

pastclim: : CHEAT SHEET



Basics

pastclim is an **R package** to easily **download and manipulate paleoclimatic data**.

INSTALLATION

Installation through devtools

```
library(devtools)
install_github("EvolEcolGroup/pastclim",
  build_vignettes = TRUE)
```

VIGNETTE

```
vignette("pastclim_overview",
  package = "pastclim")
```

Data

NATIVE DATASETS

Beyer2020:

- 22 variables
- present-120k years ago
- every 1000/2000 years

Krapp2021:

- 22 variables
- present-800k years ago
- every 1000 years

Example:

- 3 variables
- 3 time slices

Do not use for analyses!

LIST ALL AVAILABLE DATASETS

```
vignette("available_datasets",
  package = "pastclim")
```

ADD MORE DATA

Follow the instructions from the **“custom datasets” vignette**

```
vignette("custom_datasets",
  package = "pastclim")
```

Download the data

Download the full dataset

By default it will be stored within the package



```
download_dataset(dataset="Beyer2020")
```

Download only a few variables



```
download_dataset(dataset="Beyer2020",
  bio_variables = c("bio01", "bio05"))
```

You can change where the data is saved



```
set_data_path(path_to_nc = "~/climate")
```

Working with locations

Retrieve the climate for several locations scattered in time and/or space.

Format them as a data.frame with longitude, latitude, age.



```
locations <- data.frame(
  x=c(0,90,-120,-7),      # longitude
  y=c(20,45,60,37),      # latitude
  time=c(0,-10000, -20000, -10000))
```

Retrieve the climate



```
location_slice(
  x= locations[,c("x", "y")],
  time_bp = locations$time,
  bio_variables=c("bio01","bio12"),
  dataset="Example")
```

WARNING: Some locations may fall under ice or water, where the climate is unavailable. `nn_interpol = TRUE` (the default) interpolates the climate among the neighbouring cells.

Retrieve time series of the climate for each location (interpolation not available)



```
location_series(
  x= locations[,c("x", "y")],
  bio_variables=c("bio01","bio12"),
  dataset="Example")
```

Working with regions

Get climate for the world



```
region_slice(
  time_bp = -20000,
  bio_variables=c("bio01", "bio12"),
  dataset = "Example")
```

Get time series for the world



```
region_series(
  bio_variables=c("bio01", "bio12"),
  dataset = "Example")
```

df_from_region_slice() and *df_from_region_series()* will transform the outputs into dataframes

Both functions can include the arguments

- ‘crop’ to crop results based on a polygon
- ‘ext’ to crop results based on an extent

Cropping

Pre-set rectangular extents are available:



```
names(region_extent)
```

- "Africa" • "America" • "Asia"
- "Europe" • "Eurasia" • "N_America"
- "S_America" • "Oceania"

Get time series for Asia using an extent



```
region_series(
  bio_variables=c("bio01", "bio12"),
  dataset = "Example",
  ext=region_extent$Asia)
```

Pre-generated masks detailing specific continental masses



```
names(region_outline)
```

- "Africa" • "N_America" • "Eurasia"
- "Oceania" • "S_America"

Get climate from Africa using a mask



```
region_slice(
  time_bp = -20000,
  bio_variables=c("bio01", "bio12"),
  dataset = "Example",
  crop = region_outline$Africa)
```

Sampling

Sampling 100 random points from a region

```
climate_20k <- region_slice(
  time_bp = -20000,
  bio_variables = c("bio01", "bio10"),
  dataset = "Example")
```

```
sample_region_slice(
  climate_20k,
  size = 100)
```

Sampling from a time series, 100 points from the first time slice, 50 from the second.

```
climate_ts <- region_series(
  time_bp = c(-20000,-10000),
  bio_variables = c("bio10", "bio12"),
  dataset = "Example")
```

```
sample_region_series(
  climate_ts,
  size = c(100,50))
```

To sample the same amount from all time slices write only one number

Citation

Cite pastclim

```
citation("pastclim")
```

Cite database

```
help("Beyer2020")
```

Need help?



<https://rdrr.io/github/EvolEcolGroup/pastclim/man/>



<https://github.com/EvolEcolGroup/pastclim>



<https://evolecolgroup.github.io/pastclim/>



am315@cam.ac.uk (Andrea Manica)

