**A Brief Guide to the iStay Package, and R code for Graphics in Chao et al. (2025) Manuscripts**

This guide introduces the main function in the iStay package and demonstrates how to make graphics shown in Figures 1 to 4 in Chao et al.’s (2025) manuscripts; all data and R code are available on Zenodo and in Anne Chao’s Github repository at <https://github.com/AnneChao/MS_iStay>. To run the code, it is required R version > 4.0.0. Before using the data and code, the following packages on CRAN must be installed and imported:

library(tidyverse)

library(lmerTest)

Next, install and import the package “**iStay”** from Anne Chao’s Github. Please make sure to update required packages to their latest version.

library(devtools)

install\_github("AnneChao/iStay")   # Press 'enter' key to skip update options

library(iStay)

In the following, we briefly introduce two functions in the “**iStay”** package.

####################################################################

Function “Stay\_Single”⸺ a function that calculate stability of the time series data (like biomass, productivity, etc.) for single assemblage.

Stay\_Single (data, order.q = c(1,2), Alltime = TRUE, start\_T = NULL, end\_T = NULL)

A description for each argument in the function (“Stay\_Single”) is given in the following table.

|  |  |
| --- | --- |
| **Argument** | **Description** |
| data | can be input as a vector of time series data, or data.frame (assemblages by times). |
| order.q | a numerical vector specifying the orders of stability. Default is c(1, 2). |
| Alltime | TRUE or FALSE, to decide whether to use all the times in (every) dataframe. |
| start\_T | (argument only for Alltime = FALSE) a positive integer specifying the start column of time in the data. |
| end\_T | (argument only for Alltime = FALSE) a positive integer specifying the end column of time in the data. |

Use ‘?Stay\_Single’ for help.

Function “Stay\_Multiple”⸺ a function that calculate (Gamma, Alpha and Beta) stability and synchrony of the time series data (like biomass, productivity, etc.) for multiple assemblages.

Stay\_Multiple (data, order.q = c(1,2), Alltime = TRUE, start\_T = NULL, end\_T = NULL)

A description for each argument in the function (“Stay\_Single”) is given in the following table.

|  |  |
| --- | --- |
| **Argument** | **Description** |
| data | can be input as a data.frame/matrix (assemblages by times), or a list of data.frames with each dataframe representing a assemblages-by-times data |
| order.q | a numerical vector specifying the orders of stability and synchrony. Default is c(1,2). |
| Alltime | TRUE or FALSE, to decide whether to use all the times in (every) dataframe. |
| start\_T | (argument only for Alltime = FALSE) a positive integer specifying the start column of time in the data. |
| end\_T | (argument only for Alltime = FALSE) a positive integer specifying the end column of time in the data. |

Use ‘? Stay\_Multiple’ for help.

##################################################################################

The source code for graphics (Figures 1 to 4) in Chao et al. (2025) paper is provided in Anne Chao’s Github repository. First, import/load the source code:

source("Source R code.txt")

1. **Figure 1. Gamma, alpha and synchrony profiles for q between 0 and 2 within each plot.**

Figure 1. Stability profiles at (a) the gamma/plot level, (b) the alpha/species level, and (c) species synchrony within a plot, for q between 0 and 2 based on the data from Plot B1A04 (4 species) and Plot B4A14 (2 species)

**1a. R code for Figure 1a**

This code extracts two specific biomass datasets from the RDA file “Jena\_species\_biomass\_data.rda” in the “iStay” package and stores them in a list df1. It then updates each data frame's row names by prefixing them with the corresponding list name, helping to retain source identity for later merging or analysis.

df1 <- list(B4A14\_2 = Jena\_species\_biomass\_data$B4A14\_B4\_2,

B1A04\_4 = Jena\_species\_biomass\_data$B1A04\_B1\_4)

df1 <- Map(function(x, nm) {

rownames(x) <- paste0(nm, rownames(x))

x

}, df1, names(df1))

To plot Figure 1, use the function “Stay\_Multiple” to calculate gamma stability, alpha stability, and synchrony for orders *q* ranging from 0.01 to 2 in increments of 0.01. Then, use function ‘fig\_1a’ (provided in the source code) to plot Figure 1 (a).

output\_fig\_1 <- Stay\_Multiple(df1, order.q = seq(0.01, 2, 0.01))

fig\_1a(output\_fig\_1)

**1b. R code for Figure 1b**

To plot the dashed lines in Figure 1 (b), use the function “Stay\_Single” to calculate stability for each species in the two plots. Then, use the function ‘fig\_1b’ (provided in the source code) to plot Figure 1 (b).

output\_fig\_1b <- list(

B1A04\_4 = Stay\_Single(df1$B1A04\_4, order.q = seq(0.01, 2, 0.01)),

B4A14\_2 = Stay\_Single(df1$B4A14\_2, order.q = seq(0.01, 2, 0.01))

) |> bind\_rows(.id = "Site")

fig\_1b(output\_fig\_1, output\_fig\_1b)

**1c. R code for Figure 1c**

To plot Figure 1 (c), use the function ‘fig\_1c’ (provided in the source code) to plot Figure 1 (c).

fig\_1c(output\_fig\_1)

##################################################################################

1. **Figure 2. Biodiversity–stability and biodiversity–synchrony relationships based on 76 plots**

Figure 2. Relationships between the logarithm of species richness and gamma stability (a), alpha stability (b), and species synchrony (c) for orders q = 0.5, 1 and 2, based on decomposition within individual plots/communities.

**2a. R code for Figure 2a**

This code parses metadata from the names of the RDA file "Jena\_species\_biomass\_data" in the "iStay" package to construct a structure2 data frame containing block identifiers and log-transformed species richness. It then filters out monocultures (log2\_sowndiv = 0) to create structure2c, and calculates stability metrics of the 76 individual plots using function "Stay\_Multiple" for three different orders (q = 0.5, 1, 2).

split\_names2 <- str\_split(names(Jena\_species\_biomass\_data), "\_", simplify = TRUE)

structure2 <- data.frame(

block = split\_names2[, 2],

log2\_sowndiv = log2(as.numeric(split\_names2[, 3]))

)

structure2c <- structure2 |> filter(log2\_sowndiv != 0)

output2 <- Stay\_Multiple(Jena\_species\_biomass\_data, order.q = c(0.5, 1, 2))

To plot Figure 2 (a), use the function “LMM\_2\_to\_4” to fit the linear mixed model of biodiversity on gamma stability, with block as a random effect. Then, use function ‘fig\_2\_or\_4’ (provided in the source code) to plot Figure 2 (a).

output\_fig\_2a <- LMM\_2\_to\_4(output2, structure = structure2, metric\_name = "Gamma")

fig2\_or\_4(output\_fig\_2a, metric\_name = "Gamma")

**2b. R code for Figure 2b**

To plot Figure 2 (b), use the function “LMM\_2\_to\_4” to fit the linear mixed model of biodiversity on alpha stability, with block as a random effect. Then, use function ‘fig\_2\_or\_4’ (provided in the source code) to plot Figure 2 (b).

output\_fig\_2b <- LMM\_2\_to\_4(output2, structure = structure2, metric\_name = "Alpha")

fig2\_or\_4(output\_fig\_2b, metric\_name = "Alpha")

**2c. R code for Figure 2c**

To plot Figure 2 (c), use the function “LMM\_2\_to\_4” to fit the linear mixed model of biodiversity on species synchrony, with block as a random effect. Then, use function ‘fig\_2\_or\_4’ (provided in the source code) to plot Figure 2 (c).

output\_fig\_2c <- LMM\_2\_to\_4(output2 |> filter(Synchrony != 1), structure = structure2c, metric\_name = "Synchrony")

fig2\_or\_4(output\_fig\_2c, metric\_name = "Synchrony")

##################################################################################

1. **Figure 3. Temporal effects of species richness on stability and synchrony based on** **12 consecutive overlapping 10-year moving window**

Figure 3. (Left panels) The average (a) gamma (plot-level) stability and (b) alpha (species-level) stability of order q = 0.5, 1 and 2 across all plots with the same number of species. The average species synchrony values (c) were obtained across plots with monoculture plots being excluded. (Right panels) The effects of species richness on (a) gamma stability, (b) alpha stability, and (c) species synchrony, represented by the slopes of each biodiversity–stability or biodiversity–synchrony relationship across 12 consecutive 10-year windows.

**3a. R code for Figure 3a left panel**

This code calculates mean stability metrics (gamma, alpha, and synchrony) across sliding 10-year windows (excluding years containing 2004) using the “Stay\_Multiple” function. For each window, it adds the starting year and log2-transformed species richness, then summarizes the results by the staring year of each window, diversity level, and diversity order q. The output Summary\_fig\_3\_left is used to plot the left panels of Figure 3.

split\_names3 <- str\_split(names(Jena\_species\_biomass\_data), "\_", simplify = TRUE)

year\_windows <- lapply(2003:2015, function(start) {

yrs <- if (start == 2003) c(2003, 2005:2013) else start:(start + 9)

if (2004 %in% yrs) return(NULL)

as.character(yrs)

}) |> compact()

names(year\_windows) <- as.character(c(2003, 2005:2015))

output\_fig\_3\_left <- lapply(names(year\_windows), function(start\_year) {

window\_years <- year\_windows[[start\_year]]

biomass\_data <- lapply(Jena\_species\_biomass\_data, \(df) df[, window\_years, drop = FALSE])

output3\_left <- Stay\_Multiple(biomass\_data, order.q = c(0.5, 1, 2))

output3\_left |>

mutate(

Start\_year = as.numeric(start\_year),

log2\_sowndiv = rep(as.numeric(split\_names3[, 3]), 3)

)

})

Summary\_fig\_3\_left <- bind\_rows(output\_fig\_3\_left) |>

mutate(Order\_q = paste0("q = ", Order\_q)) |>

group\_by(Start\_year, Order\_q, log2\_sowndiv) |>

summarise(

mean\_gamma = mean(Gamma),

mean\_alpha = mean(Alpha),

mean\_syn = mean(Synchrony),

.groups = "drop"

)

To plot the left panel of Figure 3 (a), use the function ‘fig\_3\_left’ (provided in the source code).

fig\_3\_left(Summary\_fig\_3\_left)$Gamma\_plot

**3b. R code for Figure 3b left panel**

To plot the left panel of Figure 3 (b), use the function ‘fig\_3\_left’ (provided in the source code).

fig\_3\_left(Summary\_fig\_3\_left)$Alpha\_plot

**3c. R code for Figure 3c left panel**

To plot the left panel of Figure 3 (c), use the function ‘fig\_3\_left’ (provided in the source code).

fig\_3\_left(Summary\_fig\_3\_left)$Synchrony\_plot

**3d. R code for Figure 3a right panel**

“structure3” is used to fit the linear mixed models for Figure 3 and is identical to “structure2”.

structure3 <- data.frame(

block = split\_names3[, 2],

log2\_sowndiv = log2(as.numeric(split\_names3[, 3]))

)

To plot the right panel of Figure 3 (a), use the function “slpoe\_3” to calculate the slopes values of the fixed effect based on the fitted linear mixed model of biodiversity on gamma stability, with block as a random effect, across 12 consecutive overlapping 10-year moving window. Then, use function ‘fig\_3\_right’ (provided in the source code) to plot Figure 3 (a).

output\_fig\_3a\_right <- slope\_3(metric\_name = "Gamma")

fig\_3\_right(output\_fig\_3a\_right)

**3e. R code for Figure 3b right panel**

To plot the right panel of Figure 3 (b), use the function “slpoe\_3” to calculate the slopes values of the fixed effect based on the fitted linear mixed model of biodiversity on alpha stability, with block as a random effect, across 12 consecutive overlapping 10-year moving window. Then, use function ‘fig\_3\_right’ (provided in the source code) to plot Figure 3 (b).

output\_fig\_3b\_right <- slope\_3(metric\_name = "Alpha")

fig\_3\_right(output\_fig\_3b\_right)

**3f. R code for Figure 3c right panel**

To plot the right panel of Figure 3 (c), use the function “slpoe\_3” to calculate the slopes values of the fixed effect based on the fitted linear mixed model of biodiversity on spatial synchrony, with block as a random effect, across 12 consecutive overlapping 10-year moving window. Then, use function ‘fig\_3\_right’ (provided in the source code) to plot Figure 3 (c).

output\_fig\_3c\_right <- slope\_3(metric\_name = "Synchrony")

fig\_3\_right(output\_fig\_3c\_right)

##################################################################################

1. **Figure 4. Biodiversity–stability and biodiversity–synchrony relationships based on 20 sets**

Figure 4. Relationships between the logarithm of species richness and gamma stability (a), alpha stability (b), and spatial synchrony among plots (c) for orders q = 0.5, 1 and 2, based on stability decomposition within each of 20 metacommunities.

**4a. R code for Figure 4a**

This code parses metadata from the names of the RDA file "Jena\_plot\_biomass\_data" in the "iStay" package to construct a structure4 data frame containing block identifiers and log-transformed species richness. It then calculates stability metrics of the 20 metacommunities using function "Stay\_Multiple" for three different orders (q = 0.5, 1, 2).

split\_names4 <- str\_split(names(Jena\_plot\_biomass\_data), "\_", simplify = TRUE)

structure4 <- data.frame(

block = split\_names4[, 1],

log2\_sowndiv = log2(as.numeric(split\_names4[, 2]))

)

output4 <- Stay\_Multiple(Jena\_plot\_biomass\_data, order.q = c(0.5, 1, 2))

To plot Figure 4 (a), use the function “LMM\_2\_to\_4” to fit the linear mixed model of biodiversity on gamma stability, with block as a random effect. Then, use function ‘fig\_2\_or\_4’ (provided in the source code) to plot Figure 4 (a).

output\_fig\_4a <- LMM\_2\_to\_4(output4, structure = structure4, metric\_name = "Gamma")

fig2\_or\_4(output\_fig\_4a, metric\_name = "Gamma")

**4b. R code for Figure 4b**

To plot Figure 4 (b), use the function “LMM\_2\_to\_4” to fit the linear mixed model of biodiversity on alpha stability, with block as a random effect. Then, use function ‘fig\_2\_or\_4’ (provided in the source code) to plot Figure 4 (b).

output\_fig\_4b <- LMM\_2\_to\_4(output4, structure = structure4, metric\_name = "Alpha")

fig2\_or\_4(output\_fig\_4b, metric\_name = "Alpha")

**4c. R code for Figure 4c**

To plot Figure 4 (c), use the function “LMM\_2\_to\_4” to fit the linear mixed model of biodiversity on spatial synchrony, with block as a random effect. Then, use function ‘fig\_2\_or\_4’ (provided in the source code) to plot Figure 4 (c).

output\_fig\_4c <- LMM\_2\_to\_4(output4, structure = structure4, metric\_name = "Synchrony")

fig2\_or\_4(output\_fig\_4c, metric\_name = "Synchrony")