# Package 'biosurvey'

September 15, 2021

**Type** Package

**Title** Tools for Biological Survey Planning

**Version** 0.1.1

**Maintainer** Claudia Nuñez-Penichet <claununez199o@gmail.com>

**Date** 2021-09-15

**Description** A collection of tools that allows users to plan systems of sampling
sites, increasing the efficiency of biodiversity monitoring by considering
the relationship between environmental and geographic conditions in a
region. The options for selecting sampling sites included here differ from
other implementations in that they consider the environmental and geographic
conditions of a region to suggest sampling sites that could increase the
efficiency of efforts dedicated to monitoring biodiversity. The methods
proposed here are new in the sense that they combine various criteria and
points previously made in related literature; some of the theoretical and
methodological bases considered are described in:
Arita et al. (2011) <doi:10.1111/j.1466-8238.2011.00662.x>,
Soberón and Cavner (2015) <doi:10.17161/bi.v10i0.4801>, and Soberón et al.
(2021).

**URL** https://github.com/claununez/biosurvey

**BugReports** https://github.com/claununez/biosurvey/issues

**Imports** diptest (>= 0.75), doParallel (>= 1.0), foreach (>= 1.5),
graphics, grDevices, ks (>= 1.11), maps (>= 3.3), methods,
parallel, picante (>= 1.8), raster (>= 3.0), rgdal (>= 1.4),
rgeos (>= 0.5), sp (>= 1.3), spatstat.geom (>= 1.65), stats,
utils, vegan (>= 2.5)

**Depends** R (>= 3.6.0)

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**Suggests** knitr, rmarkdown, testthat

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Claudia Nuñez-Penichet [aut, cre]
      (<https://orcid.org/0000-0001-7442-8593>),
      Marlon E. Cobos [aut] (<https://orcid.org/0000-0002-2611-1767>),
      Jorge Soberón [aut] (<https://orcid.org/0000-0003-2160-4148>),
      Tomer Gueta [aut] (<https://orcid.org/0000-0003-1557-8596>),
      Narayani Barve [aut] (<https://orcid.org/0000-0002-7893-8774>),
      Vijay Barve [aut] (<https://orcid.org/0000-0002-4852-2567>),
      Adolfo G. Navarro-Sigüenza [aut]
      (<https://orcid.org/0000-0003-2652-7719>),
      A. Townsend Peterson [aut] (<https://orcid.org/0000-0003-0243-2379>)

# R topics documented:

| assign_blocks | *Helper to assign block numbers to data according to variables and limits* |
|---|---|

## Description

Helper to assign block numbers to data according to variables and limits

**Usage**

```
assign_blocks(data, variable_1, variable_2, n_cols, n_rows = NULL,
              xlb, ylb = NULL, block_type = "equal_area")
```

**Arguments**

| | |
|---|---|
| data | matrix or data.frame that contains at least four columns: "Longitude" and "Latitude" to represent geographic position, and two other columns to represent the variables of the 2D environmental space. |
| variable_1 | (character or numeric) name or position of the first variable (x-axis) to be used to create blocks. |
| variable_2 | (character or numeric) name or position of the second variable (y-axis) to be used to create blocks (must be different from the first one). |
| n_cols | (numeric) number of columns of a grid used to create blocks and split the bi-dimensional space. |
| n_rows | (numeric) number of rows of a grid used to create blocks and split the bi-dimensional space. If NULL, the default, n_rows = n_cols. |
| xlb | (numeric) vector of values of extremes for all blocks considering variable_1. |
| ylb | (numeric) vector of values of extremes for all blocks considering variable_2. Needed when block_type = "equal_area". Default = NULL. |
| block_type | (character) type of blocks to be used for dividing the bi-dimensional space. Two options are available: "equal_area" and "equal_points". Default = "equal_area". Note that this option has important association regarding full representation of the extreme values of environmental variables across the study region. |

**Value**

Original element defined in data plus a new column named "Block" defining the block that correspond to each of the points represented in rows.

**Examples**

```
# Data
dat <- matrix(runif(800), ncol = 4)
xlims <- quantile(dat[, 3])
ylims <- quantile(dat[, 4])

# Assigning blocks
datb <- assign_blocks(dat, variable_1 = 3, variable_2 = 4, n_cols = 10,
                      xlb = xlims, ylb = ylims, block_type = "equal_area")
```

---

| base_PAM | *Constructor of S3 objects of class base_PAM* |

---

### Description

Constructor of S3 objects of class base_PAM

### Usage

```
new_base_PAM(PAM = new("SpatialPolygonsDataFrame"), PAM_indices = NULL)
```

### Arguments

PAM           SpatialPolygonsDataFrame object associated with information about presence and absence of species in a geographic grid.

PAM_indices   list of indices derived from a PAM. Default = NULL.

### Value

An object of class base_PAM.

---

| biosurvey | *biosurvey: Tools for Biological Survey Planning* |

---

### Description

biosurvey is a collection of tools that allow users to plan sampling sites. The methods presented increase the efficiency of biodiversity monitoring by considering the relationship between environmental and geographic conditions in a region.

### Details

Three main modules are included: 1) data preparation, 2) selection of sets of sites for biodiversity sampling, and 3) tools for testing efficiency of distinct sets of sampling sites. Data are prepared in ways that avoid the need for more data in posterior analyses, and allow concentrating in critical methodological decisions to select sampling sites. Various algorithms for selecting sampling sites are available, and options for considering pre-selected sites (known to be important for biodiversity monitoring) are included. Visualization is a critical component in this set of tools and most of the results obtained can be plotted to help to understand their implications. The options for selecting sampling sites included here differ from other implementations in that they consider the environmental and geographic structure of a region to suggest sampling sites that could increase the efficiency of efforts dedicated to monitoring biodiversity.

**Main functions in biosurvey**

block_sample, compare_SAC, EG_selection, DI_dendrogram, explore_data_EG, make_blocks,
PAM_indices, plot_blocks_EG, plot_DI, plot_PAM_geo, plot_PAM_CS, plot_SAC, plot_sites_EG,
prepare_base_PAM, prepare_master_matrix, prepare_PAM_CS random_selection, selected_sites_DI
selected_sites_SAC, subset_PAM, uniformE_selection, uniformG_selection

**Other functions (important helpers)**

assign_blocks, closest_to_centroid, distance_filter, files_2data, find_clusters, find_modes,
grid_from_region, match_rformat, PAM_from_table, point_sample, point_sample_cluster,
point_thinning, refill_PAM_indices, rlist_2data, selected_sites_PAM, spdf_2data, stack_2data,
unimodal_test, wgs84_2aed_laea

**Data included**

b_pam, distance_filter, dist_list, files_2data, m_matrix, m_matrix_pre, m_selection,
mx, preselected, sp_data, species_data, sp_layers, sp_occurrences, variables,

---

block_sample                  *Selection of blocks in environmental space*

---

**Description**

Select a user-defined number of blocks in environmental space to be used in further analysis to
define sampling sites for a survey system.

**Usage**

```
block_sample(master, expected_blocks, selection_type = "uniform",
            replicates = 10, set_seed = 1)
```

**Arguments**

master          master_matrix object derived from the function prepare_master_matrix or a
                master_selection object derived from functions random_selection, uniformG_selection,
                or uniformE_selection.

expected_blocks
                (numeric) number of blocks to be selected.

selection_type  (character) type of selection. Two options are available: "uniform" and "ran-
                dom". Default = "uniform".

replicates      (numeric) number of thinning replicates performed to select blocks uniformly.
                Default = 10.

set_seed        (numeric) integer value to specify a initial seed. Default = 1.

## Details

When blocks in `master` are defined using the option "equal_points" (see [make_blocks](#)), "uniform" `selection_type` could result in blocks with high density per area being overlooked.

## Value

An S3 object of class master_matrix or master_selection, containing the same elements found in the input object, with an additional column in the master_matrix data.frame containing a binary code for selected (1) and non-selected (0) blocks.

## Examples

```
# Data
data("m_matrix", package = "biosurvey")

# Making blocks for analysis
m_blocks <- make_blocks(m_matrix, variable_1 = "PC1",
                        variable_2 = "PC2", n_cols = 10, n_rows = 10,
                        block_type = "equal_area")

# Checking column names and values in variables to define initial distance
colnames(m_blocks$data_matrix)
summary(m_blocks$data_matrix[, c("PC1", "PC2")])

# Selecting blocks uniformly in E space
block_sel <- block_sample(m_blocks, expected_blocks = 10,
                          selection_type = "uniform")

head(block_sel$data_matrix)
```

---

b_pam                           *Example of object obtained from using the function base_PAM*

---

## Description

An S3 object of class base_PAM. See functions [prepare_base_PAM](#).

## Usage

```
b_pam
```

## Format

A list of 2 elements:

**PAM** SpatialPolygonsDataFrame with 306 features

**PAM_indices** a list of 11 elements

**Examples**

```
data("b_pam", package = "biosurvey")

print(b_pam)
```

---

closest_to_centroid *Detection of the closest points to the centroid of a cloud of points*

---

**Description**

Detection of the closest points to the centroid of a cloud of points

**Usage**

```
closest_to_centroid(data, x_column, y_column, space, n = 1, id_column = NULL)
```

**Arguments**

| | |
|---|---|
| data | matrix or data.frame that contains at least two columns. |
| x_column | (character) the name of the x-axis. |
| y_column | (character) the name of the y-axis. |
| space | (character) space in which the thinning will be performed. There are two options available: "G", if it will be in geographic space, and "E", if it will be in environmental space. |
| n | (numeric) number of points that are close to the centroid to be detected. Default = 1. |
| id_column | (character or numeric) name or numeric index of the column in data containing identifiers of one or distinct sets of points. If, NULL, the default, only one set is assumed. |

**Value**

A data.frame containing n rows corresponding to the point or points that are the closest to the centroid of all other points of reference.

**Examples**

```
# Data
data("m_matrix", package = "biosurvey")
data1 <- m_matrix$data_matrix

# Finding the closest point to the centroid
centroid <- closest_to_centroid(data1, x_column = "Longitude",
                                y_column = "Latitude", space = "G",
                                n = 1, id_column = NULL)
```

---

compare_SAC                    *Comparative plots of species accumulation curves*

---

### Description

creates comparative plots of two species accumulation curves from information contained in lists
obtained with the function `selected_sites_SAC`.

### Usage

```
compare_SAC(SAC_selected_sites, element_1, element_2, col_mean1 = "blue",
            col_CI1 = "lightblue", col_mean2 = "gray15",
            col_CI2 = "gray65", lty1 = 1, lty2 = 2,
            alpha_mean = 0.9, alpha_CI = 0.3,
            xlab = "Number of sites", ylab = "Species",
            line_for_multiple = TRUE, add_legend = TRUE, ...)
```

### Arguments

SAC_selected_sites

nested list of "specaccum" objects obtained with function `selected_sites_SAC`.

element_1         (numeric or character) index of position or character indicator of the first ele-
                  ment (type of selection) in `SAC_selected_sites` to be plotted. Character op-
                  tions are: "random", "E", "G", "EG".

element_2         (numeric or character) index of position or character indicator of the second
                  element (type of selection) in `SAC_selected_sites` to be plotted. Character
                  options are: "random", "E", "G", "EG".

col_mean1         (character) color for mean value of curve in `element_1`; default = "blue".

col_CI1           (character) color for confidence interval region for the curve in `element_1`; de-
                  fault = "lightblue".

col_mean2         (character) color for mean value of curve in `element_2`; default = "gray15".

col_CI2           (character) color for confidence interval region for the curve in `element_2`; de-
                  fault = "gray65".

lty1              type of line for `element_1`. See lty in `par`.

lty2              type of line for `element_2`. See lty in `par`.

alpha_mean        (numeric) alpha level for line representing the mean, values from 0 to 1; default
                  = 0.9. Values close to 0 increase transparency.

alpha_CI          (numeric) alpha level for the region representing the confidence interval; default
                  = 0.3.

xlab              (character) label for x-axis of plot; default = "Number of sites".

ylab              (character) label for y-axis of plot; default = "Species".

line_for_multiple

(logical) whether to plot SACs only as lines when multiple objects are in one or
more of the internal lists in `SAC_selected_sites`. Default = TRUE.

add_legend          (logical) whether to add default legend to plot; default = TRUE.

...                 other arguments to be passed to plot method for objects of class "specaccum".

## Value

A comparative plot of two species "specaccum" objects done based on what is defined in element_1
and element_2.

## Examples

```
# Data
data("b_pam", package = "biosurvey")
data("m_selection", package = "biosurvey")

# Subsetting base PAM according to selections
sub_pam_all <- subset_PAM(b_pam, m_selection, selection_type = "all")

SACs <- selected_sites_SAC(PAM_subset = sub_pam_all, selection_type = "all")

compare_SAC(SAC_selected_sites = SACs, element_1 = 1, element_2 = 2)
```

---

distance_filter                 *Helper to filter sets of sites by median distance among all points*

---

## Description

Helper to filter sets of sites by median distance among all points

## Usage

```
distance_filter(site_list, median_distance_filter = "max")
```

## Arguments

site_list           list of selected sites provided as data.frames. Columns "Longitude" and "Lati-
                    tude" are needed for distance calculation.

median_distance_filter

                    (character) optional argument to define a median distance-based filter based
                    on which sets of sampling sites will be selected. Default = "max". Options
                    are:"max" and "min".

## Value

A list of sets of selected sites according to median_distance_filter.

## Examples

```
# Data
data("m_selection", package = "biosurvey")

slist <- m_selection$selected_sites_random

# Distance filter
max_sites <- distance_filter(slist, median_distance_filter = "max")
```

---

dist_list                    *A list of vectors of distances*

---

### Description

A list of six vectors of point distances.

### Usage

```
dist_list
```

### Format

A list.

**vector** numeric, values of distances (six elements)

### Examples

```
data("dist_list", package = "biosurvey")
```

---

dis_loop                    *Helper to calculate dissimilarities in loop*

---

### Description

Helper to calculate dissimilarities in loop

### Usage

```
dis_loop(site_spp_list, icol, fcol, method = "jaccard", verbose = TRUE, ...)
```

## Arguments

| | |
|---|---|
| `site_spp_list` | list of presence absence matrices for a set of sites or cells of a grid. |
| `icol` | number of column where the species list (other columns) starts. |
| `fcol` | number of column where the species list (other columns) ends. |
| `method` | (character) dissimilarity index to be passed to function `vegdist`. Default = "jaccard". See details. |
| `verbose` | (logical) whether or not to print messages about the process. Default = TRUE. |
| `...` | other arguments to be passed to function `vegdist`. |

## Value

A list of results from `vegdist`.

---

| DI_dendrogram | *Plot dissimilarities withing and among sets of selected sites as a dendrogram* |
|---|---|

---

## Description

Plot dissimilarities withing and among sets of selected sites as a dendrogram

## Usage

```
DI_dendrogram(DI_selected_sites, selection_type = "selections",
              selection_number = 1, labels = NULL, xlab = "",
              ylab = "Distance", main = "Cluster dendrogram",
              sub = "", ...)
```

## Arguments

| | |
|---|---|
| `DI_selected_sites` | |
| | list of results obtained with function `selected_sites_DI`. |
| `selection_type` | type of selection to be considered when creating DI matrix plot. Options are: "selections", "random", "E", "G", and "EG". The default, "selections", plots a comparison among all selection types. |
| `selection_number` | |
| | (numeric) number of selection to be plotted. Default = 1. Ignored if `selection_type` = "selections". |
| `labels` | (character) vector of labels for the tips of the tree. The default, NULL, uses names of sets of selected sites. If labels = FALSE no tip labels are plotted. |
| `xlab` | (character) label for x-axis of plot. Default = "". |
| `ylab` | (character) label for y-axis of plot. Default = "Distance". |
| `main` | (character) title for the plot. Default = "Cluster dendrogram". |
| `sub` | (character) subtitle for the plot. Plotted below the label of the x-axis. |
| `...` | other arguments to be passed to plot method for objects of class `"hclust"`. See more details in `hclust`. |

## Value

A dendrogram plot of a `"hclust"` object.

## Examples

```
# Data
data("b_pam", package = "biosurvey")
data("m_selection", package = "biosurvey")

# Subsetting base PAM according to selections
sub_pam_all <- subset_PAM(b_pam, m_selection, selection_type = "all")

# Calculating dissimilarities
DI_sel <- selected_sites_DI(sub_pam_all)

# Plot
DI_dendrogram(DI_sel)
```

---

| | |
|---|---|
| EG_selection | *Selection of survey sites maximizing uniformity in environmental space considering geographic structure* |

---

## Description

Selection of sites to be sampled in a survey, with the goal of maximizing uniformity of points in the environment, but considering geographic patterns of data. Sets of points that are environmentally similar and have a disjoint pattern in geography, are selected twice (two survey sites are placed so they consider the biggest geographic clusters).

## Usage

```
EG_selection(master, n_blocks, guess_distances = TRUE, initial_distance = NULL,
             increase = NULL, max_n_samplings = 1, replicates = 10,
             use_preselected_sites = TRUE, select_point = "E_centroid",
             cluster_method = "hierarchical", median_distance_filter = NULL,
             sample_for_distance = 250, set_seed = 1,
             verbose = TRUE, force = FALSE)
```

## Arguments

| | |
|---|---|
| master | master_matrix object derived from the function [prepare_master_matrix](#) or master_selection object derived from functions [random_selection](#), [uniformG_selection](#), or [uniformE_selection](#). |
| n_blocks | (numeric) number of blocks to be selected to be used as the base for further explorations. Default = NULL. |

guess_distances

> (logical) whether or not to use internal algorithm to automatically select initial_distance and increase. Default = TRUE. If FALSE, initial_distance and increase must be defined.

initial_distance

> (numeric) Euclidean distance to be used for a first process of thinning and detection of remaining blocks. See details in point_thinning. Default = NULL.

increase (numeric) initial value to be added to or subtracted from initial_distance until reaching the number of expected_points. Default = NULL.

max_n_samplings

> (numeric) maximum number of samples to be chosen after performing all thinning replicates. Default = 1.

replicates (numeric) number of thinning replicates performed to select blocks uniformly. Default = 10.

use_preselected_sites

> (logical) whether to use sites that have been defined as part of the selected sites previous any selection. Object in master must contain the site(s) preselected in and element of name "preselected_sites" for this argument to be effective. Default = TRUE. See details for more information on the approach used.

select_point (character) how or which point will be selected for each block or cluster. Three options are available: "random", "E_centroid", and "G_centroid". E_ or G_ centroid indicate that the point(s) closets to the respective centroid will be selected. Default = "E_centroid".

cluster_method (character) name of the method to be used for detecting geographic clusters of points inside each block. Options are "hierarchical" and "k-means"; default = "hierarchical". See details in find_clusters.

median_distance_filter

> (character) optional argument to define a median distance-based filter based on which sets of sampling sites will be selected. The default, NULL, does not apply such a filter. Options are: "max" and "min". See details.

sample_for_distance

> (numeric) sample to be considered when measuring the geographic distances among points in blocks created in environmental space. The distances measured are then used to test whether points are distributed uniformly or not in the geography. Default = 250.

set_seed (numeric) integer value to specify a initial seed. Default = 1.

verbose (logical) whether or not to print messages about the process. Default = TRUE.

force (logical) whether to replace existing set of sites selected with this method in master.

### Details

Two important steps are needed before using this function: 1) exploring data in environmental and geographic spaces, and 2) performing a regionalization of the environmental space. Exploring the data can be done using the function explore_data_EG. This step is optional but strongly recommended, as important decisions that need to be taken depend on the of the data in the two spaces. A

regionalization of the environmental space configuration of the region of interest helps in defining important parts of your region that should be considered to select sites. This can be done using the function make_blocks. Later, the regions created in environmental space will be used for selecting one or more sampling sites per block depending on the geographic pattern of such environmental combinations.

The process of survey-site selection with this function is the most complex among all functions in this package. The complexity derives from the aim of the function, which is to select sites that sample appropriately environmental combinations in the region of interest (environmental space), but considering the geographic patterns of such environmental regions (geographic space).

In this approach, the first step is to select candidate blocks (from the ones obtained with make_blocks) that are uniformly distributed in environmental space. The geographic configuration of points in such blocks is explored to detect whether they are clustered (i.e., similar environmental conditions are present in distant places in the region of interest). For blocks with points that present one cluster in geography, only one survey site is selected, and for those with multiple clusters in geographic space, two survey sites are selected considering the two largest clusters.

If use_preselected_sites is TRUE and such sites are included as an element in the object in master, the approach for selecting sites in environmental space considering geographic patterns is a little different. User-preselected sites will always be part of the sites selected. Other points are selected based on an algorithm that searches for sites that are uniformly distributed in environmental space but at a distance from preselected sites that helps in maintaining uniformity among environmental blocks selected. Note that preselected sites will not be processed, therefore, uniformity of blocks representing such points cannot be warrantied.

As multiple sets could result from selection, the argument of the function median_distance_filter could be used to select the set of sites with the maximum ("max") or minimum ("min") median distance among selected sites. Option "max" will increase the geographic distance among sampling sites, which could be desirable if the goal is to cover the region of interest more broadly. The other option, "min", could be used in cases when the goal is to reduce resources and time needed to sample such sites.

### Value

A master_selection object (S3) with a special element called selected_sites_EG containing one or more sets of selected sites depending on max_n_samplings and median_distance_filter.

### See Also

uniformG_selection, uniformE_selection, random_selection, make_blocks, plot_sites_EG

### Examples

```
# Data
data("m_matrix", package = "biosurvey")

# Making blocks for analysis
m_blocks <- make_blocks(m_matrix, variable_1 = "PC1", variable_2 = "PC2",
                        n_cols = 10, n_rows = 10, block_type = "equal_area")

# Checking column names
```

```
colnames(m_blocks$data_matrix)

# Selecting sites uniformly in E and G spaces
EG_sel <- EG_selection(master = m_blocks, n_blocks = 10,
                       initial_distance = 1.5, increase = 0.1,
                       replicates = 1, max_n_samplings = 1,
                       select_point = "E_centroid",
                       cluster_method = "hierarchical",
                       sample_for_distance = 100)

head(EG_sel$selected_sites_EG[[1]])
dim(EG_sel$selected_sites_EG[[1]])
```

---

explore_data_EG                    *Plots to explore environmental factors in environmental and geo-*
                                   *graphic space*

---

### Description

Creates a four-panel plot with information of two environmental predictors (at a time) in the region
of interest (or region reduced with mask, if used). The two top panels contain the information in
geographic space (one predictor per panel). The two panels at the bottom contain information in a
2D environmental space for the two variables.

### Usage

```
explore_data_EG(master, variable_1, variable_2, region_border = TRUE,
                mask_border = FALSE, col_variable1 = NULL,
                col_variable2 = NULL, col_points = NULL, col_density = NULL)
```

### Arguments

| | |
|---|---|
| master | master_matrix object derived from function [prepare_master_matrix](#) or mas- ter_selection object derived from functions [uniformG_selection](#), [uniformE_selection](#) or [EG_selection](#). |
| variable_1 | (character or numeric) name or position of the first variable (x-axis) to be ex- plored. |
| variable_2 | (character or numeric) name or position of the second variable (y-axis) to be explored (must be different from the first one). |
| region_border | (logical) whether to add region border to the plot. Default = TRUE. |
| mask_border | (logical) whether to add mask border to the plot. Ignored if mask is not present in master_selection. Default = FALSE. |
| col_variable1 | a color palette for variable_1 defined using functions like [heat.colors](#), or one generated using functions like [colorRampPalette](#). The default, NULL, uses a colorblind friendly palette similar to viridis. |

col_variable2      a color palette for `variable_2` defined using functions like [heat.colors](), or one generated using functions like [colorRampPalette](). The default, NULL, uses a colorblind friendly palette similar to viridis.

col_points      color for points in environmental space. The default, NULL, uses the 25th color of the default palette for `col_variable1` with an alpha of 0.6.

col_density      color palette to represent representation density of points in environmental space. This palette can be defined using functions like [heat.colors](), or one generated using functions like [colorRampPalette](). The default, NULL, uses a colorblind friendly palette similar to magma, and changes the first color in the palette to NA.

## Value

A multi-panel plot showing two of the environmental predictors in the region of interest in both spaces, geographic and environmental.

## Examples

```
# Data
data("m_matrix", package = "biosurvey")

colnames(m_matrix$data_matrix)

# Plot

explore_data_EG(m_matrix, variable_1 = "Mean_temperature",
                variable_2 = "Annual_precipitation")
```

---

files_2data                  *Creates a data.frame of species' references from files in a directory*

---

## Description

Creates a data.frame of species' references that contains longitude, latitude, and species name, from a character.

## Usage

```
files_2data(path, format, spdf_grid = NULL, parallel = FALSE, n_cores = NULL)
```

## Arguments

path           (character) full path name of directory containing raster, shapefiles, geopackage, or GeoJSON files representing species geographic ranges. Each file must be named as the species that it represents. All files must be in the same format. If files are raster, values in each layer must be 1 (presence) and 0 (absence).

| format | (character) the format files found in `path`. Current available formats are: "shp", "gpkg", "geojson", "GTiff", and "ascii". |
|---|---|
| spdf_grid | geographic grid for the region of interest (output of function [grid_from_region](#)). Used when format equals "shp" or "gpkg". Default = NULL. |
| parallel | (logical) whether to perform analyses in parallel. Default = FALSE. |
| n_cores | (numeric) number of cores to be used when `parallel = TRUE`. The default, NULL, uses available cores - 1. |

## Value

If files are in raster format, a data.frame of species geographic records derived from values of presence in each layer.

If files are not in raster format, a data.frame of species' found in distinct positions (defined with identifiers); includes two columns: "ID" and "Species".

## Examples

```
# Data for examples
data("mx", package = "biosurvey")
data("species_data", package = "biosurvey")

# Saving species data in a temporal directory
tdir <- file.path(tempdir(), "testbio")
dir.create(tdir)

namessp <- paste0("species_", 1:length(species_data))


for (i in 1:length(species_data)) {
  rgdal::writeOGR(species_data[i, ], dsn = tdir, layer = namessp[i],
                  driver = "ESRI Shapefile")
}

# Preparing grid for analysis
grid_reg <- grid_from_region(region = mx, cell_size = 100)

# Running analysis with data from directory
sp_data <- files_2data(path = tdir, format = "shp", spdf_grid = grid_reg)
```

---

| find_clusters | *Detection of clusters in 2D spaces* |
|---|---|

---

## Description

Finds clusters of data in two dimensions based on distinct methods.

## Usage

```
find_clusters(data, x_column, y_column, space,
              cluster_method = "hierarchical", n_k_means = NULL,
              split_distance = NULL)
```

## Arguments

| | |
|---|---|
| data | matrix or data.frame that contains at least two columns. |
| x_column | (character) the name of the x-axis. |
| y_column | (character) the name of the y-axis. |
| space | (character) space in which the thinning will be performed. There are two options available: "G", if it will be in the geographic space, and "E", if it will be in the environmental space. |
| cluster_method | (character) name of the method to be used for detecting clusters. Options are "hierarchical" and "k-means"; default = "hierarchical". |
| n_k_means | (numeric) number of clusters to be identified when using the "k-means" in cluster_method. |
| split_distance | (numeric) distance in meters (if space = "G") or Euclidean distance (if space = "E") to identify clusters if cluster_method = "hierarchical". |

## Details

Clustering methods make distinct assumptions and one of them may perform better than the other depending on the pattern of the data.

The k-means method tends to perform better when data are grouped spatially (spherically) and clusters are of a similar size. The hierarchical clustering algorithm usually takes more time than the k-means method. Both methods make assumptions and may work well on some data sets but fail on others.

## Value

A data frame containing data and an additional column defining clusters.

## Examples

```
# Data
data("m_matrix", package = "biosurvey")

# Cluster detection
clusters <-  find_clusters(m_matrix$data_matrix, x_column = "PC1",
                          y_column = "PC2", space = "E",
                          cluster_method = "hierarchical", n_k_means = NULL,
                          split_distance = 4)
head(clusters)
```

---

find_modes                   *Find modes in a multimodal distribution*

---

### Description

Find modes in a multimodal distribution of values based on the density of such values.

### Usage

```
find_modes(density)
```

### Arguments

density          object of class density obtained using the function [density](#).

### Value

A data.frame containing the values corresponding to the modes and the density for those particular
values.

### Examples

```
# Data
data("dist_list", package = "biosurvey")

dens <- density(dist_list$`12`)

# Finding modes
modes <- find_modes(density = dens)
modes
```

---

grid_from_region             *Creates grid for a given geographic region*

---

### Description

Divides the region of interest in a grid of a specific cell size.

### Usage

```
grid_from_region(region, cell_size, complete_cover = TRUE)
```

**Arguments**

| | |
|---|---|
| region | SpatialPolygonsDataFrame of the region of interest. Object must be unprojected, World Geodetic System (WGS84). |
| cell_size | (numeric) resolution for grid (single number or vector of two numbers) in kilometers (km). |
| complete_cover | (logical) whether or not to include cells of grid partially overlapped with region. Default = TRUE. |

**Value**

Gridded SpatialPolygonsDataFrame for the region of interest. Each grid cell is related to a specific ID and longitude and latitude coordinates.

**Examples**

```
# Data
data("mx", package = "biosurvey")

# Create grid from polygon
grid_reg <- grid_from_region(region = mx, cell_size = 100)

sp::plot(grid_reg)
grid_reg
```

---

| legend_bar | *Helper to add a bar image legend to plots* |
|---|---|

---

**Description**

Helper to add a bar image legend to plots

**Usage**

```
legend_bar(position, col, width_prop = 0.03, heigh_prop = 0.18,
           title = "Legend", labels = c("Low", "High"), digits = 0,
           labels_offset = 0.2, horizontal = FALSE, alpha = 1, border = NULL,
           cex = NULL, inset = 0.05, insetx = NULL, insety = NULL)
```

**Arguments**

| | |
|---|---|
| position | (numeric or character) position of the bottom left corner of the legend. If numeric, x and y coordinates. If character, options are: "bottomright", "bottom", "bottomleft", "left", "topleft", "top", "topright", "right", or "center". |
| col | color palette. A vector of contiguous colors. It can be generated using functions like [purplow](e.g., darkros(255)). |
| width_prop | width of bar legend represented as a proportion of the entire plotting width. Default = 0.03. |

| | |
|---|---|
| heigh_prop | heigh of bar legend represented as a proportion of the entire plotting heigh. Default = 0.18. |
| title | legend title. Default = "Legend". |
| labels | (numeric or character) labels for the legend. Default = c("Low", "High"). |
| digits | (numeric) number of decimal places to round numeric labels. Default = 0. |
| labels_offset | offset of labels from bar. Default = 0.2. |
| horizontal | (logical) should the legend be horizontal. Default = FALSE. |
| alpha | (numeric) alpha level 0-1. Default = 1. |
| border | color for the border of the legend bar. The default, NULL, does not plot a border. |
| cex | character expansion factor relative to current. Default = NULL. |
| inset | inset distances from plot margins relative to plot region. Default = 0.05. |
| insetx | inset from x margins. The default, NULL, uses inset. |
| insety | inset from y margins. The default, NULL, uses inset. |

## Value

A bar legend for a plot.

---

| make_blocks | *Creates a block-like regionalization of environmental space* |
|---|---|

---

## Description

Divides a two-dimensional cloud of points in blocks according to a user-defined number of rows and columns. This is applied to the element master_matrix and, if not NULL, to preselected_sites.

## Usage

```
make_blocks(master_matrix, variable_1, variable_2, n_cols, n_rows = NULL,
            block_type = "equal_area")
```

## Arguments

| | |
|---|---|
| master_matrix | object derived from function [prepare_master_matrix](#). |
| variable_1 | (character or numeric) name or position of the first variable (x-axis) to be used to create blocks. |
| variable_2 | (character or numeric) name or position of the second variable (y-axis) to be used to create blocks (must be different from the first one). |
| n_cols | (numeric) number of columns of a grid used to creates blocks and split the bi-dimensional space. |
| n_rows | (numeric) number of rows of a grid used to creates blocks and split the bi-dimensional space. If NULL, the default, n_rows = n_cols. |
| block_type | (character) type of blocks to be use for dividing the bi-dimensional space. Two options are available: "equal_area" and "equal_points". Default = "equal_area". |

**Details**

For block_type, option "equal_area" generates blocks of the same size. The other option ("equal_points"), generates blocks containing the same number of points, which generally results in blocks of different sizes.

**Value**

An S3 object of class master_matrix, containing the same elements found in a master_matrix object, with an additional column on the master_matrix data.frame containing block identifiers. If the element preselected_sites is not NULL in master_matrix, blocks are also assigned to this sites.

**Examples**

```
# Data
data("m_matrix", package = "biosurvey")

# Creating blocks
m_blocks <- make_blocks(m_matrix, variable_1 = "PC1",
                        variable_2 = "PC2", n_cols = 10, n_rows = 10,
                        block_type = "equal_area")
unique(m_blocks$data_matrix$Block)
```

---

| master_matrix | *Constructor of S3 objects of class master_matrix* |
|---|---|

---

**Description**

Constructor of S3 objects of class master_matrix

**Usage**

```
new_master_matrix(data_matrix, preselected_sites = NULL, region,
                  mask = NULL, raster_base, PCA_results = NULL)
```

**Arguments**

data_matrix      date.frame with information about geographic location of raster cells, initial environmental data, and if available, the first two principal components derived from an analysis done with environmental values.

preselected_sites

     data.frame containing sites that must be included in posterior selections of sites for the survey system. Columns must be: "Sites", "Longitude", "Latitude", in that order. Default = NULL.

region      SpatialPolygons* object representing the region of interest.

mask      SpatialPolygons* object used. Default = NULL.

raster_base      SpatialPolygonsDataFrame representing the grid of the raster layers used, which will be used for plotting purposes.

PCA_results        results of principal component analysis performed with values from raster layers
                   used. Default = NULL.

## Value

An S3 object of class `master_matrix`.

---

master_selection                *Constructor of S3 objects of class master_selection*

---

## Description

Constructor of S3 objects of class master_selection

## Usage

```
new_master_selection(data_matrix, preselected_sites = NULL, region,
                     mask = NULL, raster_base, PCA_results = NULL,
                     selected_sites_random = NULL, selected_sites_G = NULL,
                     selected_sites_E = NULL, selected_sites_EG = NULL)
```

## Arguments

data_matrix        date.frame with information about geographic location of raster cells, initial en-
                   vironmental data, and if available, the first two principal components derived
                   from an analysis done with environmental values.

preselected_sites
                   data.frame containing sites that must be included in posterior selections of sites
                   for the survey system. Columns must be: "Sites", "Longitude", "Latitude", in
                   that order. Default = NULL.

region             SpatialPolygons* object representing the region of interest.

mask               SpatialPolygons* object used. Default = NULL.

raster_base        a SpatialPolygonsDataFrame representing the grid of the raster layers used,
                   which will be used for plotting purposes.

PCA_results        results of principal component analysis performed with values from raster layers
                   used. Default = NULL.

selected_sites_random
                   data.frame with the sites selected randomly. Default = NULL.

selected_sites_G
                   data.frame with the sites selected based on geographic distances. Default =
                   NULL.

selected_sites_E
                   data.frame with the sites selected based on environmental distances. Default =
                   NULL.

selected_sites_EG
                   data.frame with the sites selected based on environmental and geographic con-
                   siderations. Default = NULL.

## Value

An S3 object of class `master_selection`.

---

| match_rformat | *Helper function to find raster extension* |
|---|---|

---

### Description

Helper function to find raster extension

### Usage

```
match_rformat(format)
```

### Arguments

format          (character) any of the format types allowed for raster objects. See `writeFormats`
                (e.g., "GTiff").

### Value

Raster extension according to format type.

### Examples

```
match_rformat("GTiff")
```

---

| mx | *Example of spatial polygon for a region of interest* |
|---|---|

---

### Description

An object of class SpatialPolygonsDataFrame.

### Usage

```
mx
```

### Format

SpatialPolygonsDataFrame:

**data** data.frame with 1 row and 11 columns

**polygons** SpatialPolygons

**proj4string** object of class CRS

## Examples

```
data("mx", package = "biosurvey")

mx
```

---

m_matrix *Example of a master_matrix object with no preselected sites*

---

### Description

An S3 object of class master_matrix. See function `prepare_master_matrix`.

### Usage

```
m_matrix
```

### Format

A list of 6 elements:

**data_matrix** data.frame with 6276 rows and 10 columns

**preselected_sites** NULL

**region** object of class SpatialPolygons*

**mask** NULL

**raster_base** object of class RasterLayer

**PCA_results** list of length 5

### Examples

```
data("m_matrix", package = "biosurvey")

print(m_matrix)
```

---

m_matrix_pre *Example of a master_matrix object containing preselected sites*

---

### Description

A S3 object of class master_matrix. See function `prepare_master_matrix`.

### Usage

```
m_matrix_pre
```

## Format

A list of 6 elements:

**data_matrix**  data.frame with 6276 rows and 10 columns

**preselected_sites**  data.frame with 5 rows and 11 columns

**region**  object of class SpatialPolygons*

**mask**  NULL

**raster_base**  object of class RasterLayer

**PCA_results**  list of length 5

## Examples

```
data("m_matrix_pre", package = "biosurvey")

print(m_matrix_pre)
```

---

| m_selection | *Example of a master_selection object from using functions for selecting sites* |
|---|---|

---

## Description

An S3 object of class master_selection. See functions `uniformE_selection`, `uniformG_selection`, `random_selection`, or `EG_selection`.

## Usage

```
m_selection
```

## Format

A list of 10 elements:

**data_matrix**  data.frame with 6276 rows and 10 columns

**preselected_sites**  NULL

**region**  object of class SpatialPolygons*

**mask**  NULL

**raster_base**  object of class RasterLayer

**PCA_results**  list of length 5

**selected_sites_random**  list with one data.frame

**selected_sites_G**  list with one data.frame

**selected_sites_E**  list with one data.frame

**selected_sites_EG**  NULL

## Examples

```
data("m_selection", package = "biosurvey")

print(m_selection)
```

---

PAM_CS                          *Constructor of S3 objects of class PAM_CS*

---

## Description

Constructor of S3 objects of class PAM_CS

## Usage

```
new_PAM_CS(Species = NA, Sites_cells = NA, Beta_W = NA, Spearman_cor = NA,
           Theoretical_boundaries = list(x = NA, y = NA),
           Richness_normalized = NA, Dispersion_field_normalized = NA,
           S_significance_id = NA, Randomized_DF = matrix())
```

## Arguments

Species         (numeric) species name. Default = NA.

Sites_cells      (numeric) number of sites or cells. Default = NA.

Beta_W         (numeric) value of Whittaker's Beta. Default = NA.

Spearman_cor    (numeric) value of Spearman's correlation. Default = NA.

Theoretical_boundaries
               list of theoretical boundaries for the values. Default = NA.

Richness_normalized
               (numeric) values of normalized richness. Default = NA.

Dispersion_field_normalized
               (numeric) values of normalized dispersion field. Default = NA.

S_significance_id
               (numeric) values indicating statistical significance of the normalized dispersion field. Default = NA.

Randomized_DF   matrix of values resulted from randomizing matrices. Default = NA.

## Value

An object of class PAM_CS.

---

PAM_from_table                  *Creates presence-absence matrix from a data.frame*

---

### Description

Creates a presence-absence matrix (PAM) from a data.frame that contains species names and identifiers of positions where species are found.

### Usage

```
PAM_from_table(data, ID_column, species_column)
```

### Arguments

| | |
|---|---|
| data | data.frame of species found in distinct positions (defined by identifiers). Must include at least two columns: "ID" and "Species". |
| ID_column | (character) name of the column containing identifiers. |
| species_column | (character) name of the column containing species names. |

### Value

Species' presence (1) and absence (0) matrix for a set of positions defined by identifiers.

### Examples

```
# Data
data("sp_data", package = "biosurvey")

# PAM
pam <- PAM_from_table(data = sp_data, ID_column = "ID",
                      species_column = "Species")
pam[1:10, c(1, 21:25)]
```

---

PAM_indices                  *Biodiversity indices derived from PAM*

---

### Description

Calculates a set of biodiversity indices using values contained in a presence-absence matrix.

### Usage

```
PAM_indices(PAM, indices = "all", exclude_column = NULL)
```

## Arguments

| | |
|---|---|
| PAM | matrix, data.frame, or base_PAM object containing information on species presence and absence for a set of sites. Sites are organized in the rows and species in the columns. See details. |
| indices | (character) code for indices to be calculated. Basic indices are calculated all the time; other indices need to be specified. Options are: "all", "basic, "AB", "BW", "BL", "SCSC", "SCSR", "DF", "CC", "WRN", "SRC", "CMSC", "CMSR", "MCC", and "MRC". Default = "all". See details. |
| exclude_column | (optional) name or numeric index of columns to be excluded. Default = NULL. |

## Details

Descriptions of the codes of all indices to be calculated are presented in the table below. If `indices` = "basic", only basic indices are calculated. However, basic indices are calculated in all cases not matter the code(s) defined in `indices`. Some indices require previous calculations of other indices, in such cases, all indices required are added to the final list. For further details on the way calculations are performed and the meaning of the indices see Soberón and Cavner (2015) doi: 10.17161/bi.v10i0.4801.

| Code | Index | Calculation |
|---|---|---|
| RI | Richness | Basic |
| RA | Range | Basic |
| RIN | Richness normalized | Basic |
| RAN | Range normalized | Basic |
| AB | Additive Beta | Needs to be defined |
| BW | Beta Whittaker | Needs to be defined |
| BL | Beta Legendre | Needs to be defined and DF |
| SCSC | Schluter covariance sites-composition | Needs to be defined and CMSC |
| SCSR | Schluter covariance species-ranges | Needs to be defined and CMSR |
| DF | Dispersion field | Needs to be defined |
| SCC | Shared community composition | Needs to be defined |
| WRN | Wright-Reeves nestedness | Needs to be defined, BW, and DF |
| SRC | Stone-Roberts C-score | Needs to be defined and DF |
| CMSC | Covariance matrix sites-composition | Needs to be defined, DF, and BW |
| CMSR | Covariance matrix species-ranges | Needs to be defined, SCC, and BW |
| MCC | Mean composition covariance | Calculated with CMSC |
| MRC | Mean range covariance | Calculated with CMSR |

## Value

If `PAM` is a matrix or data.frame, the result is a list with the results described below (depending on `indices`). If `PAM` is a base_PAM object, a base_PAM object will be returned and the list described above will be appended to the element PAM_indices in such an element.

## See Also

[prepare_base_PAM](prepare_base_PAM)

## Examples

```
# Data
data("sp_data", package = "biosurvey")

# PAM
pam <- PAM_from_table(data = sp_data, ID_column = "ID",
                      species_column = "Species")

pam_ind <- PAM_indices(pam, exclude_column = 1)
pam_ind[1:3]
```

---

PAM_subset                    *Constructor of S3 objects of class PAM_subset*

---

## Description

Constructor of S3 objects of class PAM_subset

## Usage

```
new_PAM_subset(PAM = new("SpatialPolygonsDataFrame"), PAM_indices = NULL,
               PAM_selected_sites_random = NULL, PAM_selected_sites_G = NULL,
               PAM_selected_sites_E = NULL, PAM_selected_sites_EG = NULL)
```

## Arguments

PAM                  SpatialPolygonsDataFrame object associated to information about species presence and absence in a geographic grid.

PAM_indices          list of indices derived from a PAM. Default = NULL.

PAM_selected_sites_random

                     subset of PAM for sites derived from random selection. Default = NULL.

PAM_selected_sites_G

                     subset of PAM for sites derived from selection considering geographic distances. Default = NULL.

PAM_selected_sites_E

                     subset of PAM for sites derived from selection considering environmental distances. Default = NULL.

PAM_selected_sites_EG

                     subset of PAM for sites derived from selection considering environment and geography. Default = NULL.

## Value

An object of class PAM_subset.

---

plot_blocks_EG          *Representation of environmental blocks in geography and environment*

---

### Description

Creates a plot representing environmental blocks (all or selected) in both spaces, environmental and/or geographic.

### Usage

```
plot_blocks_EG(master, region_border = TRUE, mask_border = FALSE,
                which = "all", block_ID = FALSE, col_all = NULL,
                col_selected = NULL, col_ID = NULL, cex_all = 0.7,
                cex_selected = 1, cex_ID = 1, pch_all = 16,
                pch_selected = 16, add_main = TRUE)

plot_blocks_E(master, which = "all", block_ID = FALSE, col_all = NULL,
                col_selected = NULL, col_ID = NULL, cex_all = 0.7,
                cex_selected = 1, cex_ID = 1, pch_all = 16,
                pch_selected = 16, main = "", xlab = NULL, ylab = NULL)

plot_blocks_G(master, region_border = TRUE, mask_border = FALSE,
                which = "all", block_ID = FALSE, col_all = NULL,
                col_selected = NULL, col_ID = NULL, cex_all = 0.7,
                cex_selected = 1, cex_ID = 1, pch_all = 16, pch_selected = 16)
```

### Arguments

| | |
|---|---|
| master | master_matrix object derived from function [prepare_master_matrix](#) or master_selection object derived from functions [uniformG_selection](#), [uniformE_selection](#) or [EG_selection](#). Blocks must be defined, see [make_blocks](#). |
| region_border | (logical) whether to add region border to the plot. Default = TRUE. |
| mask_border | (logical) whether to add mask border to the plot. Ignored if mask is not present in master_selection. Default = FALSE. |
| which | (character) blocks to be plotted. Options are "all" or "selected". Default = "all". |
| block_ID | (logical) whether to add a text ID to blocks plotted in environmental space. Default = FALSE. |
| col_all | colors for points in all blocks. The default, NULL, uses a color blind friendly palette to differentiate among distinct blocks when which = "all", or uses a light gray color when which = "selected". See details for explanations of how to define them. |
| col_selected | color for points in selected blocks. Ignored if which = "all". The default, NULL, uses a blue color to represent selected blocks on top of all. |
| col_ID | color for text ID to be added if block_ID = TRUE. The default, NULL, uses the "back". |

| cex_all | (numeric) value defining magnification of points in all blocks relative to the default. Default = 0.7. |
|---|---|
| cex_selected | (numeric) value defining magnification of points in selected blocks relative to the default. Default = 1. |
| cex_ID | (numeric) value defining magnification of text ID to be added if block_ID = TRUE. Default = 1. |
| pch_all | (numeric) integer specifying a symbol when plotting points of all blocks. Default = 16. |
| pch_selected | (numeric) integer specifying a symbol when plotting points of selected blocks. Default = 16. |
| add_main | (logical) whether or not to add fixed titles to the plot. Default = TRUE. Titles added are "Environmental space" and "Geographic space". |
| main | (character) the main title for the plot. |
| xlab | (character) label for the x axis. The default, NULL, uses variable_1. |
| ylab | (character) label for the y axis. The default, NULL, uses variable_2. |

## Details

Defining colors in col_all depends on what is chosen in which. If "all" is chosen, it is convenient to define col_all as a color ramp palette (randomly arranged) or a set of colors depending on the number of blocks in the object defined in master. If "selected" is chosen in which it is recommended to use a single color, preferably a light one, so the selected blocks can be easily identified. See examples.

## Value

A plot showing all the blocks of the region of interest and, if asked, the blocks that were selected. They are show in both spaces, geographic and/or environmental.

## Examples

```
# Data
data("m_matrix", package = "biosurvey")

# Creating blocks
m_blocks <- make_blocks(m_matrix, variable_1 = "PC1",
                        variable_2 = "PC2", n_cols = 10, n_rows = 10,
                        block_type = "equal_area")

plot_blocks_EG(master = m_blocks, block_ID = TRUE)
plot_blocks_E(master = m_blocks)
plot_blocks_G(master = m_blocks)

# Defining your own colors
n_blocks <- length(m_blocks$data_matrix$Block)
your_palette <- sample(heat.colors(n_blocks), n_blocks)
block_factor <- as.factor(m_blocks$data_matrix$Block)
your_colors <- your_palette[block_factor]
```

```
plot_blocks_EG(master = m_blocks, block_ID = TRUE, col_all = your_colors)
```

---

plot_DI                              *Plotting dissimilarity indices withing and among sets of selected sites*

---

### Description

Creates matrix-like plots of dissimilarities found among communities of species in distinct sites selected or sets of sites selected for sampling.

### Usage

```
plot_DI(DI_selected_sites, selection_type = "selections",
        selection_number = 1, values = TRUE,
        col = heat.colors(12, rev = TRUE),
        xlab = "", ylab = "")
```

### Arguments

DI_selected_sites

       list of results obtained with function [selected_sites_DI](#).

selection_type   type of selection to be considered when creating DI matrix plot. Options are: "selections", "random", "E", "G", and "EG". The default, "selections", plots a comparison among all selection types.

selection_number

       (numeric) number of selection to be plotted. Default = 1. Ignored if selection_type = "selections".

values           (logical) whether or not to add values of dissimilarity. Default = TRUE.

col                a list of colors derived from a palette. Default = heat.colors(12, rev = TRUE).

xlab              (character) label for x axis of plot. Default = "Number of sites".

ylab              (character) label for y axis of plot. Default = "Species".

### Value

A plot of a matrix of dissimilarities among sites selected for sampling, or among sets of sampling sites selected. Random is abbreviated as "R" in labels.

### Examples

```
# Data
data("b_pam", package = "biosurvey")
data("m_selection", package = "biosurvey")

# Subsetting base PAM according to selections
sub_pam_all <- subset_PAM(b_pam, m_selection, selection_type = "all")
```

```
# Calculating dissimilarities
DI_sel <- selected_sites_DI(sub_pam_all)

# Plotting
plot_DI(DI_sel)
```

---

plot_PAM_CS                    *Representations of diversity and dispersion indices*

---

### Description

Graphic representations of results from `prepare_PAM_CS`. Plots present the new range-diversity diagram and geographic views of results. Geographic representations are only possible when significant analyses were performed.

### Usage

```
plot_PAM_CS(PAM_CS, add_significant = FALSE,
            add_random_values = FALSE, col_all = "#CACACA",
            col_significant_low = "#6D6D6D",
            col_significant_high = "#000000",
            col_random_values = "#D2D2D2", pch_all = 1,
            pch_significant_low = 16, pch_significant_high = 16,
            pch_random_values = 1, main = NULL,
            xlab = NULL, ylab = NULL, xlim = NULL, ylim = NULL,
            ylim_expansion = 0.25, las = 1, add_legend = TRUE)

plot_PAM_CS_geo(PAM_CS, xy_coordinates = NULL, col_all = "#CACACA",
                col_significant_low = "#6D6D6D",
                col_significant_high = "#000000", border = NULL,
                pch_all = 16, pch_significant_low = 16,
                pch_significant_high = 16, xlim = NULL, ylim = NULL)
```

### Arguments

PAM_CS          object of class PAM_CS or a base_PAM object containing a PAM_CS object
                as part of PAM_indices. These objects can be obtained using the function
                `prepare_PAM_CS`.

add_significant
                (logical) whether to add statistically significant values using a different symbol.
                Default = FALSE. If TRUE and values indicating significance are not in PAM_CS,
                a message will be printed.

add_random_values
                (logical) whether to add values resulted from the randomization process done
                when preparing PAM_CS. Default = FALSE. Valid only if add_significant =
                TRUE, and randomized values are present in PAM_CS.

col_all            color code or name for all values or those that are not statistically significant.
                   Default = "#CACACA".

col_significant_low
                   color code or name for significant values below confidence limits of random
                   expectations. Default = "#000000".

col_significant_high
                   color code or name for significant values above confidence limits of random
                   expectations. Default = "#000000".

col_random_values
                   color code or name for randomized values. Default = "D2D2D2".

pch_all            point symbol to be used for all values. Defaults = 1 or 16.

pch_significant_low
                   point symbol to be used for significant values below confidence limits of random
                   expectations. Default = 16.

pch_significant_high
                   point symbol to be used for significant values above confidence limits of random
                   expectations. Default = 16.

pch_random_values
                   point symbol to be used for randomized values. Default = 1.

main               main title for the plot. Default = NULL.

xlab               label for the x-axis. Default = NULL.

ylab               label for the y-axis. Default = NULL.

xlim               x limits of the plot (x1, x2). For plot_PAM_CS, the default, NULL, uses the
                   range of normalized richness.

ylim               y limits of the plot. For plot_PAM_CS, the default, NULL, uses the range of
                   the normalized values of the dispersion field. The second limit is increased by
                   adding the result of multiplying it by ylim_expansion, if add_legend = TRUE.

ylim_expansion     value used or expanding the ylim. Default = 0.25.

las                the style of axis labels; default = 1. See par.

add_legend         (logical) whether to add a legend describing information relevant for interpreting
                   the diagram. Default = TRUE.

xy_coordinates     Only required if PAM_CS is of class PAM_CS. A matrix or data.frame containing
                   the columns longitude and latitude (in that order) corresponding to the points in
                   PAM_CS$S_significance_id. Default = NULL

border             color for cell borders of the PAM grid. The default, NULL, does not plot any
                   border.

## Value

For plot_PAM_CS:

A range-diversity plot with values of normalized richness in the x-axis, and normalized values of
the dispersion field index divided by number of species in the y-axis.

For plot_PAM_CS_geo:

A geographic view of the PAM representing the areas or points identified as non statistically significant, significant above random expectations, and significant below random expectations.

## Examples

```
# Data
data("b_pam", package = "biosurvey")

# Preparing data for CS diagram
pcs <- prepare_PAM_CS(PAM = b_pam)

# Plot
plot_PAM_CS(pcs)
```

---

plot_PAM_geo                    *Plot of PAM indices in geography*

---

## Description

Plot of PAM indices in geography

## Usage

```
plot_PAM_geo(PAM, index = "RI", master_selection = NULL,
             region_border = TRUE, mask_border = FALSE,
             selection_type = NULL, selection_number = 1,
             col_pal = NULL, border = NULL, col_sites = NULL,
             col_pre = NULL, pch_sites = 16, pch_pre = 16)
```

## Arguments

| | |
|---|---|
| PAM | object of class base_PAM. |
| index | (character) code for the index to be plotted. Options are: "RI" (Richness), "RIN" (Richness normalized), "DF" (Dispersion field), or "MCC" (Mean composition covariance). Default = "RI". |
| master_selection | |
| | master_selection object derived from functions random_selection, uniformG_selection, uniformE_selection, or EG_selection. |
| region_border | (logical) whether to add region border to the plot. Default = TRUE. |
| mask_border | (logical) whether to add mask border to the plot. Ignored if mask is not present in master_selection. Default = FALSE. |
| selection_type | (character) Type of selection depending on the function used to select sites. The options available are "random" (random_selection), "G" (uniformG_selection), "E" (uniformE_selection), and "EG" (EG_selection). |
| selection_number | |
| | (numeric) number of selection to be plotted. Default = 1. |
| col_pal | color palette function to be used in defining colors for the index to be plotted. The default, NULL, uses a simple color blind friendly palette similar to viridis, purplow. Warmer colors indicate higher values. |

| border | color for cell borders of the PAM grid. The default, NULL, does not plot any border. |
|---|---|
| col_sites | color for selected sites. The default, NULL, uses a red color to represent selected sites. |
| col_pre | color for preselected sites. The default, NULL, uses a purple color to represent preselected sites. Ignored if preselected sites are not present in master_selection. |
| pch_sites | (numeric) integer specifying a symbol when plotting points of selected sites. Default = 16. |
| pch_pre | (numeric) integer specifying a symbol when plotting points of preselected sites. Default = 16. Ignored if preselected sites are not present in master_selection. |

### Value

A plot of index represented in geography. Selected sites are added if master_selection is defined.

### Examples

```
# Data
data("b_pam", package = "biosurvey")

# Plotting
plot_PAM_geo(b_pam, index = "RI")

# You can add a legend with
legend_bar(position = "bottomleft", col = purplow(8), title = "Richness",
           labels = c("Low", "High"))
```

---

plot_SAC                    *Plotting lists of species accumulation curves*

---

### Description

Creates species accumulation curve plots (one or multiple panels) from information contained in lists obtained with the function selected_sites_SAC.

### Usage

```
plot_SAC(SAC_selected_sites, col_mean = "blue", col_CI = "lightblue",
         alpha_mean = 0.7, alpha_CI = 0.2, xlab = "Number of sites",
         ylab = "Species", line_for_multiple = TRUE, main = NULL, ...)
```

### Arguments

| SAC_selected_sites | |
|---|---|
| | nested list of "specaccum" objects obtained with function selected_sites_SAC. |
| col_mean | (character) color for mean value of curve. Default = "blue". |

| | |
|---|---|
| col_CI | (character) color for confidence interval region for the curve. Default = "light-blue". |
| alpha_mean | (numeric) alpha level for line representing the mean, values from 0 to 1. Default = 0.7. Values close to 0 increase transparency. |
| alpha_CI | (numeric) alpha level for the region representing the confidence interval. Default = 0.2. |
| xlab | (character) label for x-axis of plot. Default = "Number of sites". |
| ylab | (character) label for y-axis of plot. Default = "Species". |
| line_for_multiple | |
| | (logical) whether to plot SACs only as lines when multiple objects are in one or more of the internal lists in SAC_selected_sites. Default = TRUE. |
| main | (character) title or titles for plots. The default, NULL, adds titles according to names of elements in SAC_selected_sites. |
| ... | other arguments to be passed to plot method for objects of class "specaccum". |

## Value

A plot of "specaccum" objects. Multiple panels will be plotted if SAC_selected_sites list contains more than one element.

## Examples

```
# Data
data("b_pam", package = "biosurvey")
data("m_selection", package = "biosurvey")

# Subsetting base PAM according to selections
sub_pam_all <- subset_PAM(b_pam, m_selection, selection_type = "all")

SACs <- selected_sites_SAC(PAM_subset = sub_pam_all, selection_type = "all")

# Plotting
plot_SAC(SACs)
```

---

plot_sites_EG                *Representation of sites selected to be surveyed*

---

## Description

Plots representing sites (all and selected for survey) in environmental and/or geographic space.

**Usage**

```
plot_sites_EG(master_selection, selection_type, variable_1 = NULL,
              variable_2 = NULL, selection_number = 1,
              region_border = TRUE, mask_border = FALSE, col_all = NULL,
              col_sites = NULL, col_pre = NULL, cex_all = 0.7,
              cex_sites = 1, cex_pre = 1, pch_all = 16, pch_sites = 16,
              pch_pre = 16, add_main = TRUE)

plot_sites_E(master_selection, selection_type, variable_1 = NULL,
             variable_2 = NULL, selection_number = 1, col_all = NULL,
             col_sites = NULL, col_pre = NULL, cex_all = 0.7,
             cex_sites = 1, cex_pre = 1, pch_all = 16, pch_sites = 16,
             pch_pre = 16, main = "", xlab = NULL, ylab = NULL)

plot_sites_G(master_selection, selection_type, selection_number = 1,
             region_border = TRUE, mask_border = FALSE, col_all = NULL,
             col_sites = NULL, col_pre = NULL, cex_all = 0.7,
             cex_sites = 1, cex_pre = 1, pch_all = 16, pch_sites = 16,
             pch_pre = 16)
```

**Arguments**

master_selection
:   master_selection object derived from functions [random_selection](), [uniformG_selection](), [uniformE_selection](), or [EG_selection]().

selection_type
:   (character) type of selection depending on the function used to select sites. The options available are "random" ([random_selection]()), "G" ([uniformG_selection]()), "E" ([uniformE_selection]()), and "EG" ([EG_selection]()).

variable_1
:   (character or numeric) name or position of the first variable (x-axis) to be plotted in environmental space. Default = NULL, required when selection_type = "random" or "G".

variable_2
:   (character or numeric) name or position of the second variable (y-axis) to be plotted in environmental space. It must be different from the first one. Default = NULL, required when selection_type = "random" or "G".

selection_number
:   (numeric) number of selection to be plotted. Default = 1.

region_border
:   (logical) whether to add region border to the plot. Default = TRUE.

mask_border
:   (logical) whether to add mask border to the plot. Ignored if mask is not present in master_selection. Default = FALSE.

col_all
:   colors for points in all points in the region of interest. The default, NULL, uses a light gray color.

col_sites
:   color for selected sites. The default, NULL, uses a blue color to represent selected sites.

col_pre
:   color for preselected sites. The default, NULL, uses a red color to represent preselected sites. Ignored if preselected sites are not present in master_selection.

| | |
|---|---|
| cex_all | (numeric) value defining magnification of all points relative to the default. Default = 0.7. |
| cex_sites | (numeric) value defining magnification of selected sites relative to the default. Default = 1. |
| cex_pre | (numeric) value defining magnification of preselected sites relative to the default. Default = 1. Ignored if preselected sites are not present in master_selection. |
| pch_all | (numeric) integer specifying a symbol when plotting all points. Default = 16. |
| pch_sites | (numeric) integer specifying a symbol when plotting points of selected sites. Default = 16. |
| pch_pre | (numeric) integer specifying a symbol when plotting points of preselected sites. Default = 16. Ignored if preselected sites are not present in master_selection. |
| add_main | (logical) whether or not to add fixed titles to the plot. Default = TRUE. Titles added are "Environmental space" and "Geographic space". |
| main | (character) the main title for the plot. |
| xlab | (character) label for the x axis. The default, NULL, uses variable_1. |
| ylab | (character) label for the y axis. The default, NULL, uses variable_2. |

## Value

plot_sites_EG returns a two-panel plot showing the selected sites. They are show in both spaces, geographic and environmental.

plot_sites_E returns a plot of selected sites in environmental space.

plot_sites_G returns a plot of selected sites in geographic space.

## Examples

```
# Data
data("m_selection", package = "biosurvey")

# Plotting
plot_sites_EG(m_selection, selection_type = "E")
plot_sites_E(m_selection, selection_type = "E")
plot_sites_G(m_selection, selection_type = "E")
```

---

| point_sample | *Sample points from a 2D environmental space* |
|---|---|

---

## Description

Sample one or more points from a two dimensional environmental space according to a selection rule and with the possibility of having distinct sets of points to be sampled independently.

## Usage

```
point_sample(data, variable_1, variable_2, n = 1,
             select_point = "E_centroid", id_column = NULL)
```

**Arguments**

| | |
|---|---|
| `data` | matrix or data.frame that contains at least four columns: "Longitude" and "Latitude" to represent geographic position, and two other columns to represent the variables of the 2D environmental space. |
| `variable_1` | (character or numeric) name or position of the first variable (x-axis). |
| `variable_2` | (character or numeric) name or position of the second variable (y-axis). Must be different from the first one. |
| `n` | (numeric) number of points to be selected. If `id_column` is defined this argument indicates the number of points per set. Default = 1. |
| `select_point` | (character) how or which point will be selected. Three options are available: "random", "E_centroid", and "G_centroid". E_ or G_ centroid indicate that the point(s) closest to the respective centroid will be selected. Default = "E_centroid". |
| `id_column` | (character or numeric) name or numeric index of the column in `data` containing identifiers of one or distinct sets of points. If, NULL, the default, only one set is assumed. |

**Value**

A data.frame containing `n` rows corresponding to the point or points that were sampled.

**Examples**

```
# Data
data("m_matrix", package = "biosurvey")

# Sampling points
points_s <- point_sample(m_matrix$data_matrix,
                         variable_1 = "Max_temperature",
                         variable_2 = "Min_temperature", n = 1,
                         select_point = "E_centroid", id_column = NULL)

points_s
```

---

| | |
|---|---|
| `point_sample_cluster` | *Sample points from a 2D environmental space potentially disjoint in geography* |

---

**Description**

Sample one or more points from a two-dimensional environmental space according to a selection rule and with the possibility of having distinct sets of points to be sampled independently. Points to be sampled can be disjoint in geographic space and when that happens two points are selected considering the most numerous clusters.

**Usage**

```
point_sample_cluster(data, variable_1, variable_2, distance_list,
                     n = 1, cluster_method = "hierarchical",
                     select_point = "E_centroid", id_column = NULL)
```

**Arguments**

| | |
|---|---|
| `data` | matrix or data.frame that contains at least four columns: "Longitude" and "Latitude" to represent geographic position, and two other columns to represent the variables of the 2D environmental space. |
| `variable_1` | (character or numeric) name or position of the first variable (x-axis). |
| `variable_2` | (character or numeric) name or position of the second variable (y-axis). Must be different from the first one. |
| `distance_list` | list of vectors of geographic distances among all points. If `id_column` is not defined, only one element in the list is needed, otherwise, `distance_list` must contain as many elements as unique IDs in `id_column`. In the latter case, the names in `distance_list` must match the IDs in `id_column`. |
| `n` | (numeric) number of points that are close to the centroid to be detected. Default = 1. |
| `cluster_method` | (character) there are two options available: "hierarchical" and "k-means". Default = "hierarchical". |
| `select_point` | (character) how or which point will be selected. Three options are available: "random", "E_centroid", and "G_centroid". E_ or G_ centroid indicate that the point(s) closest to the respective centroid will be selected. Default = "E_centroid". |
| `id_column` | (character or numeric) name or numeric index of the column in `data` containing identifiers of one or distinct sets of points. If, NULL, the default, only one set is assumed. |

**Value**

A data.frame containing n rows corresponding to the point or points that were sampled.

**Examples**

```
# Data
data("m_matrix", package = "biosurvey")
data("dist_list", package = "biosurvey")

# Making blocks for analysis
m_blocks <- make_blocks(m_matrix, variable_1 = "PC1", variable_2 = "PC2",
                        n_cols = 10, n_rows = 10, block_type = "equal_area")

datam <- m_blocks$data_matrix
datam <- datam[datam$Block %in% names(dist_list), ]

# Sampling points
point_clus <- point_sample_cluster(datam, variable_1 = "PC1",
                                   variable_2 = "PC2",
```

```
                                 distance_list = dist_list, n = 1,
                                 cluster_method = "hierarchical",
                                 select_point = "E_centroid",
                                 id_column = "Block")
```

---

point_thinning                *Helps in thinning points either in geographic or environmental space*

---

### Description

Point thinning based on user-defined distances in geographic or environmental space.

### Usage

```
point_thinning(data, x_column, y_column, thinning_distance, space,
               max_n_samples = 1, replicates = 10, set_seed = 1)
```

### Arguments

| | |
|---|---|
| data | matrix or data.frame that contains at least two columns. |
| x_column | (character) the name of the x-axis. |
| y_column | (character) the name of the y-axis. |
| thinning_distance | |
| | (numeric) distance for thinning. Units must be selected according to the space, kilometers (km) for geographic and Euclidean distances for environmental space. |
| space | (character) space in which the thinning will be performed. There are two options available: "G", if it will be in geographic space, and "E", if it will be in environmental space. |
| max_n_samples | (numeric) maximum number of samples to chose with most points included. Default = 1. |
| replicates | (numeric) number of thinning replicates. Default = 10. |
| set_seed | (numeric) integer value to specify a initial seed. Default = 1. |

### Value

A list with one or more elements, depending on max_n_samples. Each element is a data.frame containing points retained after thinning. All elements are different in at least one of the selected points.

## Examples

```
# Data
data("m_matrix", package = "biosurvey")
data1 <- m_matrix$data_matrix

# Thinning the points
thin <- point_thinning(data1, x_column = "Longitude", y_column = "Latitude",
                       thinning_distance = 200, space = "G",
                       max_n_samples = 1, replicates = 5, set_seed = 1)
```

---

prepare_base_PAM                *Presence-absence matrix (PAM) linked to a spatial grid*

---

## Description

Prepares a presence-absence matrix (PAM) in which all sites of interest (rows) will have a value for presence or absence of a species of interest (columns). Initial points of interest will be represented by an ID, and longitude and latitude coordinates. The PAM will be linked to a spatial grid.

## Usage

```
prepare_base_PAM(data, format = NULL, master_matrix, cell_size,
                 complete_cover = TRUE, clip_grid = FALSE,
                 indices = "basic", parallel = FALSE, n_cores = NULL,
                 verbose = TRUE)
```

## Arguments

| | |
|---|---|
| data | species geographic ranges to be used to create a presence-absence matrix (PAM). This argument can be: character, data.frame, RasterStack, RasterBrick, list, SpatialPolygonsDataFrame, or SpatialPointsDataFrame. See details for a description of the characteristics of data for each option. |
| format | (character) if data is of class character, available options are: "shp", "gpkg", "geojson", "GTiff", and "ascii". |
| master_matrix | object of class "master_matrix" or "master_selection". See details. |
| cell_size | (numeric) resolution for grid (single number or vector of two numbers) in kilometers (km). |
| complete_cover | (logical) whether or not to include cells of grid partially overlapped with the geographic region of interest contained in master_matrix. Default = TRUE. |
| clip_grid | (logical) whether to clip the spatial grid using the region of interest. Clipping improves visualization but depending on how complex the region of interest is it could take time to perform this task. |
| indices | (character) code for indices to be calculated. Basic indices are calculated all the time, other indices need to be specified. Options are: "all", "basic, "AB", "BW", "BL", "SCSC", "SCSR", "DF", "CC", "WRN", "SRC", "CMSC", and "CMSR". Default = "basic". See details. |

| | |
|---|---|
| parallel | (logical) whether to perform analyses in parallel. Default = FALSE. Not used if data is of class data.frame, RasterStack, or RasterBrick. |
| n_cores | (numeric) number of cores to be used when parallel = TRUE. The default, NULL, uses available cores - 1. |
| verbose | (logical) whether or not to print messages about the process. Default = TRUE. |

### Details

Objects of class "master_matrix" or "master_selection" can be obtained from functions prepare_master_matrix, random_selection, uniformG_selection, uniformE_selection, or uniformEG_selection. The element region or mask if this last is not NULL is used to prepare the spatial grid.

Geographic projection of objects or coordinates involved must be WGS84 (EPSG:4326).

Description of objects to be used as data:

- character.- name of directory containing raster, shapefiles, geopackage, or geojson files representing species geographic ranges. Each file must be named as the species that it represents. All files must be in the same format. If files are in raster format, "GTiff" and "ascii" are acceptable extensions; values in each layer must be 1 (presence) and 0 (absence).
- data.frame.- a table containing three columns. Columns must be in the following order: Longitude, Latitude, Species.
- RasterStack or RasterBrick.- Each layer must be named as the species which range it represents, and values in each layer must be 1 (presence) and 0 (absence).
- list.- a list of RasterLayers that cannot be stacked because of extent or resolution differences. Each element of the list must be named as the species which range it represents, and values in each RasterLayer must be 1 (presence) and 0 (absence).
- SpatialPolygonsDataFrame.- object representing species' geographic ranges. The data.frame associated with the object must contain a column named "Species" to distinguish among features representing each species range.
- SpatialPointsDataFrame.- object of spatial points where each record of a species must be a point. The associated data.frame must contain the following columns (in that order): Longitude, Latitude, Species.

A list of codes and indices that can be calculated is described below. For further details on the way calculations are performed and the meaning of the indices see Soberon and Cavner (2015) doi: 10.17161/bi.v10i0.4801.

| Code | Index | Calculation |
|---|---|---|
| RI | Richness | Basic |
| RA | Range | Basic |
| RIN | Richness normalized | Basic |
| RAN | Range normalized | Basic |
| AB | Additive Beta | Needs to be defined |
| BW | Beta Whittaker | Needs to be defined |
| BL | Beta Legendre | Needs to be defined and DF |
| SCSC | Schluter covariance sites-composition | Needs to be defined and CMSC |
| SCSR | Schluter covariance species-ranges | Needs to be defined and CMSR |
| DF | Dispersion field | Needs to be defined |

| SCC | Shared community composition | Needs to be defined |
| WRN | Wright-Reeves nestedness | Needs to be defined, BW, and DF |
| SRC | Stone-Roberts C-score | Needs to be defined and DF |
| CMSC | Covariance matrix sites-composition | Needs to be defined, DF, and BW |
| CMSR | Covariance matrix species-ranges | Needs to be defined, SCC, and BW |
| MCC | Mean composition covariance | Calculated with CMSC |
| MRC | Mean range covariance | Calculated with CMSR |

### Value

A presence-absence matrix (PAM) of class [base_PAM](#) for the region of interest associated with a SpatialPolygonsDataFrame, as in a grid of `cell_size` resolution. Each grid cell is related to a specific ID and longitude and latitude coordinates. Presence (1) and absence (0) values for each species in every cell of the PAM are included as apart of the data.frame of the SpatialPolygonsDataFrame. PAM indices is returned with the basic indices of biodiversity as default, but can be changed using the argument `indices`.

### See Also

[PAM_indices](#)

### Examples

```
# Data
data("m_matrix", package = "biosurvey")
data("species_data", package = "biosurvey")

# Create base_PAM
b_pam <- prepare_base_PAM(data = species_data, master_matrix = m_matrix,
                          cell_size = 100)
sp::plot(b_pam$PAM)
summary(b_pam$PAM@data[, 1:6])
```

---

prepare_master_matrix    *Prepare a base object to perform further analyses*

---

### Description

prepare an S3 object that will serve as the base to perform all further analyses. This object will contain geographic and environmental information that will be used to characterize the region of interest.

### Usage

```
prepare_master_matrix(region, variables, mask = NULL,
                      preselected_sites = NULL, do_pca = FALSE,
                      center = TRUE, scale = FALSE,
                      variables_in_matrix = NULL, verbose = TRUE)
```

## Arguments

| | |
|---|---|
| region | SpatialPolygons* of the region of interest; for instance, a country, another type of administrative are, or a protected area. |
| variables | RasterStack or RasterBrick of environmental variables. |
| mask | (optional) SpatialPolygons* object to mask variables and reduce region to an area that is more relevant for analysis (e.g., only areas with natural vegetation cover). Default = NULL. |
| preselected_sites | |
| | data.frame containing sites that must be included in posterior selections of sites for the survey system. Columns must be: "Sites", "Longitude", "Latitude", in that order. |
| do_pca | (logical) whether or not to perform a principal component analysis. Default = FALSE. |
| center | (logical) whether or not to center variables. Argument to be passed to the function prcomp. Default = TRUE. |
| scale | (logical) whether or not to scale the variables. Recommended when variables are in different units. Argument to be passed to the function prcomp. Default = FALSE. |
| variables_in_matrix | |
| | (character) name of variables to include in matrix. If NULL (the default) all variables will be included. |
| verbose | (logical) whether or not to print messages about the process. Default = TRUE. |

## Details

This function helps in preparing all data as needed for further analyses aiming to define a survey sampling system considering geographic and environmental spaces in the region of interest.

If mask is defined all analyses will be restricted to such an area. If mask is not fully contained by region, the mask used for reducing variables, and returned as part of the S3 object (master_matrix) is the intersection between them.

If preselected_sites is defined, environmental values and, if do_pca = TRUE, principal components are added to such records. These records and their characteristics will be considered in further analyses.

## Value

An S3 object of class master_matrix containing the following elements:

- data_matrix: a date.frame with information about geographic location of raster cells, initial environmental data, and if do_pca is TRUE, the first two principal components derived from original data.
- region: a SpatialPolygons* representing the region of interest.
- mask: SpatialPolygons* object used. NULL if mask was not defined.
- preselected_sites: sites defined by used. NULL if preselected_sites was not defined.

- raster_base: a RasterLayer representing the raster masked to region or mask, which will be used for plotting purposes.

- PCA_results: if do_pca is TRUE, other results from principal component analysis. If FALSE, PCA_results element of the object is NULL.

## Examples

```
# Data
data("mx", package = "biosurvey")
variables <- raster::stack(system.file("extdata/variables.tif",
                                       package = "biosurvey"))

# Create master matrix object
m_matrix <- prepare_master_matrix(region = mx, variables = variables,
                                  do_pca = TRUE, center = TRUE, scale = TRUE)
```

---

| prepare_PAM_CS | *Preparing data for new range-diversity plot* |
|---|---|

---

## Description

Preparation of data and details to create range-diversity plots.

## Usage

```
prepare_PAM_CS(PAM, exclude_column = NULL, id_column = NULL,
               significance_test = FALSE, randomization_iterations = 100,
               CL = 0.05, picante_iterations = NULL,
               keep_randomizations = FALSE, parallel = FALSE,
               n_cores = NULL)
```

## Arguments

PAM                       matrix, data.frame, or base_PAM object containing information on presence and
                          absence of species for a set of sites. Sites are organized in the rows and species
                          in the columns. See details.

exclude_column   (optional) name or numeric index of columns to be excluded. Default = NULL.

id_column         (optional) name or numeric index of column containing the ID of sites (cells of
                          the PAM). Default = NULL.

significance_test
                          (logical) whether to perform a test to detect sites (cells) that are statistically
                          significant (i.e., the pattern detected can be distinguished from random expecta-
                          tions). Default = FALSE.

randomization_iterations
                          (numeric) number of iterations for the randomization test used to calculate sta-
                          tistical significance. Default = 100.

| | |
|---|---|
| CL | (numeric) confidence limit to detect statistically significant values. Default = 0.05. |
| picante_iterations | |
| | (numeric) number of iterations to be used for each matrix randomization process (to be done randomization_iterations times). This process is done using the function randomizeMatrix from the package picante. The default, NULL, uses 2 * sum(PAM). |
| keep_randomizations | |
| | (logical) whether to keep a matrix with all values from the randomization process. Default = FALSE. |
| parallel | (logical) whether to perform analyses in parallel. Default = FALSE. |
| n_cores | (numeric) number of cores to be used when parallel = TRUE. The default, NULL, uses available cores - 1. |

## Details

Range-diversity plot allow explorations of patterns of biodiversity in a region based on the data of presence-absence matrices. The plots to be produced using the information prepared here are a modification of those presented in Arita et al. (2011) doi: 10.1111/j.14668238.2011.00662.x.

## Value

An S3 object of class PAM_CS if PAM is a matrix or data.frame, otherwise, an object of class base_PAM that contains the PAM_CS object as a part of PAM_indices.

Significant values are presented as a vector in which 0 means non-significant, and 1 and 2 represent significant values below and above confidence limits of random expectations, respectively.

## Examples

```
# Data
data("b_pam", package = "biosurvey")

# Preparing data for CS diagram
pcs <- prepare_PAM_CS(PAM = b_pam)

summary(pcs$PAM_indices$CS_diagram)
```

---

| preselected | *Example of a data.frame of preselected sites* |
|---|---|

---

## Description

A data.frame with 5 rows and three columns: "Site", "Longitude", and "Latitude".

## Usage

```
preselected
```

## Format

data.frame:

**Site** name of preselected sites

**Longitude** x coordinates

**Latitude** y coordinates

## Examples

```
data("preselected", package = "biosurvey")

print(preselected)
```

---

preselected_dist_mask  *Helper to create objects to detect points to close to preselected sites*

---

## Description

Helper to create objects to detect points to close to preselected sites

## Usage

```
preselected_dist_mask(master, expected_points, space, variable_1 = NULL,
                      variable_2 = NULL, use_blocks = FALSE,
                      verbose = TRUE)
```

## Arguments

| | |
|---|---|
| master | master_matrix object derived from function [prepare_master_matrix](#) or master_selection object derived from functions [random_selection](#), [uniformE_selection](#), or [EG_selection](#). |
| expected_points | (numeric) number of survey points (sites) to be selected. |
| space | (character) space in which the thinning will be performed. There are two options available: "G", if it will be in geographic space, and "E", if it will be in environmental space. |
| variable_1 | (character or numeric) name or position of the first variable (x-axis). Default = NULL. |
| variable_2 | (character or numeric) name or position of the second variable (y-axis). Default = NULL. |
| use_blocks | (logical) whether or not to use block centroids instead of all points when space = "E". Default = FALSE. |
| verbose | (logical) whether or not to print messages about the process. Default = TRUE. |

## Value

A list of two elements: the distance used to obtain `expected_points` and a SpatialPolygons-DataFrame object created from preselected_sites in master.

## Examples

```
# Data
data("m_matrix_pre", package = "biosurvey")

# Running
pdm <- preselected_dist_mask(master = m_matrix_pre, expected_points = 20,
                             space = "G")
```

---

print                               *Print a short version of elements in master and base objects*

---

## Description

Print a short version of elements in master and base objects

## Usage

```
## S3 method for class 'master_matrix'
print(x, ...)

## S3 method for class 'master_selection'
print(x, ...)

## S3 method for class 'base_PAM'
print(x, ...)

## S3 method for class 'PAM_subset'
print(x, ...)

## S3 method for class 'PAM_CS'
print(x, ...)
```

## Arguments

x                object of class master_matrix, master_selection, base_PAM, or PAM_subset.

...              further arguments to be passed to or from other methods. Ignored in these functions.

## Value

A short description of objects in the console.

---

purplow            *Simple color palettes*

---

### Description

Simple color palettes

### Usage

```
# sequential palettes
purplow(n)

darkros(n)

daright(n)

# diverging palettes
greeple(n)

bluered(n)
```

### Arguments

n                    number of colors to be in the palette.

### Value

A character vector of hex color codes.

### Examples

```
purplow(3)
darkros(3)
daright(3)
greeple(3)
bluered(3)
```

---

random_selection       *Random selection of survey sites*

---

### Description

Random selection of sites to be sampled in a survey. Sites are selected from a set of points provided in master.

**Usage**

```
random_selection(master, n_sites, n_samplings = 1,
                 use_preselected_sites = TRUE, median_distance_filter = NULL,
                 set_seed = 1, verbose = TRUE, force = FALSE)
```

**Arguments**

master            master_matrix object derived from function `prepare_master_matrix`, or mas-
                  ter_selection object derived from functions `uniformG_selection`, `uniformE_selection`
                  or `EG_selection`.

n_sites           (numeric) number of sites to be selected from master_matrix to be used as
                  sites to be sampled in survey.

n_samplings       (numeric) number of processes of selection, which will turn into multiple op-
                  tions for a process of survey planning. Default = 1.

use_preselected_sites
                  (logical) whether to use sites that have been defined as part of the selected sites
                  previous to any selection. Object in master must contain the site(s) preselected
                  in and element of name "preselected_sites" for this argument to be effective.
                  Default = TRUE.

median_distance_filter
                  (character) optional argument to define a median distance-based filter based on
                  which sets of sampling sites will be selected. The default, NULL, does not apply
                  such a filter. Options are: "max" and "min".

set_seed          (numeric) integer value to specify a initial seed. Default = 1.

verbose           (logical) whether or not to print messages about the process. Default = TRUE.

force             (logical) whether to replace existing set of sites selected with this method in
                  master.

**Details**

Survey sites are selected randomly from the entire set of points provided in master$data_matrix.
Considering the environmental space, as points are selected randomly, sites that have environmental
conditions that are common in the region of interest will be sampled more that other sites that
present condition that are not as common.

To see how common or rare are distinct environments in the region of interest, the function `explore_data_EG`
can be used. Common environmental conditions are those that are present in areas of higher density
in one of the plots obtained with `explore_data_EG`.

As multiple sets could result from selection, the argument of the function median_distance_filter
could be used to select the set of sites with the maximum ("max") or minimum ("min") median dis-
tance among selected sites. Option "max" will increase the geographic distance among sampling
sites, which could be desirable if the goal is to cover the region of interest more broadly. The other
option, "min", could be used in cases when the goal is to reduce resources and time needed to
sample such sites.

### Value

A [master_selection](#) object (S3) with an element called selected_sites_random containing one or more sets of selected sites.

### See Also

[uniformG_selection](#), [uniformE_selection](#), [EG_selection](#), [plot_sites_EG](#)

### Examples

```
# Data
data("m_matrix", package = "biosurvey")

r_selection <- random_selection(m_matrix, n_sites = 20, n_samplings = 5)
```

---

refill_PAM_indices    *Helper to refill a list of PAM indices with new or more results*

---

### Description

Helper to refill a list of PAM indices with new or more results

### Usage

```
refill_PAM_indices(initial_index_list, new_index_list)
```

### Arguments

initial_index_list

list of PAM indices to be refill. Indices present in this list and absent in new_index_list are maintained.

new_index_list  list of PAM indices to be used to refill initial_index_list. New indices are included in the resulting list. Indices present in both lists are updated using the values of this list.

### Value

A list of PAM indices containing old and new values for its indices.

---

rlist_2data                  *Creates a data.frame of species' references from a list of raster layers*

---

### Description

Creates a data.frame of species' references that contains longitude, latitude, and species name, using a list of raster layers as input. Useful when raster layers have distinct extent or resolution.

### Usage

```
rlist_2data(raster_list, parallel = FALSE, n_cores = NULL)
```

### Arguments

raster_list    list of RasterLayer objects. Each raster layer must be named as the species that
               it represents, and values in each layer must be 1 (presence) and 0 (absence).

parallel       (logical) whether to perform analyses in parallel. Default = FALSE.

n_cores        (numeric) number of cores to be used when parallel = TRUE. The default,
               NULL, uses available cores - 1.

### Value

A data.frame of species geographic records derived from values of presence in each layer from the list of raster layers.

### Examples

```
# Data
rsp <- raster::stack(system.file("extdata/sp_layers.tif",
                     package = "biosurvey"))
names(rsp) <- paste0("Species_", 1:5)

rlist <- lapply(1:5, function(x) {rsp[[x]]})

# Species data from RasterStack
sp_data <- rlist_2data(raster_list = rlist)
summary(sp_data)
```

---

selected_sites_DI            *Dissimilarity indices from PAM_subset*

---

### Description

Computes dissimilarity indices for each set of selected sites contained in elements of PAM_subset that contain information of species incidence (presence-absence). Calculations are done also among sets of selected sites.

### Usage

```
selected_sites_DI(PAM_subset, selection_type = "all", method = "jaccard",
                  verbose = TRUE, ...)
```

### Arguments

| | |
|---|---|
| PAM_subset | object of class PAM_subset obtained using the function subset_PAM. |
| selection_type | type of selection to be considered when creating dissimilarity matrices for elements in PAM_subset. Options are: "all", "random", "E", "G", and "EG". The default, "all", uses all selection types present in PAM_subset. |
| method | (character) dissimilarity index to be passed to function vegdist. Default = "jaccard". See details. |
| verbose | (logical) whether or not to print messages about the process. Default = TRUE. |
| ... | other arguments to be passed to function vegdist. |

### Details

Important details about the process performed to compute dissimilarity indices can be seen in the documentation of vegdist.

### Value

A list containing:

- Dissimilarity matrices for all PAMs reduced based on distinct sets of selected sites.
- A matrix summarizing incidences from all sets of selected sites.
- A dissimilarity matrix for the summary of incidences for all sets of selected sites.
- The result of clustering sets of selected sites based on dissimilarities.

### Examples

```
# Data
data("b_pam", package = "biosurvey")
data("m_selection", package = "biosurvey")

# Subsetting base PAM according to selections
```

```
sub_pam_all <- subset_PAM(b_pam, m_selection, selection_type = "all")

# Calculating dissimilarities
DI_sel <- selected_sites_DI(sub_pam_all)
```

---

selected_sites_PAM         *Helper to subset PAM according to selected sites*

---

### Description

Helper to subset PAM according to selected sites

### Usage

```
selected_sites_PAM(selected_sites, base_PAM)
```

### Arguments

selected_sites  list of selected sites. See any of the main functions to perform such a se-
                lection: random_selection, uniformG_selection, uniformE_selection, or
                EG_selection.
base_PAM        object of class base_PAM obtained using the function prepare_base_PAM.

### Value

A list of selected site data.frames with information of PAM added as additional columns.

---

selected_sites_SAC         *Species accumulation curves from PAM_subset*

---

### Description

Creates species accumulation curves for each set of selected sites contained in elements of PAM_subset
that contain information of species incidence (presence-absence).

### Usage

```
selected_sites_SAC(PAM_subset, selection_type = "all", method = "exact", ...)
```

### Arguments

PAM_subset      object of class PAM_subset obtained using the function subset_PAM.
selection_type  type of selection to be considered when creating SAC for elements in PAM_subset.
                Options are: "all", "random", "E", "G", #' and "EG". The default, "all", uses all
                selection types present in PAM_subset.
method          (character) species accumulation method to be passed to function specaccum.
                Default = "exact".
...             other arguments to be passed to function specaccum.
```

## Details

Important details about the process performed to obtain each of the SACs can be seen in the help
for function `specaccum`.

## Value

A list of species accumulation curves (SACs, "specaccum" objects) for all sets of selected sites
according to option defined in selection_type.

## Examples

```
# Data
data("b_pam", package = "biosurvey")
data("m_selection", package = "biosurvey")

# Subsetting base PAM according to selections
sub_pam_all <- subset_PAM(b_pam, m_selection, selection_type = "all")

SACs <- selected_sites_SAC(PAM_subset = sub_pam_all, selection_type = "all")
```

---

| spdf_2data | *Creates a data.frame of species' references from SpatialPolygons-DataFrame* |
|---|---|

---

## Description

Creates a data.frame of species' references that contains identifiers of position and species name,
using a SpatialPolygonsDataFrame as input.

## Usage

```
spdf_2data(spdf_object, spdf_grid, parallel = FALSE, n_cores = NULL)
```

## Arguments

| | |
|---|---|
| spdf_object | SpatialPolygonsDataFrame representing species' geographic distributions. The data.frame associated with the object must contain a column named "Species" to distinguish among features. |
| spdf_grid | geographic grid for the region of interest (output of function `grid_from_region`). |
| parallel | (logical) whether to perform analyses in parallel. Default = FALSE. |
| n_cores | (numeric) number of cores to be used when `parallel` = TRUE. The default, NULL, uses available cores - 1. |

## Value

A data.frame of species' found in distinct positions (defined with identifiers); includes two columns:
"ID" and "Species".

## Examples

```
# Data
data("species_data", package = "biosurvey")
data("mx", package = "biosurvey")

# GRID
grid_reg <- grid_from_region(region = mx, cell_size = 100)

# Species data from polygons
sp_data <- spdf_2data(spdf_object = species_data, spdf_grid = grid_reg)
summary(sp_data)
```

---

| species_data | *Example of species ranges as SpatialPolygonsDataFrame* |

---

## Description

An object of class SpatialPolygonsDataFrame.

## Usage

```
species_data
```

## Format

data.frame:

**data** data.frame with 25 rows and 1 column

**polygons** SpatialPolygons

**proj4string** object of class CRS

## Examples

```
data("species_data", package = "biosurvey")

species_data
```

---

sp_data                           *Example of a data.frame of species' found in distinct positions*

---

### Description

A data.frame with 590 rows and two columns: "ID" and "Species". It is the output of the function spdf_2data.

### Usage

```
sp_data
```

### Format

data.frame:

**ID** identifier of position

**Species** different species in the table

### Examples

```
data("sp_data", package = "biosurvey")

summary(sp_data)
```

---

sp_layers                         *Example of stack of layers of suitable and unsuitable conditions for species*

---

### Description

An object of class RasterStack containing information about suitable and unsuitable conditions for five species.

### Format

RasterStack with 109 rows, 182 columns, 19838 cells, and 5 layers:

**RasterLayer** suitable (1) and unsuitable (0) conditions

### Examples

```
sp_layers <- raster::stack(system.file("extdata/sp_layers.tif",
                           package = "biosurvey"))

sp_layers
```

---

sp_occurrences                     *Occurrence records for the species Parides gundlachianus*

---

### Description

Dataset containing geographic coordinates of a Cuban butterfly.

### Usage

```
sp_occurrences
```

### Format

A data.frame with 19 rows and 3 columns.

**name**  character, species scientific name.

**longitude**  numeric, longitude values.

**latitude**  numeric, latitude values.

### Source

<https://www.gbif.org/>

### Examples

```
data("sp_occurrences", package = "biosurvey")
head(sp_occurrences)
```

---

stack_2data                     *Creates a data.frame of species' references from RasterStack*

---

### Description

Creates a data.frame of species' references that contains longitude, latitude, and species name, using a RasterStack or a RasterBrick as input.

### Usage

```
stack_2data(species_layers)
```

### Arguments

species_layers  RasterStack or RasterBrick object. Each layer must be named as the species that it represents, and values in each layer must be 1 (presence) and 0 (absence).

## Value

A data.frame of species geographic records derived from values of presence in each layer from the RasterStack.

## Examples

```
# Data
rsp <- raster::stack(system.file("extdata/sp_layers.tif",
                                 package = "biosurvey"))
names(rsp) <- paste0("Species_", 1:5)

# Species data from RasterStack
sp_data <- stack_2data(species_layers = rsp)
summary(sp_data)
```

---

subset_PAM                *Subset PAM according to selected sites*

---

## Description

Subsets of a base_PAM object according to survey sites contained in a master_selection object.

## Usage

```
subset_PAM(base_PAM, master_selection, selection_type = "all")
```

## Arguments

base_PAM          object of class base_PAM obtained using the function prepare_base_PAM.

master_selection
                  object of class master_selection. This object can be obtained using the functions: random_selection, uniformG_selection, uniformE_selection, or EG_selection.

selection_type    type of selection to be considered to subset base_PAM. Options are: "all", "random", "E", "G", and "EG". The default, "all", uses all selection types present in master_selection.

## Value

An object of class PAM_subset containing the original base_PAM and other subsets of the PAM according to selection_type.

**Examples**

```
# Data
data("b_pam", package = "biosurvey")
data("m_selection", package = "biosurvey")

# Subsetting base PAM according to selections
## only uniform in G
sub_pam_G <- subset_PAM(b_pam, m_selection, selection_type = "G")

## All at the time
sub_pam_all <- subset_PAM(b_pam, m_selection, selection_type = "all")
```

---

summary                              *Summary of attributes and results*

---

**Description**

Summary of attributes and results

**Usage**

```
## S3 method for class 'master_matrix'
summary(object, ...)

## S3 method for class 'master_selection'
summary(object, nrow = 6, ncol = 2, ...)

## S3 method for class 'base_PAM'
summary(object, ...)

## S3 method for class 'PAM_subset'
summary(object, ...)

## S3 method for class 'PAM_CS'
summary(object, ...)
```

**Arguments**

| | |
|---|---|
| object | object of class master_matrix or master_selection. |
| ... | additional arguments affecting the summary produced. Ignored in these functions. |
| nrow | number of rows to be printed for selected_sites in a master_selection object. |
| ncol | number of columns to be printed for selected_sites in a master_selection object. |

**Value**

A printed summary.

---

uniformE_selection        *Selection of survey sites maximizing uniformity in environmental space*

---

### Description

Selection of sites to be sampled in a survey, with the goal of maximizing uniformity of points in environmental space.

### Usage

```
uniformE_selection(master, variable_1 = NULL, variable_2 = NULL,
                   selection_from = "all_points", expected_points,
                   guess_distances = TRUE, initial_distance = NULL,
                   increase = NULL, max_n_samplings = 1,
                   replicates = 10, use_preselected_sites = TRUE,
                   median_distance_filter = NULL, set_seed = 1,
                   verbose = TRUE, force = FALSE)
```

### Arguments

| | |
|---|---|
| master | master_matrix object derived from function [prepare_master_matrix](#) or master_selection object derived from functions [random_selection](#), [uniformG_selection](#), or [EG_selection](#). |
| variable_1 | (character or numeric) name or position of the first variable (x-axis). If the function [make_blocks](#) was used in a previous step, the default, NULL, will use the same two variables, otherwise this argument must be defined. |
| variable_2 | (character or numeric) name or position of the second variable (y-axis). If the function [make_blocks](#) was used in a previous step, the default, NULL, will use the same two variables, otherwise this argument must be defined. |
| selection_from | (character) set of points to perform the selection from. Two options are available, "all_points" or "block_centroids". The first option picks the points from all points in the environmental cloud, and the second one selects points only from centroids of environmental blocks. See [make_blocks](#). Default = "all_points". |
| expected_points | |
| | (numeric) number of survey points (sites) to be selected. |
| guess_distances | |
| | (logical) whether or not to use internal algorithm to automatically select initial_distance and increase. Default = TRUE. If FALSE, initial_distance and increase must be defined. |
| initial_distance | |
| | (numeric) Euclidean distance to be used for a first process of thinning and detection of remaining points. Default = NULL. |
| increase | (numeric) initial value to be added to or subtracted from initial_distance until reaching the number of expected_points. Default = NULL. |

max_n_samplings

        (numeric) maximum number of samples to be chosen after performing all thinning `replicates`. Default = 1.

replicates        (numeric) number of thinning replicates. Default = 10.

use_preselected_sites

        (logical) whether to use sites that have been defined as part of the selected sites previous any selection. Object in `master` must contain the site(s) preselected in and element of name "preselected_sites" for this argument to be effective. Default = TRUE. See details for more information on the approach used.

median_distance_filter

        (character) optional argument to define a median distance-based filter based on which sets of sampling sites will be selected. The default, NULL, does not apply such a filter. Options are: "max" and "min".

set_seed          (numeric) integer value to specify a initial seed. Default = 1.

verbose           (logical) whether or not to print messages about the process. Default = TRUE.

force             (logical) whether to replace existing set of sites selected with this method in `master`.

### Details

Survey sites are selected in ways in which points will be uniformly dispersed in environmental space, helping to select sites that present different environmental conditions across the area of interest. This type of selection is very useful to include, in the selected sites, distinct environmental combinations existent in the area of interest. However, as the distribution of climatic or other environmental combinations is not uniform in geography, the sites selected with this function could appear clustered when looked in a map.

Exploring the geographic and environmental spaces of the region of interest would be a crucial first step before selecting survey sites. Such explorations can be done using the function [explore_data_EG](#).

If use_preselected_sites = TRUE and such sites are included as an element in the object in `master`, the approach for selecting uniform sites in environmental space is different than what was described above. User-preselected sites will always be part of the sites selected. Other points are selected based on an algorithm that searches for sites that are uniformly distributed in environmental space but at a distance from preselected sites that helps in maintaining uniformity. Note that preselected sites will not be processed; therefore, uniformity of such points cannot be warrantied. As multiple sets could result from selection, the argument of the function median_distance_filter could be used to select the set of sites with the maximum ("max") or minimum ("min") median distance among selected sites. Option "max" will increase the geographic distance among sampling sites, which could be desirable if the goal is to cover the region of interest more broadly. The other option, "min", could be used in cases when the goal is to reduce resources and time needed to sample such sites.

### Value

A [master_selection](#) object (S3) with an element called selected_sites_E containing one or more sets of selected sites.

## See Also

[uniformG_selection](), [random_selection](), [EG_selection](), [make_blocks](), [plot_sites_EG]()

## Examples

```
# Data
data("m_matrix", package = "biosurvey")

# Making blocks for analysis
m_blocks <- make_blocks(m_matrix, variable_1 = "PC1",
                        variable_2 = "PC2", n_cols = 10, n_rows = 10,
                        block_type = "equal_area")

# Checking column names
colnames(m_blocks$data_matrix)

# Selecting sites uniformly in E space
# because the make_blocks function was used, the same two variables will be
# used by default.
selectionE <- uniformE_selection(m_blocks, selection_from = "block_centroids",
                                 expected_points = 15, max_n_samplings = 1,
                                 replicates = 5, set_seed = 1)
```

---

uniformG_selection          *Selection of survey sites maximizing uniformity in geography*

---

## Description

Selection of sites to be sampled in a survey, with the goal of maximizing uniformity of points in geographic space.

## Usage

```
uniformG_selection(master, expected_points, guess_distances = TRUE,
                   initial_distance = NULL, increase = NULL,
                   max_n_samplings = 1, replicates = 10,
                   use_preselected_sites = TRUE,
                   median_distance_filter = NULL, set_seed = 1,
                   verbose = TRUE, force = FALSE)
```

## Arguments

master          master_matrix object derived from function [prepare_master_matrix]() or master_selection object derived from functions [random_selection](), [uniformE_selection](), or [EG_selection]().

expected_points

                (numeric) number of survey points (sites) to be selected.

guess_distances

> (logical) whether or not to use internal algorithm to select automatically initial_distance and increase. Default = TRUE. If FALSE, initial_distance and increase must be defined.

initial_distance

> (numeric) distance in km to be used for a first process of thinning and detection of remaining points. Default = NULL.

increase            (numeric) initial value to be added to or subtracted from initial_distance until reaching the number of expected_points. Default = NULL.

max_n_samplings

> (numeric) maximum number of samples to be chosen after performing all thinning replicates. Default = 1.

replicates          (numeric) number of thinning replicates. Default = 10.

use_preselected_sites

> (logical) whether to use sites that have been defined as part of the selected sites previous any selection. Object in master must contain the site(s) preselected in and element of name "preselected_sites" for this argument to be effective. Default = TRUE. See details for more information on the approach used.

median_distance_filter

> (character) optional argument to define a median distance-based filter based on which sets of sampling sites will be selected. The default, NULL, does not apply such a filter. Options are: "max" and "min".

set_seed            (numeric) integer value to specify an initial seed. Default = 1.

verbose             (logical) whether or not to print messages about the process. Default = TRUE.

force               (logical) whether to replace existing set of sites selected with this method in master.

### Details

Survey sites are selected searching for maximum geographic distances among all sites. This approach helps in selecting points that can cover most of the geographic extent of the region of interest. This type of selection could be appropriate when the region of interest has a complex geographic pattern (e.g., an archipelago). This type of selection does not consider environmental conditions in the region of interest, which is why important environmental combinations may not be represented in the final selection of sites.

Exploring the geographic and environmental spaces of the region of interest would be a crucial first step before selecting survey sites. Such explorations can be done using the function [explore_data_EG](#).

If use_preselected_sites = TRUE and such sites are included as an element in the object in master, the approach for selecting uniform sites in geography is different than what was described above. User-preselected sites will always be part of the sites selected. Other points are selected based on an algorithm that searches for sites that are uniformly distributed in geographic space but at a distance from preselected sites that helps in maintaining uniformity. Note that preselected sites will not be processed; therefore, uniformity of such points cannot be warrantied.

As multiple sets could result from selection when the use_preselected_sites is set as FALSE, the argument of the function median_distance_filter could be used to select the set of sites with the maximum ("max") or minimum ("min") median distance among selected sites. The option

"max" will increase the geographic distance among sampling sites, which could be desirable if the goal is to cover the region of interest more broadly. The other option, "min", could be used in cases when the goal is to reduce resources and time needed to sample such sites.

## Value

A [master_selection](#) object (S3) with an element called selected_sites_G containing one or more sets of selected sites.

## See Also

[random_selection](#), [uniformE_selection](#), [EG_selection](#), [plot_sites_EG](#)

## Examples

```
# Data
data("m_matrix", package = "biosurvey")

# Selecting sites uniformly in G space
selectionG <- uniformG_selection(m_matrix, expected_points = 40,
                                 max_n_samplings = 1, replicates = 5)
```

---

unimodal_test                  *Unimodality test for list of one or multiple sets of values*

---

## Description

Test of unimodality based in Hartigans' dip statistic D. Calculations of the statistic and p-value are done as in [dip.test](#).

## Usage

```
unimodal_test(values_list, MC_replicates= 1000)
```

## Arguments

values_list     named list of vectors of class numeric. Names in values_list are required. If only one set of values is used the list must contain only one element.

MC_replicates   (numeric) number of replicates for the Monte Carlo test to calculate p-value. Default = 1000.

## Value

A data.frame with the results of the test.

### Examples

```
# Data
data("dist_list", package = "biosurvey")

# Testing unimodality
u_test <- unimodal_test(values_list = dist_list, MC_replicates = 500)
u_test
```

---

variables                          *Example of variables to be used for preparing a master matrix*

---

### Description

A dataset containing raster variables for an area that is relevant for used in examples included in the
package biosurvey.

### Format

A RasterStack with 109 rows, 190 columns, 20710 cells, and 6 layers:

**Temperature**  temperature, in degrees Celsius times 10.

**Precipitation**  precipitation, in millimeters.

### Source

https://www.worldclim.org/data/index.html

### Examples

```
variables <- raster::stack(system.file("extdata/variables.tif",
                                        package = "biosurvey"))

raster::plot(variables[[1]])
```

---

wgs84_2aed_laea                    *Project spatial points from geographic coordinates*

---

### Description

Project spatial points from geographic coordinates

### Usage

```
wgs84_2aed_laea(data, longitude, latitude, which = "ED")
```

## Arguments

| | |
|---|---|
| `data` | matrix or data.frame that contains at least two columns, one with longitude information and the other with latitude information. |
| `longitude` | (character) the name of the column that contains the longitude information. |
| `latitude` | (character) the name of the column that contains the latitude information. |
| `which` | (character) type of projection. There are two options available: "ED", for Azimuthal Equidistant and "EA", for Lambert Azimuthal Equal-Area. Default = "ED". |

## Value

SpatialPointsDataFrame projected to an option in `which`.

## Examples

```
data("sp_occurrences", package = "biosurvey")

sp_occ <- wgs84_2aed_laea(sp_occurrences, longitude = "longitude",
                          latitude = "latitude", which = "EA")
```

# Index