

# Package: Mappinguari (via r-universe)

September 18, 2024

**Type** Package

**Title** Process-Based Biogeographical Analysis

**Version** 2.0.1

**Description** Facilitates the incorporation of biological processes in biogeographical analyses. It offers conveniences in fitting, comparing and extrapolating models of biological processes such as physiology and phenology. These spatial extrapolations can be informative by themselves, but also complement traditional correlative species distribution models, by mixing environmental and process-based predictors. Caetano et al (2020) <[doi:10.1111/oik.07123](https://doi.org/10.1111/oik.07123)>.

**Depends** R (>= 3.5)

**License** GPL-2

**Encoding** UTF-8

**LazyData** true

**URL** <https://github.com/gabrielhoc/Mappinguari>

**BugReports** <https://github.com/gabrielhoc/Mappinguari/issues>

**Suggests** geosphere, mgcv

**Imports** dplyr, magrittr, parallel, raster, rlang, stringr, testthat

**RoxygenNote** 7.2.1

**Repository** <https://gabrielhoc.r-universe.dev>

**RemoteUrl** <https://github.com/gabrielhoc/Mappinguari>

**RemoteRef** HEAD

**RemoteSha** 5bb10b933b65e8603fa184a6e29b2a153bf6b017

## Contents

clean_points . . . . .	2
get_predict . . . . .	3
get_rasters . . . . .	4

Mapinguari . . . . .	5
multi_extract . . . . .	6
sin_h . . . . .	7
transform_rasters . . . . .	7
TtorquatusBreeding . . . . .	8
TtorquatusDistribution . . . . .	9
TtorquatusGradient . . . . .	10
TtorquatusOperative . . . . .	10
TtorquatusPerformance . . . . .	11

<b>Index</b>	<b>12</b>
--------------	-----------

---

clean_points	<i>Clean occurrence records</i>
--------------	---------------------------------

---

## Description

clean\_points Eliminates species occurrence records that are too close to each other or at undesired locations.

## Usage

```
clean_points(
  coord,
  merge_dist,
  coord_col = c("Lon", "Lat"),
  filter_layer = NULL,
  na.rm = FALSE
)
```

## Arguments

coord	data.frame. Data frame containing longitudes (Lon) and latitudes (Lat) of occurrence records of a species.
merge_dist	numeric. Maximum distance between points to be merged, in meters.
coord_col	vector of strings or integers. If x has more than two columns, indicate the name or position of longitude and latitude columns
filter_layer	RasterLayer. Binary raster with 1 representing the regions where records should be kept and 0 the regions where they should be eliminated.
na.rm	logical. if TRUE, remove lines with NA in any coordinate.

## Value

Data frame with remaining longitudes and latitudes.

**Examples**

```
## Not run:
#First, we need to obtain an altitude raster to filter by altitude.
library(raster)
alt <- raster::getData("alt", country = "BRA", mask = TRUE)

# Then, we clean the points
TtorquatusDistribution_clean <-
  clean_points(coord = TtorquatusDistribution,
              merge_dist = 20000,
              filter_layer = !is.na(alt))

## End(Not run)
```

---

get\_predict

*Creates vectorized predict functions from models.*


---

**Description**

get\_predict Takes inputted models and create vectorized functions able to get the model predictions for any value inputted. Also outputs a table comparing models. nls gam glm lm randomForest gbm gls bam

**Usage**

```
get_predict(models, separator = "_", ...)
```

**Arguments**

models	list. List with models to create the prediction function. The model objects must have methods for function 'predict'.
separator	character. Character that separates variable names, years and scenarios.
...	additional arguments to be passed to predict function (specific for the method of the models supplied).

**Value**

Returns a list of vectorized functions that get predictions for the models inputted. The functions generated do not perform lazy evaluation, the user must be explicit

**Examples**

```
library(mgcv)

perf_no_size <-
  gamm(performance ~ s(temp, bs = 'cs'),
       random = list(id = ~ 1),
       data = TtorquatusPerformance)
```

```

perf_size <-
  gamm(performance ~ s(temp, bs = 'cs') + size,
        random = list(id = ~ 1),
        data = TtorquatusPerformance)

perf_functions <- get_predict(list(perf_s = perf_size,
                                  perf_ns = perf_no_size),
                             type = "response")

perf_nsFUN <- perf_functions$perf_ns
perf_sFUN <- perf_functions$perf_s

perf_nsFUN(temp = 30)
perf_sFUN(temp = 30, size = 70)
perf_nsFUN(temp = 30:35)
perf_sFUN(temp = 30, size = 70:75)
perf_sFUN(temp = 30:35, size = 70:75)

```

---

get\_rasters

*Retrieve and organize spatial rasters.*


---

### Description

get\_rasters Loads rasters from directory and returns them in an organized list of specified scenarios.

### Usage

```

get_rasters(
  var = NULL,
  scenario = NULL,
  raster_path = NULL,
  ext = c(-180, 180, -60, 90),
  coord_col = c("Lon", "Lat"),
  margin = 0,
  separator = "_"
)

```

### Arguments

var	character. Names of variables to be loaded.
scenario	character. Names of scenarios for the variables.
raster_path	character. Path to folder with raster files. See <a href="#">writeFormats</a> for supported formats
ext	numeric, data, frame or Extent object. Extent to crop rasters.
coord_col	character. Names of columns containing Longitude and Latitude.

margin            numeric. Additional distance to be added to margin of extent, in degrees.  
separator        character. Character that separates variable names and scenario names.

### Value

Returns a list of raster stacks for the variables required, organized by year/scenario combination.

### Examples

```
## Not run:  
# replace rasterpath with the directory on your computer containing worldclim data  
  
Fulanus_Ecorasters_present <-  
  get_rasters(  
    var = c('prec', 'tmin', 'tmax'),  
    scenario = 'present',  
    raster_path = "C:/Users/gabri/Dropbox/Mapinguari/global_grids_10_minutes",  
    ext = FulanusDistribution[c(2,3)],  
    margin = 5)  
  
## End(Not run)
```

### Description

Mapinguari provides solutions for incorporating biological processes in biogeographical analysis.

### Mapinguari functions

- [clean\\_points](#)
- [get\\_predict](#)
- [get\\_rasters](#)
- [multi\\_extract](#)
- [transform\\_rasters](#)
- [sin\\_h](#)

---

multi_extract	<i>Gets values from multiple rasters.</i>
---------------	---

---

### Description

multi\_extract Extract values of multiple spatial rasters for a set of geographical coordinates.

### Usage

```
multi_extract(  
  raster_path,  
  coord,  
  folders = NULL,  
  files = NULL,  
  layers = NULL,  
  ncores = 1  
)
```

### Arguments

raster_path	character. Path to the folder with raster folders.
coord	data.frame or matrix. Longitude and Latitude from where to extract raster values.
folders	character. folders from which to get rasters for extraction. If NULL, all folders are selected.
files	numeric. Index for raster files to be extracted from each folder. If NULL, all files are selected.
layers	numeric. Index for layers to be extracted from each raster file. If NULL, all layers are selected.
ncores	integer. Number of cores to use in parallel processing.

### Value

Data frame with extracted values from multiple rasters

### Examples

```
## Not run:  
# replace rasterpath with the directory on your computer containing worldclim data  
  
temp_pres <-  
  multi_extract(raster_path = "mydir/rasters/worldclim/global_rasters_10min/",  
               coord = TtorquatusDistribution[-1],  
               folders = c("tmax_present", "tmin_present"))  
  
## End(Not run)
```

---

sin_h	<i>Sinervo (2010) hours of activity model</i>
-------	---

---

### Description

sin\_h Simulates daily variation in temperature and counts amount of time above a temperature threshold, as seen in Sinervo et al. 2010.

### Usage

```
sin_h(tmax, tmin, thrs, res)
```

### Arguments

tmax	Raster* object. Maximum temperature raster.
tmin	Raster* object. Minimum temperature raster.
thrs	numeric. Temperature threshold in same unit as rasters.
res	numeric. time resolution in parts of hour.

### Value

numeric. Amount of time in hours above temperature threshold in simulated daily temperature variation.

### Examples

```
sin_h(28, 10, 23, 3)
```

---

transform_rasters	<i>Transform raster values using custom calls.</i>
-------------------	--

---

### Description

transform\_rasters Applies custom expressions to transform the values of spatial rasters in a stack, taking into account temporal repetition of those rasters.

### Usage

```
transform_rasters(raster_stack, separator = "_", ncores = 1, ...)
```

**Arguments**

raster\_stack    RasterStack. Stack with environmental layers.  
 separator       character. Character that separates variable names, years and scenarios.  
 ncores          integer. Number of cores to use in parallel processing.  
 ...             New rasters created.

**Value**

Returns a RasterStack with layers for the predictions required.

**Examples**

```
## Not run:
FulanusEcoRasters_present <-
  get_rasters(
    var = c('prec', 'tmax', 'tmin'),
    scenarios = 'present',
    source = "C:/Users/gabri/Dropbox/Mapinguari/global_grids_10_minutes",
    ext = FulanusDistribution[c(2,3)],
    margin = 5,
    reorder = c(1, 10, 11, 12, 2, 3, 4, 5, 6, 7, 8, 9))

# You can apply any function to subsets of rasters in the stack,
# by selecting the layers with double brackets.

transform_rasters(raster_stack = FulanusEcoRasters_present$present,
  total_1sem = sum(tmax[1:6]),
  mean_1sem = mean(tmax[1:6]),
  sd_1sem = sd(tmax[1:6]),
  total_2sem = sum(tmax[7:12]),
  mean_2sem = mean(tmax[7:12]),
  sd_2sem = sd(tmax[7:12]))

## End(Not run)
```

---

TtorquatusBreeding    *Breeding status of \*Tropidurus torquatus\* lizards at 15 locations in Brazil during each month of the year.*

---

**Description**

A dataset containing information on if *\*Tropidurus torquatus\** is breeding or not at specific locations and times.

**Usage**

TtorquatusBreeding



**Format**

A data frame with 15 rows and 14 variables:

**Lon** Longitude of occurrence records in decimal degrees

**Lat** Latitude of occurrence records in decimal degrees

**January** Binary breeding status at each location for the month of January, 1 means breeding, 0 means not breeding

**February** Binary breeding status at each location for the month of February, 1 means breeding, 0 means not breeding

**March** Binary breeding status at each location for the month of March, 1 means breeding, 0 means not breeding

**April** Binary breeding status at each location for the month of April, 1 means breeding, 0 means not breeding

**May** Binary breeding status at each location for the month of May, 1 means breeding, 0 means not breeding

**June** Binary breeding status at each location for the month of June, 1 means breeding, 0 means not breeding

**July** Binary breeding status at each location for the month of July, 1 means breeding, 0 means not breeding

**August** Binary breeding status at each location for the month of August, 1 means breeding, 0 means not breeding

**September** Binary breeding status at each location for the month of September, 1 means breeding, 0 means not breeding

**October** Binary breeding status at each location for the month of October, 1 means breeding, 0 means not breeding

**November** Binary breeding status at each location for the month of November, 1 means breeding, 0 means not breeding

**December** Binary breeding status at each location for the month of December, 1 means breeding, 0 means not breeding ...

---

TtorquatusDistribution

*359 occurrence records of \*Tropidurus torquatus\* in Brazil*

---

**Description**

A dataset containing \*Tropidurus torquatus\* distribution records

**Usage**

TtorquatusDistribution

**Format**

A data frame with 359 rows and 3 variables:

**species** species name

**Lon** Longitude of occurrence point

**Lat** Latitude of occurrence point ...

---

TtorquatusGradient      *\*Tropidurus torquatus\* body temperatures at temperature gradient experiments.*

---

**Description**

A dataset containing 3443 body temperature records of 52 *\*Tropidurus torquatus\** from 6 localities at temperature gradients.

**Usage**

TtorquatusGradient

**Format**

A data frame with 3443 rows and 3 variables:

**id** individual identity of the lizard perform

**temp** lizard body temperature at the moment of the trial, in Celsius

**site** place where lizard was collected ...

---

TtorquatusOperative      *Operative temperatures of multiple microhabitats at 6 localities in Brazil from 2014 to 2016.*

---

**Description**

A dataset containing operative temperatures of multiple microhabitats at 6 localities in Brazil from 2014 to 2016.

**Usage**

TtorquatusOperative

**Format**

A data frame with 915684 rows and 13 variables:

**site** place where temperatures were collected

**description** description of site

**Lon** Longitude of sampling point

**Lat** Latitude of sampling point

**temp** temperature at microhabitat, in degrees Celsius

**microhabitat** microhabitat sampled

**year** year of sampling

**month** month of sampling

**day** day of sampling

**hour** hour of sampling

**minute** minute of sampling

**t\_air\_max** maximum air temperature of the day at nearest weather station, in degrees Celsius ...

---

TtorquatusPerformance *Running speed achieved by 72 *Tropidurus torquatus* lizards in 274 trials under different temperatures.*

---

**Description**

A dataset containing 274 running speed trials of *Tropidurus torquatus* lizards under different temperatures, the temperatures of the runs, individual identities for each lizard, body size of each individual and the site where they were captured.

**Usage**

TtorquatusPerformance

**Format**

A data frame with 274 rows and 6 variables:

**species** species name

**id** individual identity of the lizard perform

**temp** lizard body temperature at the moment of the trial, in Celsius

**performance** maximum running speed at trial, in meters per second

**size** lizard body size, in centimeters

**site** place where lizard was collected ...

# Index

## \* datasets

- TtorquatusBreeding, 8
- TtorquatusDistribution, 9
- TtorquatusGradient, 10
- TtorquatusOperative, 10
- TtorquatusPerformance, 11

clean\_points, 2, 5

get\_predict, 3, 5

get\_rasters, 4, 5

Mapinguari, 5

multi\_extract, 5, 6

sin\_h, 5, 7

transform\_rasters, 5, 7

TtorquatusBreeding, 8

TtorquatusDistribution, 9

TtorquatusGradient, 10

TtorquatusOperative, 10

TtorquatusPerformance, 11

writeFormats, 4