

Package ‘iSTAY’

October 28, 2025

Type Package

Title Information-Based Stability and Synchrony Measures

Version 1.0.0

Date 2025-10-16

Author Anne Chao [aut, cre],
Yu-Ting Huang [aut],
Johnny Shia [aut]

Maintainer Anne Chao <chao@stat.nthu.edu.tw>

Description Provides functions to compute a continuum of information-based measures for quantifying the temporal stability of populations, communities, and ecosystems, as well as their associated synchrony, based on species (or species assemblage) biomass or other key variables. When biodiversity data are available, the package also enables the assessment of the corresponding diversity–stability relationships. All measures are applicable in both temporal and spatial contexts. The theoretical and methodological background is detailed in Chao et al. (2025)
[<doi:10.1101/2025.08.20.671203>](https://doi.org/10.1101/2025.08.20.671203).

License GPL (>= 3)

Imports ggplot2 (>= 3.5.0), stringr, lmerTest, lme4, dplyr, ggpublisher

Encoding UTF-8

LazyData true

Depends R (>= 4.1.0)

Suggests knitr, rmarkdown

VignetteBuilder knitr

RoxygenNote 7.3.3

NeedsCompilation no

Repository CRAN

Date/Publication 2025-10-28 08:30:02 UTC

Contents

iSTAY-package	2
Data_Jena_20_metacommunities	3
Data_Jena_462_populations	4
Data_Jena_76_metapopulations	5
Data_Jena_hierarchical_structure	5
ggiSTAY_analysis	6
ggiSTAY_qprofile	8
iSTAY_Hier	9
iSTAY_Multiple	11
iSTAY_Single	12

Index	14
--------------	-----------

iSTAY-package	<i>Information-based stability and synchrony measures.</i>
---------------	--

Description

iSTAY (information-based stability and synchrony measures) is an R package that provides functions to compute a continuum of information-based measures for quantifying the temporal stability of populations, communities, and ecosystems, as well as their associated synchrony, based on species (or species assemblage) biomass or other key variables. When biodiversity data are available, the package also enables the assessment of the corresponding diversity–stability relationships. The information-based measures are derived from Hill numbers parameterized by an order $q > 0$; see Chao et al. (2025) for the theoretical and methodological background. All measures are illustrated using temporal biomass data from the Jena experiment (Roscher et al. 2004; Weisser et al. 2017; Wagg et al. 2022). See the iSTAY vignette for examples with output.

Specifically, iSTAY provides the following measures for three types of analyses:

- (1) **Single time series:** computes stability measures of order $q > 0$ and displays the corresponding stability profile for a time series (or for each time series analyzed individually). The stability profile illustrates how stability varies with the order q . When biodiversity data are available, iSTAY also assesses the diversity–stability relationship across individual time series.
- (2) **Multiple time series:** computes four measures—gamma, alpha, and beta stability, as well as synchrony—and displays the corresponding profiles for multiple time series (or for each set of time series within a collection). When biodiversity data are available, iSTAY also assesses the diversity–stability and diversity–synchrony relationships.
- (3) **Hierarchical Series:** computes four measures—gamma, alpha, and beta stability, as well as synchrony—for each hierarchical level, and provides the corresponding stability and synchrony profiles.

This package contains five main functions:

1. iSTAY_Single calculates stability measures of order $q > 0$ for a single time series of biomass or other relevant variables.
2. iSTAY_Multiple computes gamma, alpha, and beta stability, as well as synchrony for multiple time series of biomass or other relevant variables.
3. iSTAY_Hier computes gamma, alpha, and beta stability, as well as synchrony, at each hierarchical level for time series of biomass or other variables.
4. ggiSTAY_qprofile plots the stability and synchrony profiles based on the output obtained from the functions iSTAY_Single, iSTAY_Multiple or iSTAY_Hier.
5. ggiSTAY_analysis plots the diversity–stability and diversity–synchrony relationships based on the output obtained from the function iSTAY_Single or iSTAY_Multiple.

References

Chao, A., Colwell, R. K., Shia J., Thorn, S., Yang, M.-Y., Mitesser, O., et al. (2025) A continuum of information-based temporal stability measures and their decomposition across hierarchical level. BioRxiv <<https://doi.org/10.1101/2025.08.20.671203>>

Roscher C. Schumacher, J., Baade, J., Wilcke, W., Gleixner, G., Weisser, W. W. et al. (2004). The role of biodiversity for element cycling and trophic interactions: an experimental approach in a grassland community. Basic and Applied Ecology, 5, 107–121.

Wagg, C., Roscher, C., Weigelt, A., Vogel, A., Ebeling, A., De Luca, E. et al. (2022) Biodiversity–stability relationships strengthen over time in a long-term grassland experiment. Nature Communications, 13, 7752.

Weisser, W. W., Roscher, C., Meyer, S. T., Ebeling, A., Luo, G., Allan, E. et al. (2017). Biodiversity effects on ecosystem functioning in a 15-year grassland experiment: Patterns, mechanisms, and open questions. Basic and Applied Ecology, 23, 1–73.

Description

Because the Jena Experiment includes four blocks and five richness levels, all 20 block-richness combinations are considered, resulting in 20 sets of plots. In other words, all 76 plots are grouped into 20 sets based on their block identifiers and species richness levels. Each set is treated as a metacommunity, comprising three or four plots (communities). All plots within a metacommunity share the same number of species but differ in species composition.

Usage

```
data(Data_Jena_20_metacommunities)
```

Format

Data_Jena_20_metacommunities contains 20 lists; each list is a data frame with plot identifiers as rows and time points as columns.

Data_Jena_462_populations

Biomass time series data of 462 populations in the Jena experiment

Description

In the Jena experiment, the biomass of each species sown in a plot was recorded annually from 2003 to 2024, except in 2004. This dataset includes the biomass time series of all 462 populations across 76 plots.

Usage

```
data(Data_Jena_462_populations)
```

Format

Data_Jena_462_populations is a population-by-time (462×21) data frame with population identifiers as rows and time points as columns.

Data_Jena_76_metapopulations

Biomass time series data of 76 sets of populations in the Jena experiment

Description

The biomass data for all species sown within each plot are used to form 76 metapopulations. Because the number of species sown per plot ranges from 1 to 16, the number of populations within the 76 plots also ranges from 1 to 16. Each metapopulation dataset therefore includes all population-level records for its constituent species.

Usage

```
data(Data_Jena_76_metapopulations)
```

Format

`Data_Jena_76_metapopulations` contains 76 lists; each list is a data frame with population identifiers as rows and time points as columns.

Data_Jena_hierarchical_structure

Four-level hierarchical structure of the Jena Experiment

Description

All plots in the Jena Experiment were arranged into 4 blocks, each containing 18–20 plots. Plots were sown along a gradient of 1, 2, 4, 8, or 16 species, comprising a total of 462 populations. The biomass of each species sown in a plot was recorded annually from 2003 to 2024, except in 2004. The entire dataset forms a four-level hierarchy: Level 1: population, Level 2: community, Level 3: block, and level 4: overall data. This dataset describes the four-level structure. Each row corresponds to a population identifier (matching the same row `Jena_hierarchical_data`), and records the names of block, plot and species to which that population belongs.

Usage

```
data(Data_Jena_hierarchical_structure)
```

Format

`Data_Jena_hierarchical_structure` is a population-by-level (462×3) data frame with population identifiers as rows and hierarchical level ("block", "plot" and "species") as columns.

ggiSTAY_analysis *ggplot2 extension for plotting diversity-stability and diversity-synchrony relationships.*

Description

`ggiSTAY_analysis` is a graphical function based on the output from the functions `iSTAY_Single` or `iSTAY_Multiple`. It generates plots showing the relationships between stability (and synchrony if multiple time series are included) and an additional variable, such as diversity or another relevant factor.

Usage

```
ggiSTAY_analysis(output, x_variable, by_group = NULL, model = "LMM")
```

Arguments

<code>output</code>	The output obtained from <code>iSTAY_Single</code> or <code>iSTAY_Multiple</code> . It must include (or be combined with) a column corresponding to the variable specified in <code>x_variable</code> . If <code>by_group</code> is not <code>NULL</code> , must also include a column corresponding to the variable specified in <code>by_group</code> .
<code>x_variable</code>	The name of the column representing the diversity (or other) variable to be used as the x-axis in the plot.
<code>by_group</code>	The name of the column representing a categorical variable used to color points by group. The argument is required if <code>model = "LMM"</code> , as the model uses it as random effect for both intercept and slope. Default is <code>NULL</code> .
<code>model</code>	Specifies the fitting model. Use <code>model = "lm"</code> for a linear model; or <code>model = "LMM"</code> for a linear mixed model with random effects for both intercept and slope. Default is <code>model = "LMM"</code> .

Value

For an `iSTAY_Single` object, this function returns a figure showing the relationship between the diversity (or other) variable and stability.

For an `iSTAY_Multiple` object, this function returns a figure showing the diversity (or other) variable and gamma, alpha, and beta stability, as well as synchrony.

Examples

```
data("Data_Jena_20_metacommunities")
data("Data_Jena_76_metapopulations")
data("Data_Jena_462_populations")
data("Data_Jena_hierarchical_structure")

## Single time series analysis
# Analyze the stability of individual plots and diversity-stability
```

```

# relationship based on 76 plots
# See Example 2 in the iSTAY vignette for the output.
individual_plots <- do.call(rbind, Data_Jena_20_metacommunities)
output_individual_plots_div <- iSTAY_Single(data=individual_plots, order.q=c(1,2), Alltime=TRUE)
output_individual_plots_div <- data.frame(output_individual_plots_div,
                                           log2_sowndiv = log2(as.numeric(do.call(rbind,
                                           strsplit(output_individual_plots_div[,1],"[.]+"))[,2])),
                                           block = do.call(rbind,
                                           strsplit(output_individual_plots_div[,1],"[.]+"))[,1])

ggiSTAY_analysis(output = output_individual_plots_div, x_variable = "log2_sowndiv",
                 by_group="block", model="LMM")

# Analyze the stability of individual populations and diversity-stability
# relationships based on 462 populations
# See Example 4 in the iSTAY vignette for the output.
individual_populations <- do.call(rbind, Data_Jena_76_metapopulations)
output_individual_populations_div <- iSTAY_Single(data = individual_populations,
                                                   order.q=c(1,2), Alltime=TRUE)
output_individual_populations_div <- data.frame(output_individual_populations_div,
                                                 log2_sowndiv = log2(as.numeric(do.call(rbind,
                                                 strsplit(output_individual_populations_div[,1],"[.]+"))[,3])),
                                                 block=do.call(rbind,
                                                 strsplit(output_individual_populations_div[,1],"[.]+"))[,2])

ggiSTAY_analysis(output = output_individual_populations_div, x_variable = "log2_sowndiv",
                 by_group = "block", model = "LMM")

## Multiple time series analysis
# Analyze the stability and synchrony within each metacommunity
# and their relationships with diversity based on 20 metacommunities
# See Example 6 in the iSTAY vignette for the output.
metacommunities <- Data_Jena_20_metacommunities
output_metacommunities_div <- iSTAY_Multiple(data = metacommunities,
                                              order.q = c(1,2), Alltime = TRUE)
output_metacommunities_div <- data.frame(output_metacommunities_div,
                                           log2_sowndiv = log2(as.numeric(do.call(rbind,
                                           strsplit(output_metacommunities_div[, 1], "_"))[, 2])),
                                           block = do.call(rbind,
                                           strsplit(output_metacommunities_div[, 1], "_"))[, 1])

ggiSTAY_analysis(output = output_metacommunities_div, x_variable = "log2_sowndiv",
                 by_group = "block", model = "LMM")

# Analyze the stability and synchrony within each metapopulation
# and their relationships with diversity based on 76 metapopulations
# See Example 8 in the iSTAY vignette for the output.

metapopulations <- Data_Jena_76_metapopulations
output_metapopulations_div <- iSTAY_Multiple(data = metapopulations,
                                              order.q=c(1,2), Alltime=TRUE)
output_metapopulations_div <- data.frame(output_metapopulations_div,

```

```

log2_sowndiv = log2(as.numeric(do.call(rbind,
strsplit(output_metapopulations_div[,1],"[.]+"))[,3])),
block = do.call(rbind,
strsplit(output_metapopulations_div[,1],"_"))[,2])

ggiSTAY_analysis(output = output_metapopulations_div, x_variable = "log2_sowndiv",
by_group = "block", model = "LMM")

```

ggiSTAY_qprofile*ggplot2 extension for plotting stability and synchrony profiles.***Description**

`ggiSTAY_qprofile` is a graphical function based on the output from the function `iSTAY_Single`, `iSTAY_Multiple` or `iSTAY_Hier`. It generates stability (and synchrony, if multiple time series are included) profiles that depict how stability and synchrony vary with the order $q > 0$.

Usage

```
ggiSTAY_qprofile(output)
```

Arguments

output	the output obtained from <code>iSTAY_Single</code> , <code>iSTAY_Multiple</code> or <code>iSTAY_Hier</code> .
--------	---

Value

For an `iSTAY_Single` object, this function return a figure showing the stability profile.

For an `iSTAY_Multiple` object, it returns a figure displaying the profiles for gamma, alpha, and beta stability, as well as synchrony.

For an `iSTAY_Hier` object, it returns a figure displaying the profiles for gamma, alpha, and beta stability, as well as synchrony.

Examples

```

data("Data_Jena_20_metacommunities")
data("Data_Jena_76_metapopulations")
data("Data_Jena_462_populations")
data("Data_Jena_hierarchical_structure")

## Single time series analysis
# Plot the stability profiles of two selected plots
# See Example 1 in the iSTAY vignette for the output.

individual_plots <- do.call(rbind, Data_Jena_20_metacommunities)

```

```

output_two_plots_q <- iSTAY_Single(
  data = individual_plots[which(rownames(individual_plots)
    %in% c("B1_4.B1A04", "B4_2.B4A14")),],
  order.q = seq(0.1,2,0.1), Alltime = TRUE)
ggiSTAY_qprofile(output = output_two_plots_q)

# Plot the stability profiles of two selected populations
# See Example 3 in the iSTAY vignette for the output.
individual_populations <- do.call(rbind, Data_Jena_76_metapopulations)
output_two_populations_q <- iSTAY_Single(
  data = individual_populations[which(rownames(individual_populations)
    %in% c("B1A06_B1_16.BM_Ant.odo", "B1A06_B1_16.BM_Cam.pat")),],
  order.q = seq(0.1,2,0.1), Alltime = TRUE)
ggiSTAY_qprofile(output = output_two_populations_q)

## Multiple time series analysis
# Plot the gamma, alpha and beta stability profiles,
# as well as synchrony profiles for two selected metacommunities
# See Example 5 in the iSTAY vignette for the output.

metacommunities <- Data_Jena_20_metacommunities
output_two_metacommunities_q <- iSTAY_Multiple(
  data = metacommunities[which(names(metacommunities) %in% c("B1_1", "B3_2"))],
  order.q = seq(0.1,2,0.1), Alltime = TRUE)
ggiSTAY_qprofile(output = output_two_metacommunities_q)

# Plot the gamma, alpha and beta stability profiles,
# as well as synchrony profiles of two selected metapopulations
# See Example 7 in the iSTAY vignette for the output.

metapopulations <- Data_Jena_76_metapopulations
output_two_metapopulations_q <- iSTAY_Multiple(
  data = metapopulations[which(names(metapopulations) %in% c("B1A04_B1_4", "B4A14_B4_2"))],
  order.q = seq(0.1,2,0.1), Alltime = TRUE)
ggiSTAY_qprofile(output = output_two_metapopulations_q)

## Hierarchical time series analysis
# See Example 9 in the iSTAY vignette for the output.
output_hier_q <- iSTAY_Hier(data = Data_Jena_462_populations,
  structure = Data_Jena_hierarchical_structure,
  order.q = seq(0.1,2,0.1), Alltime = TRUE)
ggiSTAY_qprofile(output = output_hier_q)

```

Description

iSTAY_Hier computes gamma, alpha, and beta stability, as well as synchrony, at each hierarchical level for time series of biomass or other variables.

Usage

```
iSTAY_Hier(
  data,
  structure,
  order.q = c(1, 2),
  Alltime = TRUE,
  start_T = NULL,
  end_T = NULL
)
```

Arguments

data	A <code>data.frame</code> containing the hierarchical data, with sampling units as rows and time points as columns.
structure	The hierarchical structure of the input data.
order.q	A numerical vector specifying the orders of stability. Default is <code>c(1,2)</code> .
Alltime	Logical (TRUE or FALSE), indicating whether to use all time points in the data.
start_T	(Applicable only if <code>Alltime = FALSE</code>) a positive integer specifying the starting column (time point) for the analysis interval.
end_T	(Applicable only if <code>Alltime = FALSE</code>) a positive integer specifying the ending column (time point) for the analysis interval.

Value

a data frame with the following columns:
 Hier_level: hierarchical level (e.g. Level 1: population, Level 2: community, Level 3: block, and level 4: overall data)
 Order_q: order of stability or synchrony
 Gamma, Alpha, Beta: stability measures of order q
 Synchrony: synchrony measure of order q

Examples

```
data("Data_Jena_462_populations")
data("Data_Jena_hierarchical_structure")
output_hier <- iSTAY_Hier(data = Data_Jena_462_populations,
                           structure = Data_Jena_hierarchical_structure,
                           order.q=c(1,2), Alltime=TRUE)
output_hier
```

<code>iSTAY_Multiple</code>	<i>Calculate stability and synchrony for multiple time series.</i>
-----------------------------	--

Description

`iSTAY_Multiple` computes gamma, alpha, and beta stability, as well as synchrony, for multiple time-series data.

Usage

```
iSTAY_Multiple(
  data,
  order.q = c(1, 2),
  Alltime = TRUE,
  start_T = NULL,
  end_T = NULL
)
```

Arguments

<code>data</code>	A <code>data.frame</code> containing multiple time series data, with sampling units as rows and time points as columns, or a list of <code>data.frames</code> with each data frame representing multiple time series.
<code>order.q</code>	A numerical vector specifying the orders of stability and synchrony. Default is <code>c(1,2)</code> .
<code>Alltime</code>	Logical (TRUE or FALSE), indicating whether to use all time points in the data.
<code>start_T</code>	(Applicable only if <code>Alltime = FALSE</code>) a positive integer specifying the starting column (time point) for the analysis interval.
<code>end_T</code>	(Applicable only if <code>Alltime = FALSE</code>) a positive integer specifying the ending column (time point) for the analysis interval.

Value

a data frame with the following columns:
 Dataset: the input dataset
 Order_q: order of stability or synchrony
 Gamma, Alpha, Beta: stability measures of order q
 Synchrony: synchrony measure of order q

Examples

```
# Stability of multiple time series
data("Data_Jena_20_metacommunities")
metacommunities <- Data_Jena_20_metacommunities
output_metacommunities <- iSTAY_Multiple(data = metacommunities, order.q = c(1,2), Alltime = TRUE)
output_metacommunities
```

```
# Stability of metapopulations
data("Data_Jena_76_metapopulations")
metapopulations <- Data_Jena_76_metapopulations
output_metapopulations <- iSTAY_Multiple(data = metapopulations, order.q = c(1,2), Alltime = TRUE)
output_metapopulations
```

iSTAY_Single*Calculate stability for a single time series.***Description**

`iSTAY_Single` computes the stability of order q for a single time series.

Usage

```
iSTAY_Single(
  data,
  order.q = c(1, 2),
  Alltime = TRUE,
  start_T = NULL,
  end_T = NULL
)
```

Arguments

<code>data</code>	A vector of time series data, or a <code>data.frame</code> with sampling units as rows and time points as columns.
<code>order.q</code>	a numerical vector specifying the orders of stability. Default is <code>c(1,2)</code> .
<code>Alltime</code>	Logical (TRUE or FALSE), indicating whether to use all time points in the data.
<code>start_T</code>	(Applicable only if <code>Alltime = FALSE</code>) a positive integer specifying the starting column (time point) for the analysis interval.
<code>end_T</code>	(Applicable only if <code>Alltime = FALSE</code>) a positive integer specifying the ending column (time point) for the analysis interval.

Value

a dataframe with columns:
 Dataset: the input dataset
 Order_q: order of stability
 Stability: stability measures of order q

Examples

```
# Compute the stability of individual plots
data("Data_Jena_20_metacommunities")
individual_plots <- do.call(rbind, Data_Jena_20_metacommunities)
output_individual_plots <- iSTAY_Single(data = individual_plots,
                                         order.q=c(1,2), Alltime = TRUE)
output_individual_plots

# Compute the stability of individual populations
data("Data_Jena_76_metapopulations")
individual_populations <- do.call(rbind, Data_Jena_76_metapopulations)
output_individual_populations <- iSTAY_Single(data = individual_populations,
                                               order.q = c(1,2), Alltime = TRUE)
output_individual_populations
```

Index

* datasets

Data_Jena_20_metacommunities, [3](#)
Data_Jena_462_populations, [4](#)
Data_Jena_76_metapopulations, [5](#)
Data_Jena_hierarchical_structure,
[5](#)

* package

iSTAY-package, [2](#)

Data_Jena_20_metacommunities, [3](#)
Data_Jena_462_populations, [4](#)
Data_Jena_76_metapopulations, [5](#)
Data_Jena_hierarchical_structure, [5](#)

ggiSTAY_analysis, [6](#)

ggiSTAY_qprofile, [8](#)

iSTAY(iSTAY-package), [2](#)

iSTAY-package, [2](#)

iSTAY_Hier, [9](#)

iSTAY_Multiple, [11](#)

iSTAY_Single, [12](#)