E calculation; If `NULL`, then `endpoint`=double the reference sample size;
`knots`	an integer specifying the number of equally-spaced (by default) knots between size 1 and the endpoint;
`se`	a logical variable to calculate the bootstrap standard error and confidence interval of a level specified by `conf`;
`conf`	a positive number < 1 specifying the level of confidence interval.
`nboot`	an integer specifying the number of bootstrap replications.

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

DATA FORMAT/INFORMATION

Three types of data are supported: (`"abundance"`, `"incidence_raw"`, or `"incidence_freq"`):

1. Individual-based abundance data (`datatype="abundance"`): Input data for each assemblage/site include sample species abundances in an empirical sample of n individuals ("reference sample"). When there are *N* assemblages, input data consist of an *S* by *N* abundance `matrix`, or *N* `lists` of species abundances.
2. Sampling-unit-based incidence data: There are two kinds of input data.

   (2a) Incidence-raw data (`datatype="incidence_raw"`): for each assemblage, input data for a reference sample consist of a species-by-sampling-unit matrix; when there are *N* assemblages, input data consist of *N* `lists` of matrices, and each matrix is a species-by-sampling-unit matrix.

(2b) Incidence-frequency data (`datatype="incidence_freq"`): input data for each assemblage consist of species sample incidence frequencies (row sums of each incidence matrix). When there are *N* assemblages, input data consist of an *S* by *N* `matrix`, or *N* `lists` of species incidence frequencies. The first entry of each column/list must be the total number of sampling units, followed by the species incidence frequencies.

Four data sets (`spider` and `bird` for abundance data, and `ant` and `ciliates` for incidence data) are included in the `iNEXT` package for illustrating the data input formats and running procedures.

RAREFACTION/EXTRAPOLATION VIA EXAMPLES (ABUNDANCE DATA)

We begin by making use of the `spider` data in order to demonstrate basic `iNEXT()` functions and graphical displays. The spider data consist of species sample abundances from two canopy manipulation treatments ("Girdled" and "Logged") of hemlock trees (Ellison et al. 2010); see Chao et al. (2014) for analysis details and data interpretations. For these data, the following commands display the sample species abundances and run the `iNEXT()` function for `q=0`.

```
data(spider)
str(spider)
List of 2
 $ Girdled: num [1:26] 46 22 17 15 15 9 8 6 6 4 ...
 $ Logged : num [1:37] 88 22 16 15 13 10 8 8 7 7 ...
```

```
iNEXT(spider, q=0, datatype="abundance")
```

The `iNEXT()` function returns the `"iNEXT"` object including three data frames: `$DataInfo` for summarizing data information; `$iNextEst` for showing diversity estimates along with related statistics for a series of rarefied and extrapolated samples; and `$AsyEst` for showing asymptotic diversity estimates along with related statistics.

`$DataInfo`, as shown below, returns basic data information including the site name (`site`), reference sample size (`n`), observed species richness (`S.obs`), a sample coverage estimate (`SC`), and the first ten frequency counts (`f1-f10`). This latter output can also be produced by calling the function `DataInfo()`, whereby `f1` denotes the number of species represented by exactly one individual (i.e., "singletons"), `f2` denotes the number of species represented by exactly two individuals (i.e., "doubletons"), and `fk` denotes the number of species represented by exactly k individuals.

```
$DataInfo: basic data information

     site   n S.obs     SC f1 f2 f3 f4 f5 f6 f7 f8 f9 f10
1 Girdled 168    26 0.9289 12  4  0  1  0  2  0  1  1   0
2  Logged 252    37 0.9446 14  4  4  3  1  0  3  2  0   1
```

In the Girdled treatment site, by default, 40 equally spaced knots (samples sizes) between 1 and 336 (=2 x 168, double the reference sample size, Chao et al. 2014) are selected. Diversity estimates and related statistics are computed for these 40 knots (corresponding to sample sizes m =1, 10, 19, …, 336), which locates the reference sample at the mid-point of the selected knots. If the argument `se=TRUE`, then the bootstrap method is applied to obtain the confidence intervals at a specified level (default =0.95) for each diversity and sample coverage estimate.

For each sample size corresponding to a knot, the list $iNextEst (as shown below for the Girdled treatment site) includes the sample size (`m`, i.e., size for each of the 40 knots), the `method` (`interpolated`, `observed`, or `extrapolated`, depending on whether the size `m` is less than, equal to, or greater than the reference sample size), the diversity order, the diversity estimate of order q (`qD`), the 95% (default) lower and upper confidence limits of diversity (`qD.LCL`, `qD.UCL`), and the sample coverage estimate (`SC`) along with the 95% (default) lower and upper confidence limits of sample coverage (`SC.LCL`, `SC.UCL`). These sample coverage estimates with confidence intervals are used for plotting the sample completeness curve and coverage-based R/E curves. The following output shows a partial list of diversity estimates:

```
$iNextEst: diversity estimates with rarefied and extrapolated samples.

$Girdled
     m       method order     qD qD.LCL qD.UCL    SC SC.LCL SC.UCL

1    1 interpolated     0  1.000  1.000  1.000 0.122  0.089  0.156
10  84 interpolated     0 18.912 15.902 21.923 0.900  0.872  0.927
20 168     observed     0 26.000 21.492 30.508 0.929  0.904  0.954
30 248 extrapolated     0 30.883 25.149 36.618 0.948  0.918  0.979
40 336 extrapolated     0 34.731 27.187 42.275 0.964  0.931  0.996
```

`$AsyEst` lists the observed diversity, asymptotic estimates, estimated bootstrap s.e. and 95% (default) confidence intervals for Hill numbers of order q = 0, 1, and 2. The estimated asymptotes are calculated via the functions `ChaoRichness()` for q=0, `ChaoShannon()` for q=1 and `ChaoSimpson()` for q=2; see Chao et al. (2014) for the formulas of these asymptotic estimators. The output for the spider data is shown below. All row and column variables are self-explanatory.

```
$AsyEst: asymptotic diversity estimates along with related statistics.
```

```
                            Observed Estimator    s.e.     LCL       UCL

Girdled Species richness      26.000    43.893   14.306   30.511   96.971
        Shannon diversity     12.060    13.826    1.531   12.060   16.827
        Simpson diversity      7.840     8.175    0.879    7.840    9.897
Logged  Species richness      37.000    61.403   18.532   43.502  128.583
        Shannon diversity     14.421    16.337    1.629   14.421   19.531
        Simpson diversity      6.761     6.920    0.802    6.761    8.492
```

The user may specify an integer sample size for the argument endpoint to designate the maximum sample size of the R/E calculation. For species richness, the extrapolation method is reliable up to double the reference sample size; beyond that, the prediction bias may be large. However, for measures of order q = 1 and 2, the extrapolation can usually be safely extended to the asymptote if data are not sparse; thus there is no limit for the value of the endpoint for these two measures.

The user may also specify the number of `knots` (i.e., specify some particular sample sizes) between 1 and the `endpoint`. If you choose a large number of knots, then it may take a long time to obtain the output due to the time-consuming bootstrap method. Alternatively, the user may specify a series of sample `sizes` for R/E computation, as in the following example:

```
# set a series of sample sizes (m) for R/E computation
m <- c(1, 5, 20, 50, 100, 200, 400)
iNEXT(spider, q=0, datatype="abundance", size=m)
```

The above code will return species richness estimates for the specified sample sizes as well as those for the reference samples size and two neighboring sizes. Further, iNEXT can simultaneously run R/E computation for Hill numbers of order q = 0, 1, and 2 by specifying a vector for the argument q as follows:

```
out <- iNEXT(spider, q=c(0,1,2), datatype="abundance", size=m)
```

In many applications, species data only consist of abundance frequency counts (f_1, f_2, …, f_L), where L denotes the maximum frequency; see the output in the list `$DataInfo`. In this case, the frequency counts must be converted to species abundances. As an example, the frequency counts for the spider data are given in Table 3 of Chao et al. (2014), the following code will convert the frequency counts to `iNEXT` input data:

```
# Convert abundance frequency counts to species abundance data
count_1=c(12,4,1,2,1,1,2,1,1,1)
count_2=c(14,4,4,3,1,3,2,1,1,1,1,1,1)
X1=rep(c(1,2,4,6,8,9,15,17,22,46), count_1)
X2=rep(c(1:5,7,8,10,13,15,16,22,88), count_2)
spider= list(Girdled=X1, Logged=X2)
```

Then the converted data are the same as those included in the `spider` set included in the iNEXT package.

BASIC GRAPHIC DISPLAYS: FUNCTION ggiNEXT()

The function `ggiNEXT()`, which extends `ggplot2` to the `"iNEXT"` object, is described as follows with default arguments:

```
ggiNEXT(x, type=1, se=TRUE, facet.var="none", color.var="site", grey=FALSE)
```

Here `x` is an `iNEXT` object. The `ggiNEXT()` function is a wrapper around the `ggplot2` package to create R/E curves using a single line of code. The resulting object is of class `"ggplot"`, so it can be manipulated using the `ggplot2` tools. Three types of curves are supported:

(1) Sample-size-based R/E curve (`type=1`): see Figs. 1a and 2a in Hsieh, Ma and Chao (2016). This curve plots diversity estimates with confidence intervals (if `se=TRUE`) as a function of sample size up to double the reference sample size by default, or a user-specified endpoint.

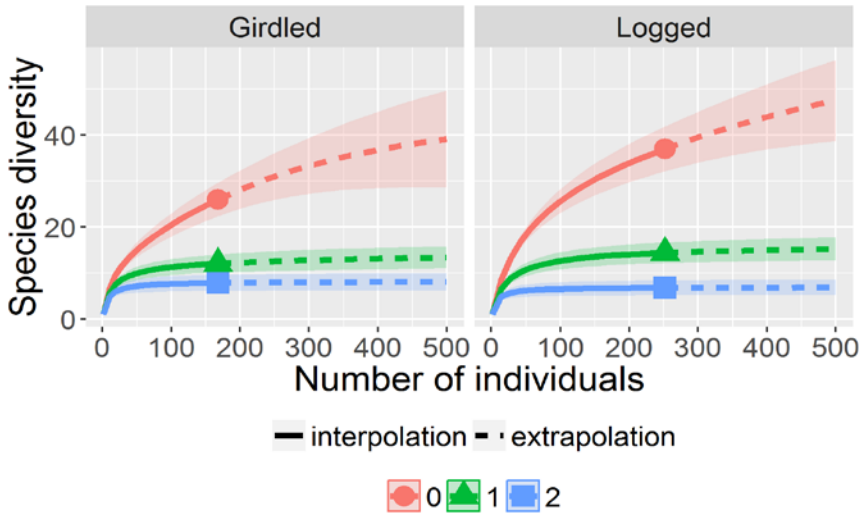
(2) Sample completeness curve (`type=2`) with confidence intervals (if `se=TRUE`): see Figs. 1b and 2b in Hsieh, Ma and Chao (2016). This curve plots the sample coverage with respect to sample size for the same range described in (1).

(3) Coverage-based R/E curve (`type=3`): see Figs. 1c and 2c in Hsieh, Ma and Chao (2016). This curve plots the diversity estimates with confidence intervals (if `se=TRUE`) as a function of sample coverage up to the maximum coverage obtained from the maximum size described in (1).

The argument `se is` a logical variable to plot the confidence interval at a level specified by the argument `conf`. The argument `facet.var ("none", "order", "site" or "both")` is used to create a separate plot for each value of the specified variable. When `facet.var="both"`, we can further use the argument `color.var ("none", "order", "site" or "both")` to display curves in a different color for each value of the values of the specified variable. The user may also use the argument `grey=TRUE` to plot black/white figures. Several examples are given below for the `spider` data.

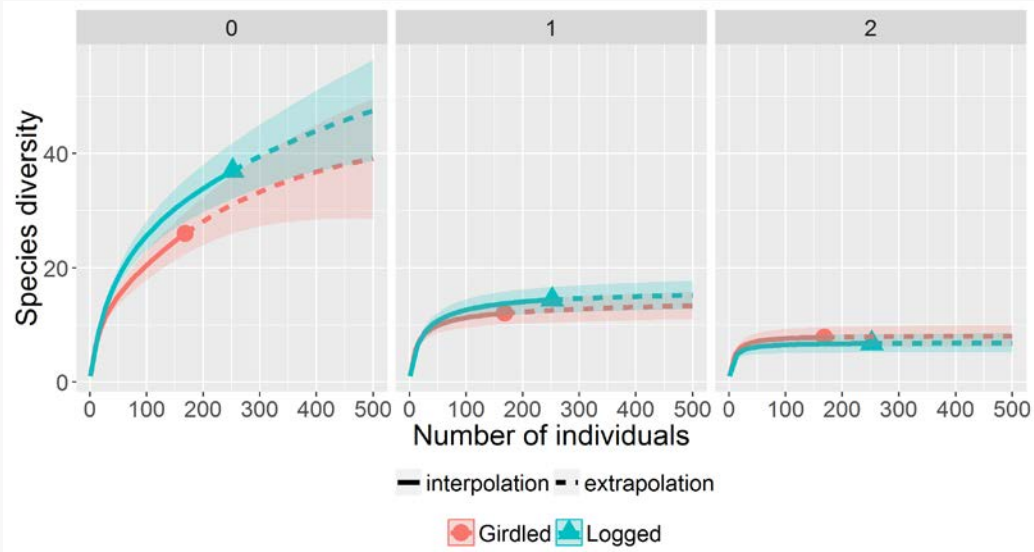
The following commands return the sample-size-based R/E sampling curves. The argument `facet.var="site"` in the ggiNEXT function creates a separate plot for each site; within each site, three measures (q = 0, 1 and 2) are shown. The legend position (by default) is placed below the graphical displays.

```
out <- iNEXT(spider, q=c(0, 1, 2), datatype="abundance", endpoint=500)
# Sample-size-based R/E curves, separating plots by "site"
ggiNEXT(out, type=1, facet.var="site")
```



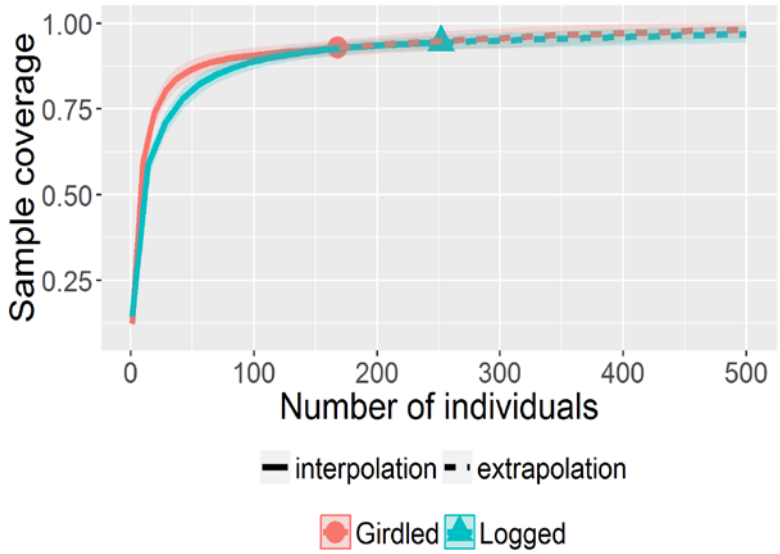
The following commands return the sample-size-based R/E sampling curves. The argument `facet.var="order"` in the ggiNEXT function creates a separate plot for each diversity order 0, 1 and 2. Within each order, curves for two sites (Girdled and Logged) are shown.

```
# Sample-size-based R/E curves, separating plots by "order"
ggiNEXT(out, type=1, facet.var="order")
```



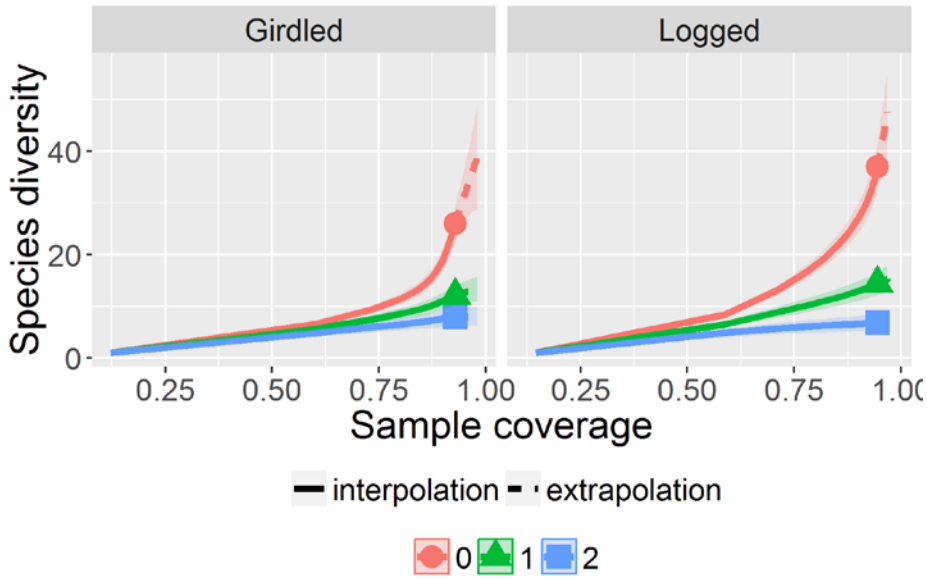
To link the sample-sized and coverage-based sampling curves, it would be informative to first examine the sample completeness curve using the following command:

```
ggiNEXT(out, type=2)
```



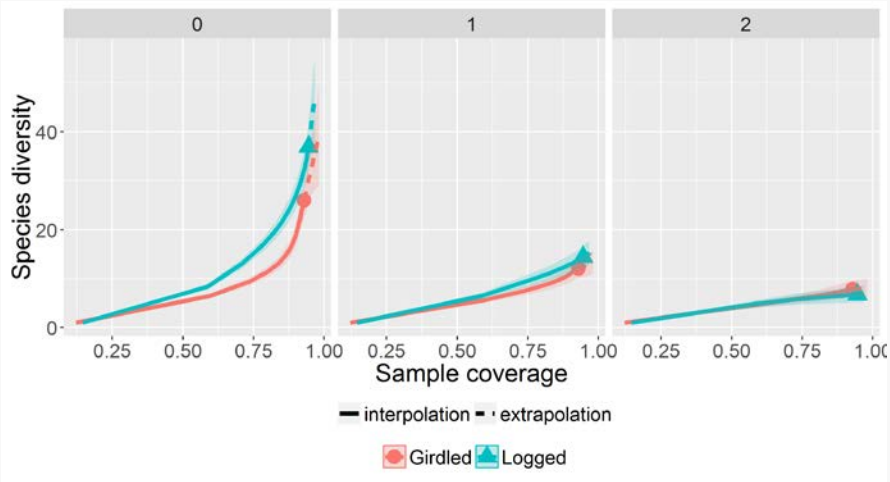
The following commands return the coverage-based R/E sampling curves. The argument `facet.var="site"` in the ggiNEXT function creates a separate plot for each site, as shown below:

```
ggiNEXT(out, type=3, facet.var="site")
```



The argument `facet.var="order"` creates a separate plot for each diversity order, and within each plot, as shown below.

```
ggiNEXT(out, type=3, facet.var="order")
```

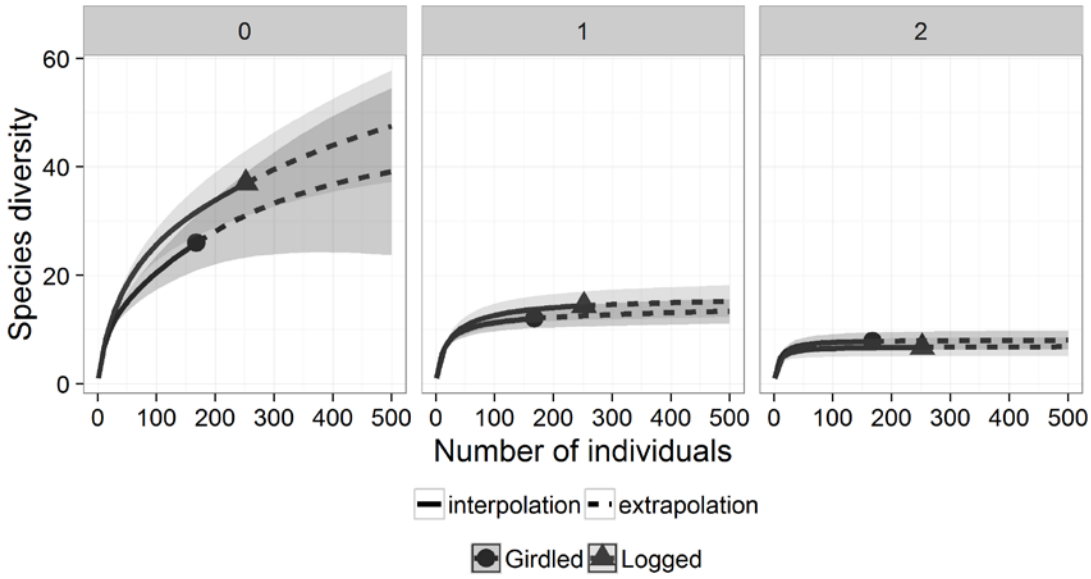


The above graphical displays depict the typical color plots to standardize biodiversity samples in order to compare equally-large (sample-size-based) or equally-complete (coverage-based) samples. More graphic display options are described below.

MORE GRAPHIC DISPLAY OPTIONS

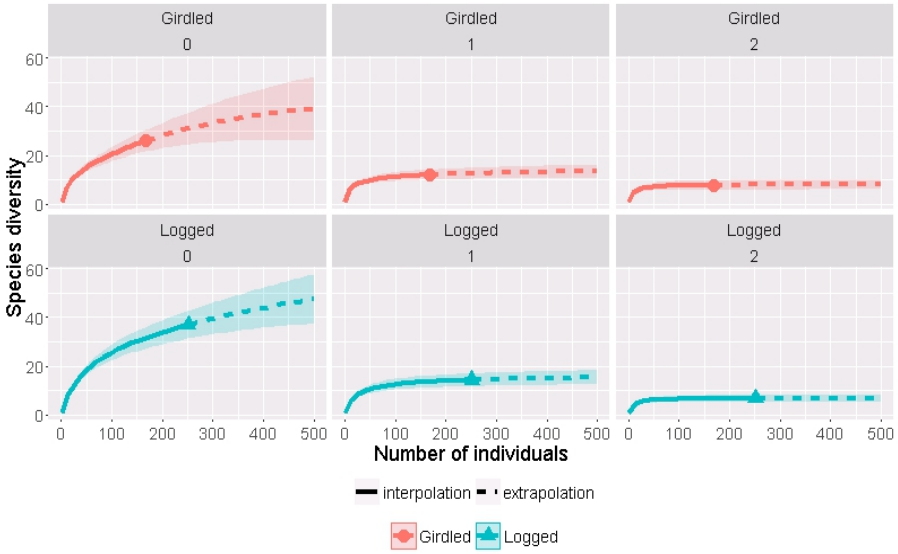
You can use the optional argument `grey=TRUE` in the **`ggiNEXT()`** function to output black-and-white plots. The following commands display the sample-size-based R/E sampling curves in black-and-white separately for three diversity orders: (Similar black-and-white plots can be made for the corresponding sample-completeness curve and coverage-based curves.)

```
# Separating plots by "order", display black-white plots
ggiNEXT(out, type=1, facet.var="order", grey=TRUE)
```



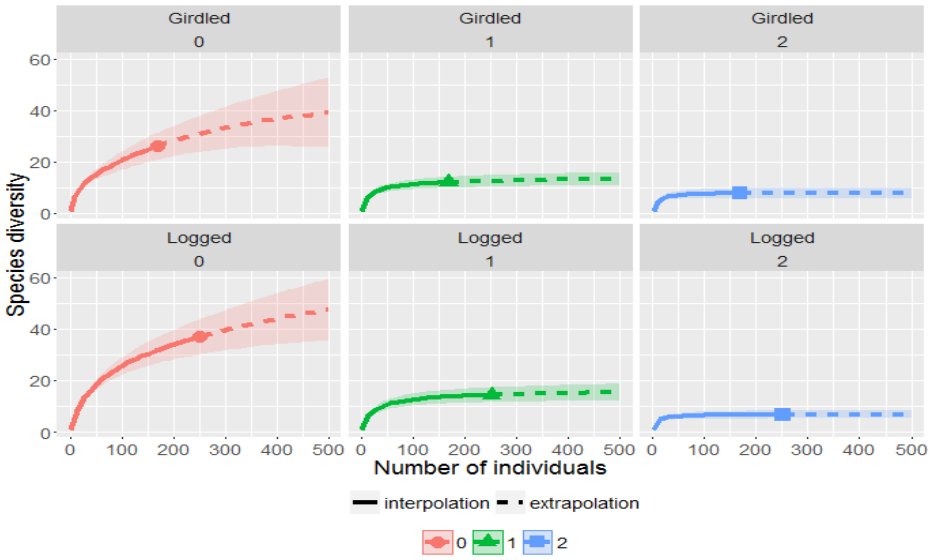
The argument `facet.var="both"` and `color.var="site"` creates a separate plot for each combination of diversity order and site, with different colors used for the two sites, as shown below for sample-size-based R/E curves. Similar plots can be made for coverage-based curves.

```
ggiNEXT(out, type=1, facet.var="both", color.var="site")
```



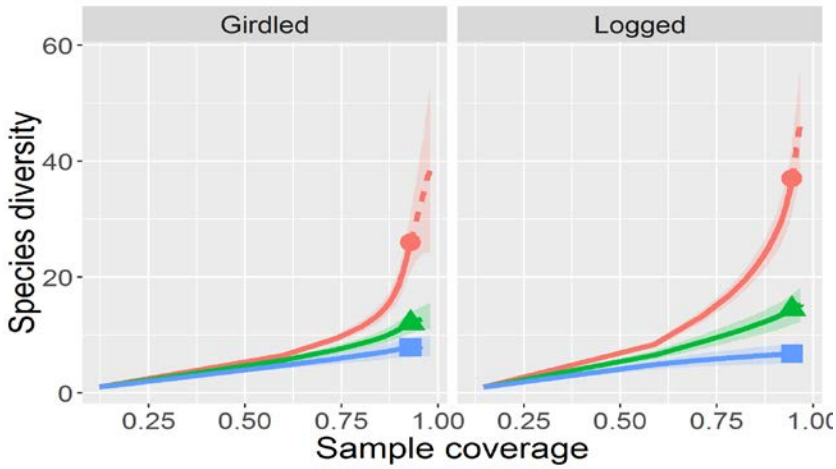
The argument `facet.var="both"` and `color.var="order"` creates a separate plot for each combination of diversity order and site, with different colors used for the three orders, as shown below.

```
ggiNEXT(out, type=1, facet.var="both", color.var="order")
```



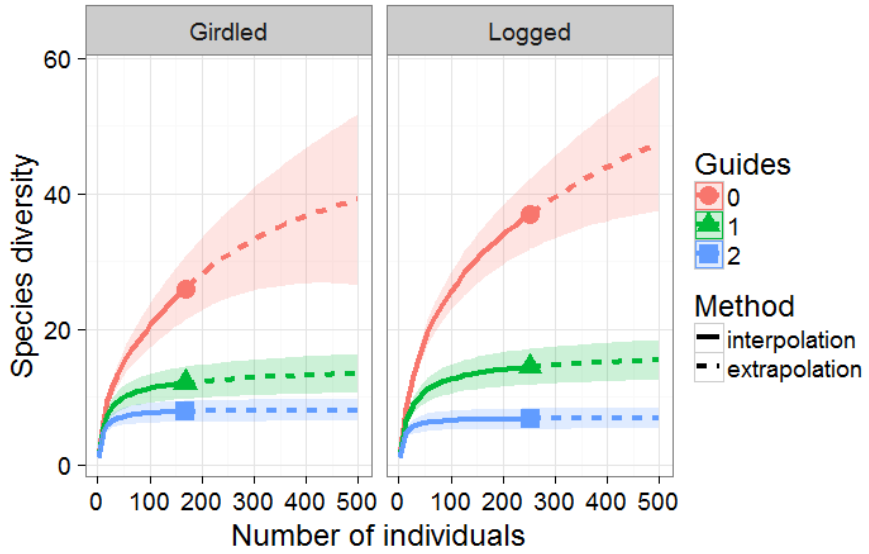
The legend can be removed by adding the code `theme(legend.position="none")` as shown below:

```
# Remove Legend

out <- iNEXT(spider, q=c(0, 1, 2), datatype="abundance", endpoint=500)
ggiNEXT(out, type=3, facet.var="site") + theme(legend.position="none")
```



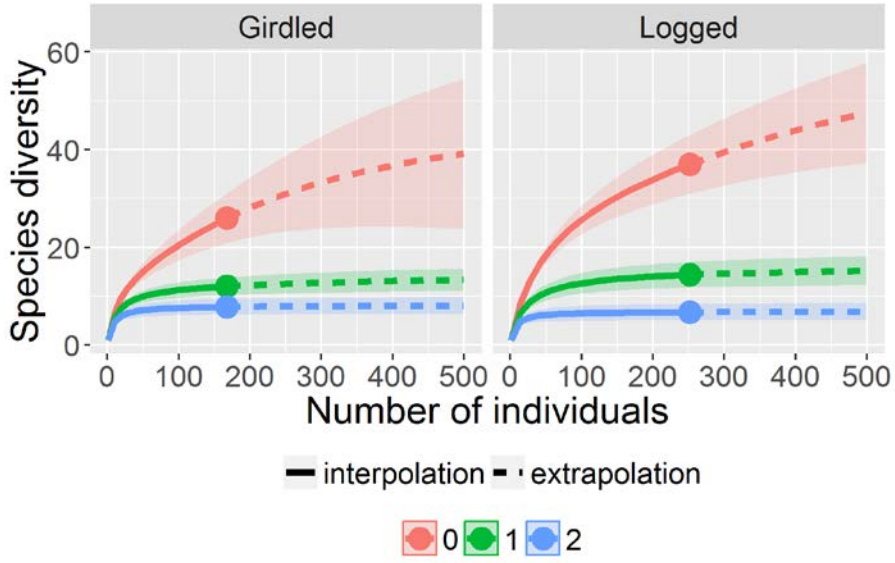
The gray-grid theme can be changed to a black-and-white theme by adding the code `theme_bw()`. For the black-and-white theme, the legend position (by default) is placed on the right of the displays. The size of all legends/labels can also be enlarged as shown in the following example.

```
# Change to B-W theme
ggiNEXT(out, type=1, facet.var="site") + theme_bw(base_size=18)
```



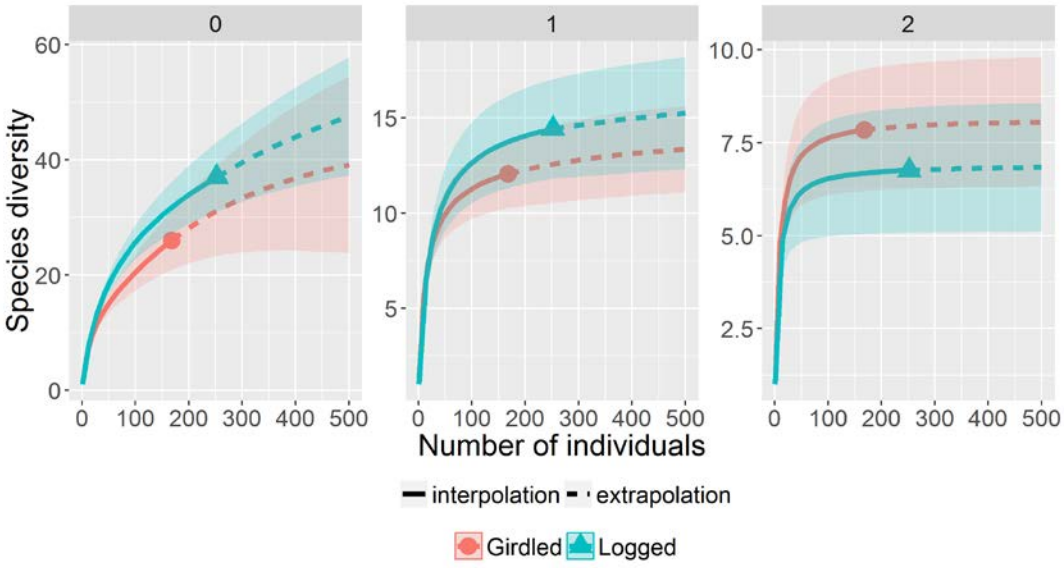
By default, `iNEXT` uses different shapes for the reference-sample points. The shape can be changed to be the same for all reference sample points:

```
# Change the shape of reference-sample points
ggiNEXT(out, type=1, facet.var="site") +
  scale_shape_manual(values=c(19,19,19))
```



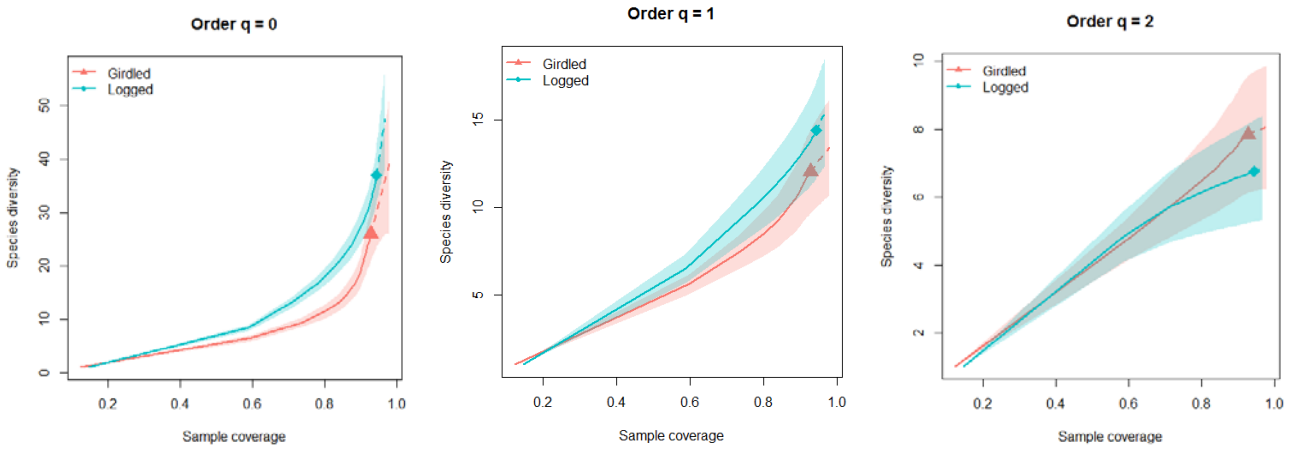
The scale of the Y-axis can be made to be free by the following code:

```
# free the scale of axis
ggiNEXT(out, type=1, facet.var="order") +
  facet_wrap(~order, scales="free")
```



Conventional plots can also be produced for each separate diversity order as shown below for coverage-based R/E curves: (Similar black-and-white plots can be made for the corresponding sample-size-based R/E curve and coverage-based curves.)

```
# Change to conventional plots
plot (out, type=3)
```

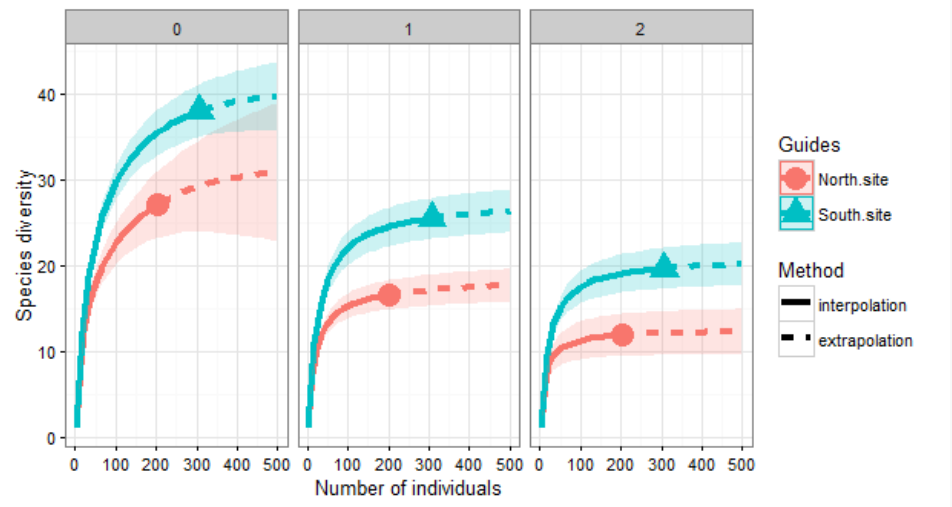


In addition to the `spider` data, we also include in `iNEXT` an abundance data set which is in the data.frame format. The data were collected in 2012 at Barrington Tops National Park, Australia. A total of 41 bird species from two sites (North-site and South-site) were observed; see Chao et al. (2015) for details.

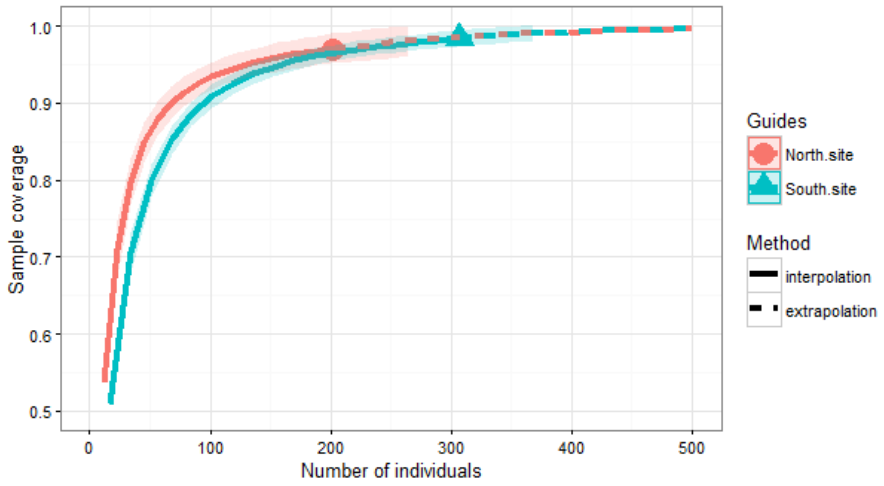
```
data(bird)
str(bird) # 41 species as rows, 2 sites as columns


'data.frame': 41 obs. of 2 variables:
 $ North.site: int  0 0 41 0 3 1 5 4 4 11 ...
 $ South.site: int  3 18 31 2 1 2 5 1 6 32 ...
```

We show the sample-size- and coverage-based sampling curves separately for each diversity order along with the sample completeness curve using the following codes.

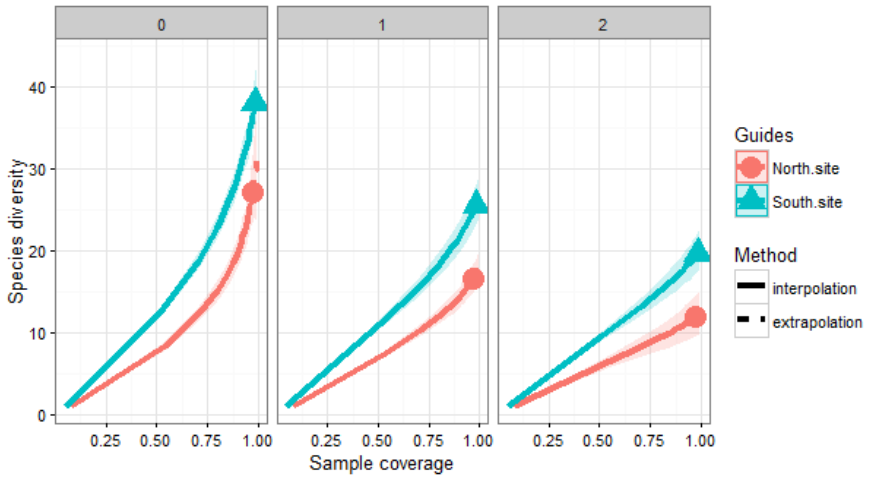
```
out1 <- iNEXT(bird, q=c(0, 1, 2), datatype="abundance", endpoint=500)
ggiNEXT(out1, type=1, facet.var="order") + theme_bw(base_size=10)
```



```
ggiNEXT(out1, type=2) + theme_bw(base_size=10)+ ylim(c(0.5, 1))
```




```
ggiNEXT(out1, type=3, facet.var="order") + theme_bw(base_size=10)
```



POINT ESTIMATION FUNCTION: estimateD()

We also supply the function

```
estimateD(x, datatype="abundance", base="size", level=NULL)
```

to compute diversity estimates of order q = 0, 1, 2 for any particular `level` of sample size (`base="size"`) or any specified `level` of sample coverage (`base="coverage"`) for either abundance data (`datatype="abundance"`) or incidence data (`datatype="incidence_freq" or "incidence_raw"`). If `level=NULL`, this function computes the diversity estimates for the minimum sample size/coverage among all sites.

For example, the classic rarefaction method involves rarefying all sample sizes to the minimum sample size, and then comparing the diversities for the minimum sample size. For the `spider` data, the sample sizes for the Girdled and Logged sites are respectively 168 and 252; thus classic rarefaction is to down-sample the Logged data to a size of 168. The following commands return the corresponding diversities of three orders (`q=0, 1` and `2`) along with sample coverage (`SC`) for the size of 168:

```
estimateD(spider, datatype="abundance", base="size", level=NULL)
```

```
         m      method    SC  q = 0  q = 1 q = 2
Girdled 168 observation 0.929 26.000 12.060 7.840
Logged  168 rarefaction 0.927 31.707 13.745 6.685
```

The sample completeness for the Girdled and Logged sites are respectively 92.89% and 94.46%. As with classic rarefaction, we can also rarefy the Logged data to the lower coverage value; here we can only rarefy to the closet value of 92.90% due to the constraint that the sample size must be an integer. The following commands return the diversity estimates along with the required sample size for the standardized coverage:

```
estimateD(spider, datatype="abundance", base="coverage", level=NULL)
```

```
          m      method     SC  q = 0 q = 1 q = 2
Girdled 168 observation 0.9289 26.000 12.06 7.840
Logged  175 rarefaction 0.9290 32.213 13.82 6.694
```

RAREFACTION/EXTRAPOLATION VIA EXAMPLES (INCIDENCE DATA)

Two incidence data sets (`ant` and `ciliates`) with different input formats are included in the `iNEXT` package. For illustration, we use the tropical ant data collected at five elevations (50m, 500m, 1070m, 1500m, and 2000m) in Costa Rica by Longino & Colwell (2011). The 5 lists of incidence frequencies are shown below. The first entry of each list must be the total number of sampling units, followed by the species incidence frequencies.

```
data(ant)
str(ant)

List of 5
 $ h50m  : num [1:228] 599 330 263 236 222 195 186 183 182 129 ...
 $ h500m : num [1:242] 230 133 131 123 78 73 65 60 60 56 ...
 $ h1070m: num [1:123] 150 99 96 80 74 68 60 54 46 45 ...
 $ h1500m: num [1:57] 200 144 113 79 76 74 73 53 50 43 ...
 $ h2000m: num [1:15] 200 80 59 34 23 19 15 13 8 8 ...
```

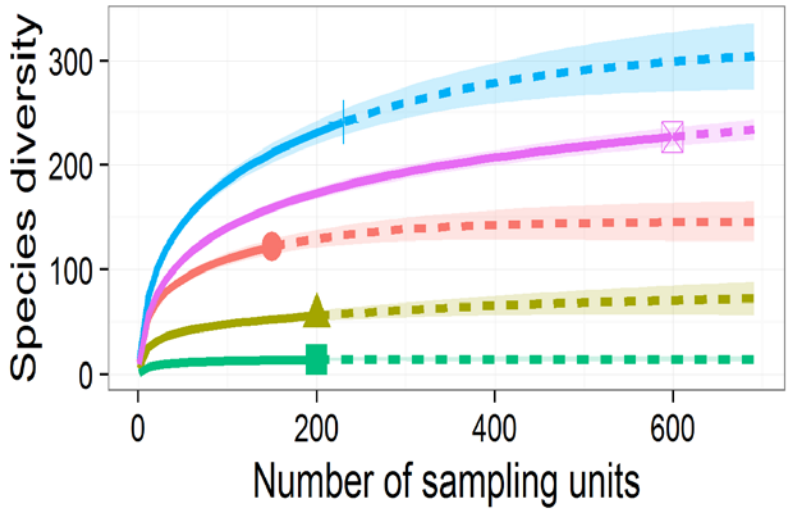
For `incidence` data, the list `$DataInfo` includes the site name (`site`), reference sample size (`T`), observed species richness (`S.obs`), total number of incidences (`U`), a sample coverage estimate (`SC`), and the first ten incidence frequency counts (`Q1-Q10`).

```
$DataInfo: basic data information
    site   T    U S.obs     SC Q1 Q2 Q3 Q4 Q5 Q6 Q7 Q8 Q9 Q10
1   h50m 599 5976   227 0.9918 49 23 18 14  9 10  4  8  6   2
2  h500m 230 2943   241 0.9760 71 34 12 14  9 11  8  4  7   5
3 h1070m 150 1730   122 0.9839 28 16 13  3  1  3  6  1  1   1
4 h1500m 200 1170    56 0.9889 13  4  2  2  4  2  0  0  4   0
5 h2000m 200  271    14 0.9964  1  2  1  1  0  0  0  2  0   0
```

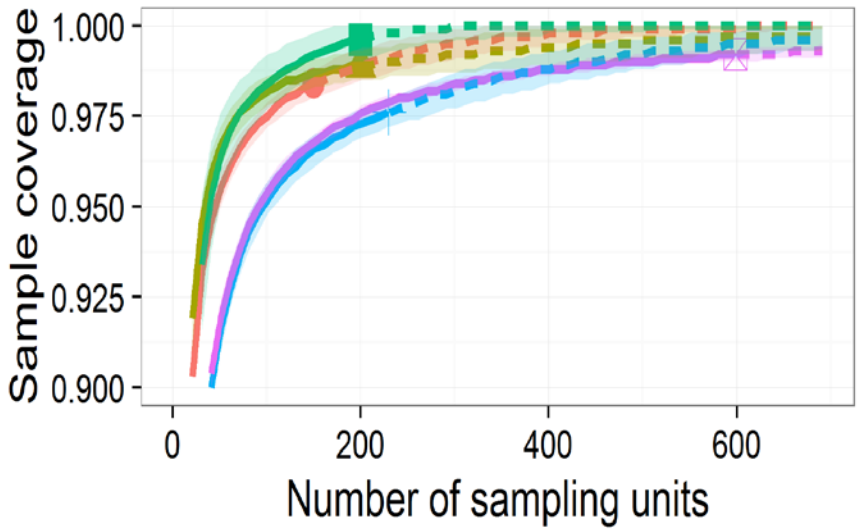
All running procedures are parallel to those for abundance data, except that the datatype is changed to datatype="incidence_freq" or "incidence_raw" as shown below. As described earlier, `theme_bw()` is a `ggplot2` function used to modify the display setting from the default gray background to a black-and-white theme. The following commands return three types of R/E sampling curves for ant data using a black-and-white theme with the figure legend placed at the bottom of all three curves.

```
t <- seq(1, 700, by=10)
out.inc <- iNEXT(ant, q=0, datatype="incidence_freq", size=t)
```

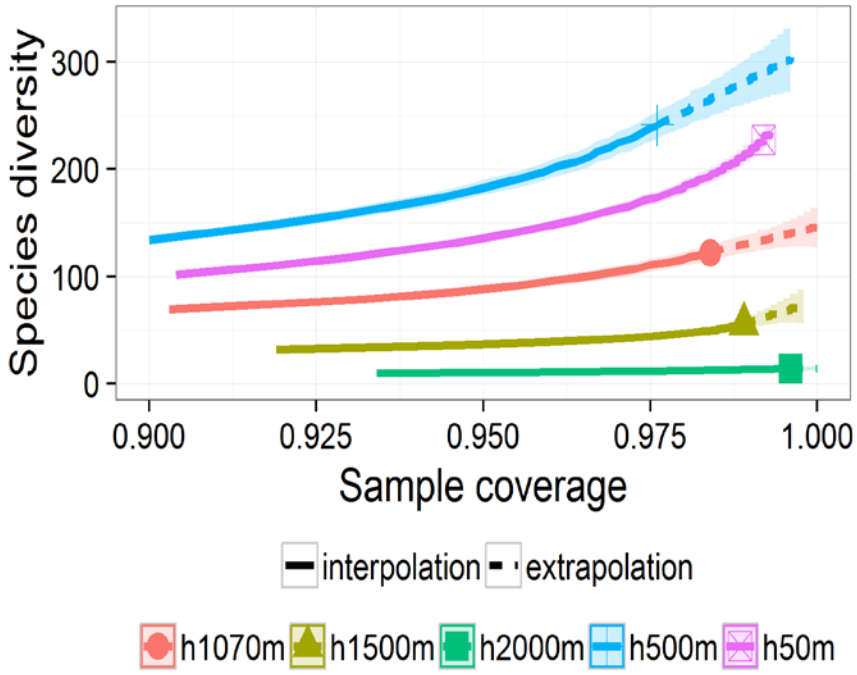
```
# Sample-size-based R/E curves
ggiNEXT(out.inc, type=1) +
  theme_bw(base_size = 18) + theme(legend.position="none")
```




```
# Sample completeness curves
ggiNEXT(out.inc, type=2) +ylim(c(0.9,1)) +
  theme_bw(base_size = 18) + theme(legend.position="none")
```



```
# Coverage-based R/E curves
ggiNEXT(out.inc, type=3) + xlim(c(0.9,1)) +
  theme_bw(base_size = 18) +
  theme(legend.position="bottom", legend.title=element_blank())
```



For the `ant` data, we can also apply the `estimateD()` function to obtain diversity estimates of order q = 0, 1, 2 for any particular `level` of sample size (`base="size"`) or any specified `level` of sample coverage (`base="coverage"`) when the data type is changed to `datatype="incidence_freq"`. For example, the following command returns the species diversity with a specified level of sample coverage of 98.5% for the ant data. For some sites, this coverage value corresponds to rarefaction whereas for others, it corresponds to extrapolation, as indicated in the `method` column of the output.

```
estimateD(ant, datatype="incidence_freq", base="coverage", level=0.985)
```

```
        t        method     SC   q = 0   q = 1  q = 2
h50m   327   rarefaction 0.9850 197.463  78.051 50.461
h500m  343 extrapolation 0.9850 268.753 103.844 64.759
h1070m 159 extrapolation 0.9850 123.617  59.592 41.775
h1500m 126   rarefaction 0.9850  50.482  26.249 18.649
h2000m 105   rarefaction 0.9851  12.917   7.712  5.795
```

When species data only consist of incidence frequency counts (Q_1, Q_2, ..., Q_T), where T denotes the total number of sampling units; see the output in the list `$DataInfo` for the `ant` data above. In this case, the incidence frequency counts must be converted to species incidences in order to fit with the argument `datatype="incidence_freq"`. As an example, the incidence counts for the `ant` data are given in Table 6 of Colwell et al. (2012), the following code will convert the incidence counts to `iNEXT` input data:

```
# Convert incidence frequency counts to species incidence data
h2000m=rep(c(1:4,8,13,15,19,23,34,59,80), c(1,2,1,1,2,1,1,1,1,1,1,1))

h1500m=rep(c(1:6,9,11,17,18,19,23,24,25,29,30,32,33,43,50,53,73,74,76,79,1
13,144),c(13,4,2,2,4,2,4,2,2,1,1,2,1,3,rep(1,13)))

h1070m=rep(c(1:16,18,19,21:26,30,31,32,34,36,38,39,43,45,46,54,60,68,74,80,
96,99),c(28,16,13,3,1,3,6,1,1,1,4,3,4,1,1,4,1,2,1,1,1,1,3,1,1,3,1,1,1,1,1,
2,2,rep(1,8)))

h500m=rep(c(1:20,21,23:27,30:34,36:39,41:47,49,52,53,54,56,60,65,73,78,123,
131,133),c(71,34,12,14,9,11,8,4,7,5,2,3,4,2,1,2,4,1,1,1,2,1,1,3,1,1,1,2,2,
1,1,1,1,4,2,rep(1,8),2,1,1,1,2,rep(1,6)))

h50m=rep(c(1:23,25,27,29,30,31,33,39,40,43,46,47,48,51,52,56,58,61,65,69,7
2,77,79,82,83,84,86,91,95,97,98,106,113,124,126,127,128,129,182,183,186,19
5,222,236,263,330),c(49,23,18,14,9,10,4,8,6,2,1,2,2,5,2,4,3,2,2,3,1,1,2,1,
2,1,2,rep(1,5),2,1,1,1,2,2,2,2,rep(1,12),2,rep(1,6),2,rep(1,8)))

ant=list(h50m=c(599,h50m),h500m=c(230, h500m),h1070m=c(150, h1070m),h1500m
=c(200, h1500m),h2000m=c(200,h2000m))

out.inc <- iNEXT(ant, q=0, datatype="incidence_freq")
```

Then the converted data are the same as those included in the `ant` set included in the `iNEXT` package.

Note that `datatype="incidence_raw"` is a new feature in `iNEXT` version 2.0.6. We here demonstrate its use via the `ciliates` data. A total of 51 soil samples were taken from three sites (15 samples from Southern Namib Desert, 17 samples from Central Namib Desert and 19 samples from Etosha Pan) in Namibia. The presence/absence of soil for each ciliate species was recorded for any sample, and a total of 331 species were found in the data; see Foissner, Agatha & Berger (2002) for details.

The data set `ciliates` included in the package is a `list` of three matrices; each `list` corresponds to a species by sites incidence (presence/absence) records in `matrix` input format. Running the following commands will result in the output graphics. All plots are omitted here.

```r
data(ciliates)
str(ciliates)
```

```
List of 3
$ EtoshaPan          : int [1:365, 1:19] 0 0 0 0 0 0 0 0 0 0 ...
$ CentralNamibDesert : int [1:365, 1:17] 0 0 0 0 0 1 0 0 0 0 ...
$ SouthernNamibDesert: int [1:365, 1:15] 0 0 0 0 0 0 0 0 0 0 ...
```

```r
out2 <- iNEXT(ciliates, q=c(0,1,2), datatype="incidence_raw")

ggiNEXT(out2, facet.var="order", type=1)
```

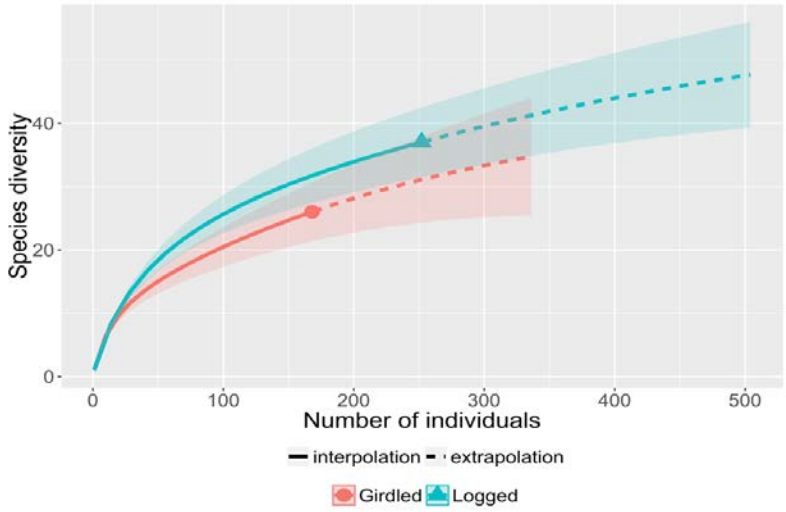
GENERAL CUSTOMIZATION

The data visualization package *ggplot2* provides the `scale_` function to customize data which is mapped into an aesthetic property of a `geom_`. The following functions can be **used to customize the `ggiNEXT` output.**

- Change point shape: `scale_shape_manual`
- Change line type : `scale_linetype_manual`
- Change line color: `scale_colour_manual`
- Change band color: `scale_fill_manual`
  see quick reference for style setting.

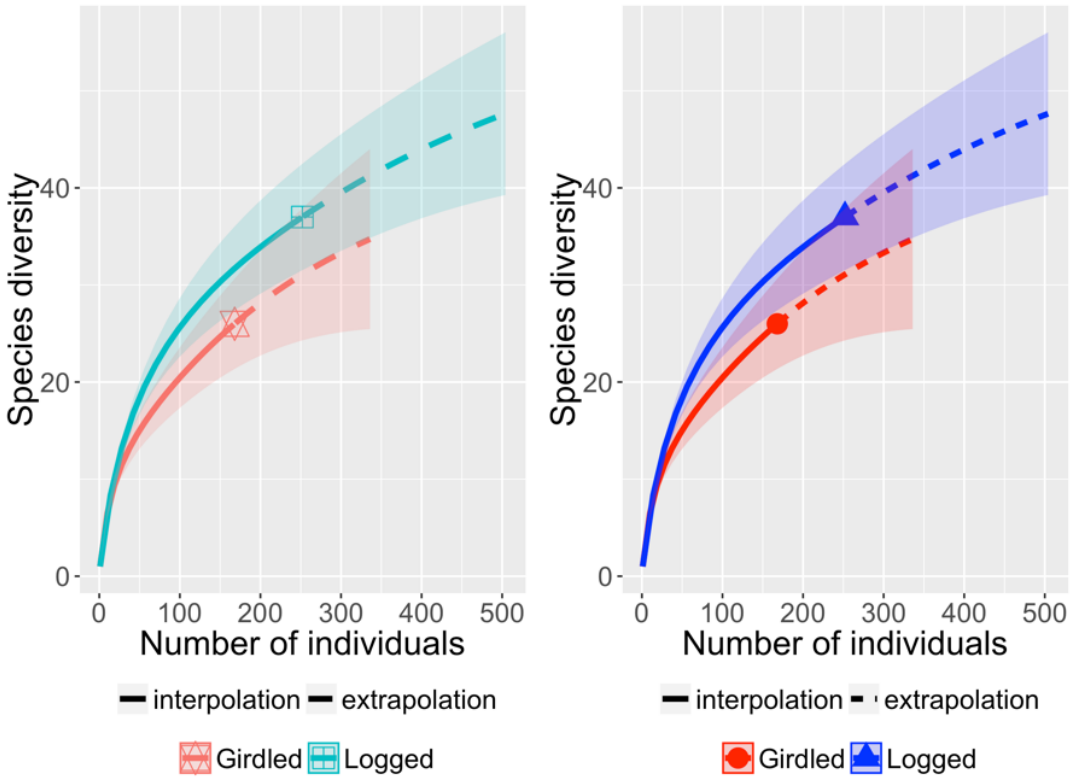
For illustrative purposes, we first provide the default sample-size-based R/E curves for the species richness of the `spider` abundance data. Then we show how the `ggiNEXT` output for the same data can be customized.

```
library(iNEXT)
library(ggplot2)
library(gridExtra)
library(grid)
data("spider")
out <- iNEXT(spider, q=0, datatype="abundance")
g <- ggiNEXT(out, type=1)
g
```



Change point shapes, line types and colors

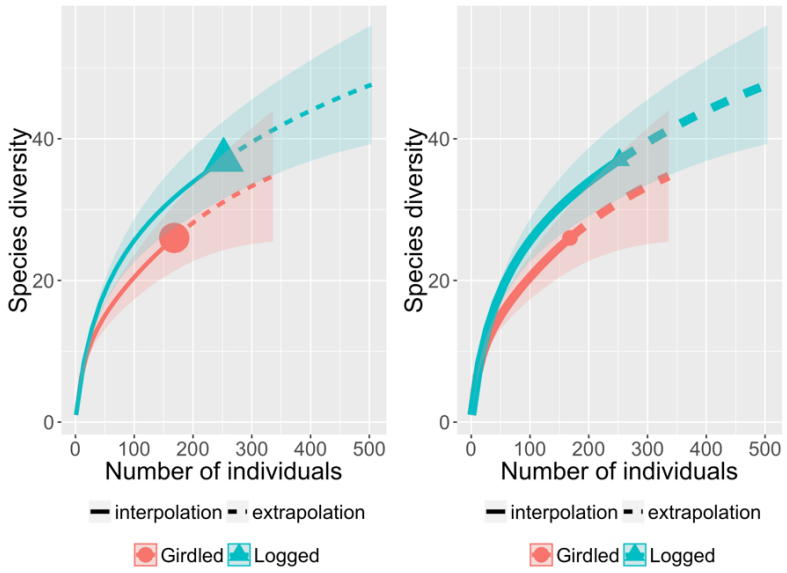
```
g1 <- g + scale_shape_manual(values=c(11, 12)) +
          scale_linetype_manual(values=c(1,2))
g2 <- g + scale_colour_manual(values=c("red", "blue")) +
          scale_fill_manual(values=c("red", "blue"))

# Draw multiple graphical objects on a page
# library(gridExtra)
grid.arrange(g1, g2, ncol=2)
```



Customizing point/line size

In order to change the size of the reference sample point or the rarefaction/extrapolation curve, users can modify the `ggplot` object.

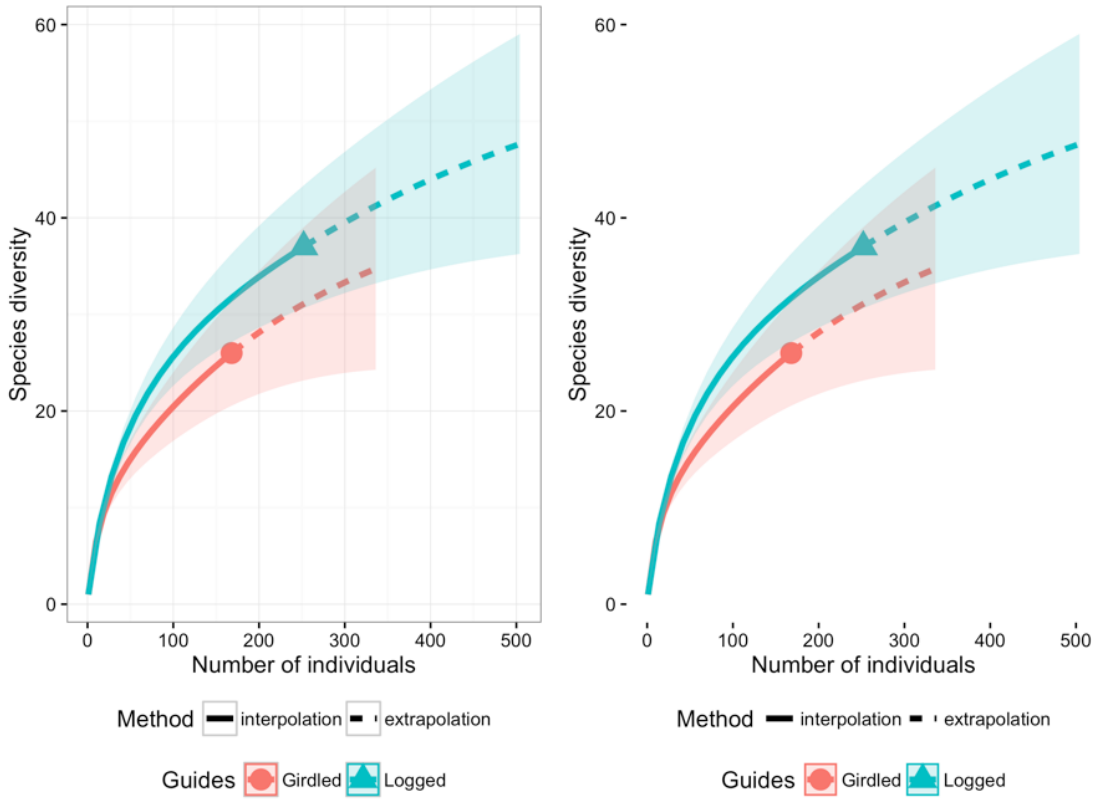
```
# Left panel: change reference-sample-point size to 10 (default size is 5)
gb3 <- ggplot_build(g)
gb3$data[[1]]$size <- 10
gt3 <- ggplot_gtable(gb3)

# use grid.draw to draw the graphical object
# library(grid)
# grid.draw(gt3)



# Right panel: change line size to 3 (default size is 1.5)
gb4 <- ggplot_build(g)
gb4$data[[2]]$size <- 3
gt4 <- ggplot_gtable(gb4)
# grid.draw(gt4)

grid.arrange(gt3, gt4, ncol=2)
```



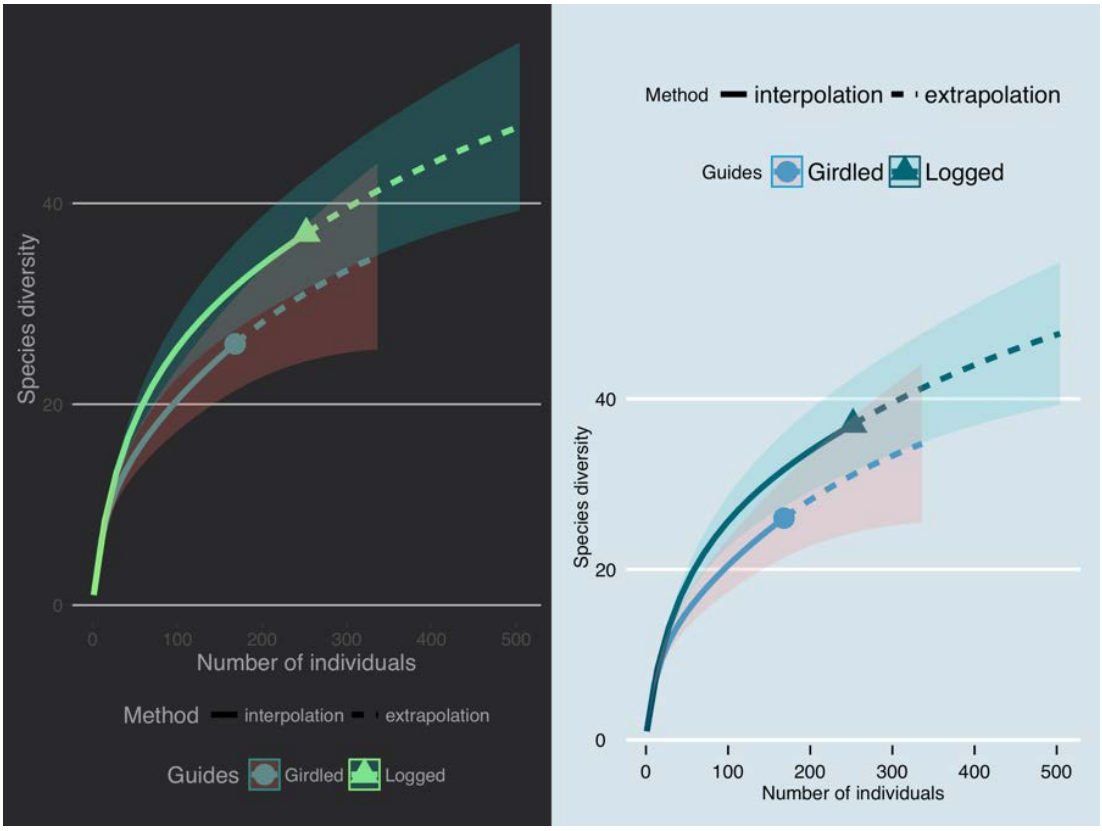
Customizing theme

Users can run `help(theme_grey)` to show the default themes in `ggplot2`. Some examples are shown below. More additional themes are provided by the *ggthemes* package.

```
# Left panel: change to black-and-white theme
g5 <- g + theme_bw() + theme(legend.position="bottom") +
 theme(legend.position="bottom"

# Right panel: change to classic black-and-white theme
g6 <- g + theme_classic()+ theme(legend.position="bottom")
grid.arrange(g5, g6, ncol=2)
```



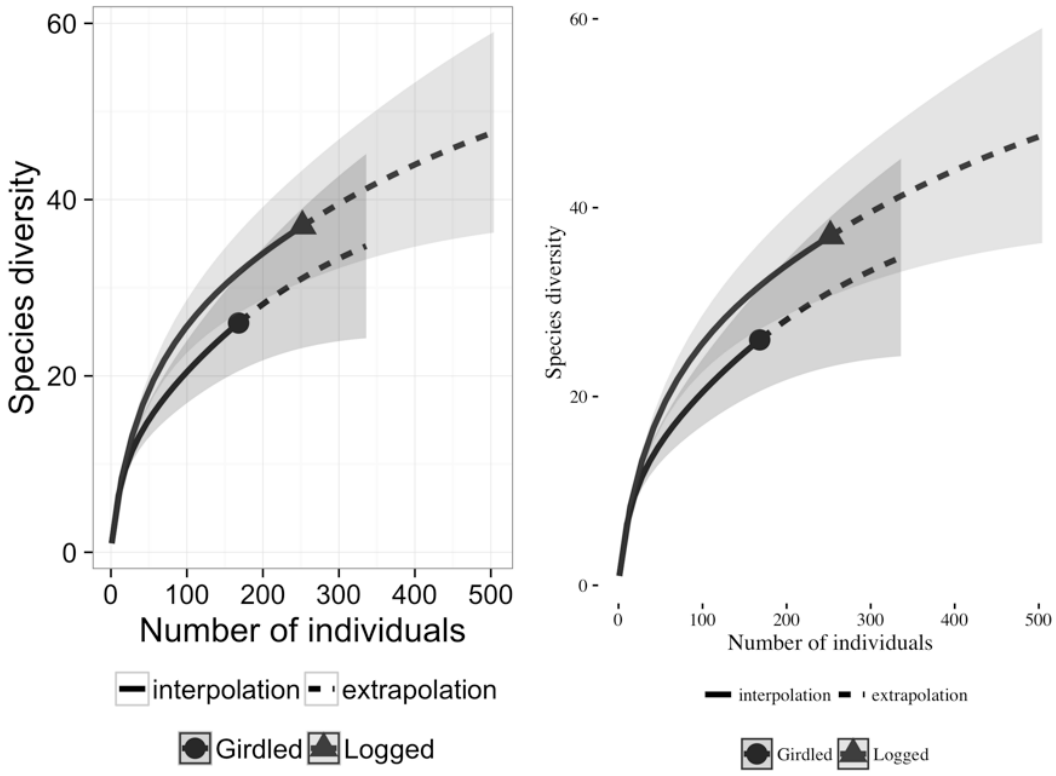
```r
# More themes

library(ggthemes)
g7 <- g + theme_hc(bgcolor = "darkunica") +
          scale_colour_hc("darkunica")

g8 <- g + theme_economist() + scale_colour_economist()
grid.arrange(g7, g8, ncol=2)
```



Black-White theme

The following are customized themes for black-and-white figures. For information on how to modify legends, see Cookbook for R.

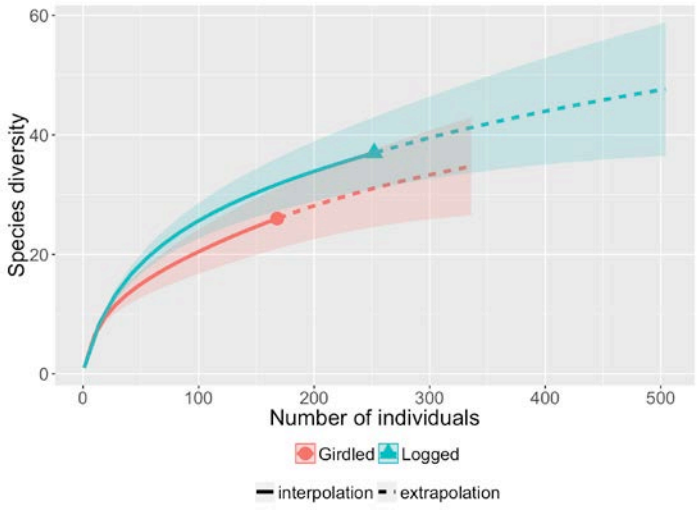
```
g9 <- g + theme_bw(base_size = 18) +
      scale_fill_grey(start = 0, end = .4) +
      scale_colour_grey(start = .2, end = .2) +
      theme(legend.position="bottom",
            legend.title=element_blank())

g10 <- g + theme_tufte(base_size = 12) +
    scale_fill_grey(start = 0, end = .4) +
    scale_colour_grey(start = .2, end = .2) +
    theme(legend.position="bottom",
          legend.title=element_blank())
grid.arrange(g9, g10, ncol=2)
```



Draw R/E curves by yourself

In *iNEXT*, we provide a S3 `ggplot2::fortify` method for the class `iNEXT`. The function `fortify` offers a single plotting interface for rarefaction/extrapolation curves. Set argument `type = 1, 2, 3` to plot the corresponding rarefaction/extrapolation curves.

```
df <- fortify(out, type=1)
head(df)
   datatype plottype    site       method order  x      y  y.lwr  y.upr
1 abundance        1 Girdled interpolated     0  1  1.000  1.000  1.000
2 abundance        1 Girdled interpolated     0 10  6.479  6.063  6.894
3 abundance        1 Girdled interpolated     0 19  9.450  8.635 10.265
4 abundance        1 Girdled interpolated     0 28 11.514 10.327 12.701
5 abundance        1 Girdled interpolated     0 37 13.127 11.595 14.659
6 abundance        1 Girdled interpolated     0 47 14.622 12.733 16.511

df.point <- df[which(df$method=="observed"),]
df.line <- df[which(df$method!="observed"),]
df.line$method <- factor(df.line$method,
                         c("interpolated", "extrapolated"),
                         c("interpolation", "extrapolation"))

ggplot(df, aes(x=x, y=y, colour=site)) +
  geom_point(aes(shape=site), size=5, data=df.point) +
  geom_line(aes(linetype=method), lwd=1.5, data=df.line) +
  geom_ribbon(aes(ymin=y.lwr, ymax=y.upr,
                  fill=site, colour=NULL), alpha=0.2) +
  labs(x="Number of individuals", y="Species diversity") +
  theme(legend.position = "bottom",
        legend.title=element_blank(),
        text=element_text(size=18))
```



LICENSE

The iNEXT package is licensed under the GPLv3. If you would like to provide any feedback or suggestions on how to help refine `iNEXT`, please contact Anne Chao (chao@stat.nthu.edu.tw) or report an issue on `iNEXT` github reop).

REFERENCES

Chao, A., Chiu, C.-H., Hsieh, T. C., Davis, T., Nipperess, D. & Faith, D. (2015) Rarefaction and extrapolation of phylogenetic diversity. *Methods in Ecology and Evolution*, 6, 380–388

Chao, A., Gotelli, N.J., Hsieh, T.C., Sander, E.L., Ma, K.H., Colwell, R.K. & Ellison, A.M. (2014) Rarefaction and extrapolation with Hill numbers: a framework for sampling and estimation in species diversity studies. *Ecological Monographs*, **84**, 45–67.

Chao, A. & Jost, L. (2012) Coverage-based rarefaction and extrapolation: standardizing samples by completeness rather than size. *Ecology*, **93**, 2533–2547.

Colwell, R.K., Chao, A., Gotelli, N.J., Lin, S.-Y., Mao, C.X., Chazdon, R.L. & Longino, J.T. (2012) Models and estimators linking individual-based and sample-based rarefaction, extrapolation and comparison of assemblages. *Journal of Plant Ecology*, **5**, 3–21.

Ellison, A.M., Barker-Plotkin, A.A., Foster, D.R. & Orwig, D.A. (2010) Experimentally testing the role of foundation species in forests: the Harvard Forest Hemlock Removal Experiment. *Methods in Ecology and Evolution*, **1**, 168–179.

Foissner, W., Agatha, S. & Berger, H. (2002) Soil ciliates (protozoa, ciliophora) from Namibia (Southwest Africa), with emphasis on two contrasting environments, the Etosha Region and the Namib Desert. *Denisia*, 5, 1–1459.

Hsieh, T.C., Ma, K.H. & Chao, A. (2016) iNEXT: An R package for interpolation and extrapolation of species diversity (Hill numbers). Under revision, *Methods in Ecology and Evolution*.

Longino, J.T. & Colwell, R.K. (2011) Density compensation, species composition, and richness of ants on a neotropical elevational gradient. *Ecosphere*, **2**:art29.