Package 'shoredate'

October 24, 2023

Type Package

Title Shoreline Dating Coastal Stone Age Sites

Version 1.1.1

Description Provides tools for shoreline dating coastal Stone Age sites. The
implemented method was developed in Roalkvam (2023)
<doi:10.1016/j.quascirev.2022.107880> for the Norwegian Skagerrak coast.
Although it can be extended to other areas, this also forms the core area for
application of the package. Shoreline dating is based on the present-day
elevation of a site, a reconstruction of past relative sea-level change, and
empirically derived estimates of the likely elevation of the sites above the
contemporaneous sea-level when they were in use. The geographical and temporal
coverage of the method thus follows from the availability of local geological
reconstructions of shoreline displacement and the degree to which the
settlements to be dated have been located on or close to the shoreline when
they were in use. Methods for numerical treatment and visualisation of the
dates are provided, along with basic tools for visualising and evaluating the
location of sites.

Language en-US

License GPL (>= 3)

URL https://github.com/isakro/shoredate,
https://isakro.github.io/shoredate/

BugReports https://github.com/isakro/shoredate/issues

Encoding UTF-8

RoxygenNote 7.2.1

Imports ggplot2, ggrepel, ggridges, ggspatial, sf, terra, utils

Suggests covr, elevatr, knitr, rmarkdown, testthat (>= 3.0.0), vdiffr,
rnaturalearth

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Author Isak Roalkvam [aut, cre] (<<https://orcid.org/0000-0001-6974-1374>>)

Maintainer Isak Roalkvam <isakroa@protonmail.com>

Depends R (>= 3.5.0)

Repository CRAN

Date/Publication 2023-10-24 16:10:03 UTC

R topics documented:

---

create_isobases	*Create isobases with different directions within the spatial coverage in south-eastern Norway*

Description

Function to create isobases for interpolating shoreline displacement curves within the spatial coverage in south-eastern Norway. This is done from the centre points of the supplied displacement curves. Isobases can also be created for multiple directions, which is useful for testing the sensitivity of dates to the direction of the isobases.

Usage

```
create_isobases(isobase_direction)
```

Arguments

isobase_direction
                A numerical vector holding a single or multiple directions for the isobases.

Value

A simple feature holding the isobases represented as lines.

Examples

```
# Create isobases in a specified direction
isobases <- create_isobases(327)
plot(sf::st_geometry(isobases))

# Create isobases using different directions
isobases <- create_isobases(c(327, 338))

# Plot for visualisation
plot(sf::st_geometry(isobases))
```

---

displacement_plot   *Plot shoreline displacement curves*

Description

Function for plotting shoreline displacement curves. Calling to plot without providing a target curve will display the four underlying geologically derived displacement curves.

Usage

```
displacement_plot(
  target_curve = NA,
  displacement_curves = c("Horten", "Porsgrunn", "Tvedestrand", "Arendal"),
  target_name = "Target curve",
  target_line = "solid",
  target_col = "red",
  target_alpha = 1,
 displacement_line = c(Horten = "solid", Porsgrunn = "solid", Tvedestrand = "solid",
    Arendal = "solid"),
 displacement_col = c(Horten = "darkorange", Porsgrunn = "darkgreen", Tvedestrand =
    "blue", Arendal = "black"),
  displacement_alpha = 1,
  greyscale = FALSE
)
```

Arguments

target_curve   Data frame holding a shoreline displacement curve.

displacement_curves

Character vector specifying which geologically informed displacement curves should be plotted. Accepted values are c("Horten", "Porsgrunn", "Tvedestrand", "Arendal"). All are included by default.

target_name   Character value specifying the name that is given to the target curve, if provided. Defaults to "Target curve".

target_line   Character value specifying the line type that is used for the target curve, if this is provided. Defaults to "solid".

target_col   Character value specifying the colour that is used for the target curve, if this is
                  provided. Defaults to "red".

target_alpha   Numerical value specifying the alpha value that is used for the target curve, if
                  this is provided. Defaults to 1.

displacement_line

                  Character vector specifying the line types that are used for the geological dis-
                  placement curves to be plotted. Defaults to c("Horten" = "solid", "Porsgrunn"
                  = "solid", "Tvedestrand" = "solid", "Arendal" = "solid").

displacement_col

                  Character vector specifying the colours that are used for the geological displace-
                  ment curves to be plotted. Defaults to c("Horten" = "darkorange", "Porsgrunn"
                  = "darkgreen", "Tvedestrand" = "blue", "Arendal" = "black").

displacement_alpha

                  Numerical value specifying the alpha value that are used for all of the geological
                  displacement curves to be plotted. Defaults to 1.

greyscale   Logical value indicating whether the plot should be in greyscale or not. Defaults
                  to FALSE.

Value

A plot displaying the underlying shoreline displacement curves and, if provided, a target curve.

Examples

```
# Empty plot for speed
displacement_plot(displacement_curves = "")
```

---

interpolate_curve   *Interpolate displacement curve to a target location within the spatial
                         coverage in south-eastern Norway*

Description

Interpolate the trajectory of past shoreline displacement to a target location within the spatial cov-
erage on the Skagerrak coast of south-eastern Norway. This based on the distance of the location
to the shoreline isobases of the geologically derived displacement curves and is done using inverse
distance weighting.

Usage

```
interpolate_curve(
  target,
  isobases = NA,
  power = 2,
  cal_reso = 10,
  verbose = FALSE
)
```

Arguments

`target`	A spatial target location to where the new displacement curve is interpolated.
`isobases`	4 spatial lines representing the shoreline isobases of the existing displacement curves. Multiple sets of 4 isobases with different isobase directions can be provided (see `create_isobases()`). Defaults to isobases with a direction of 327.
`power`	A numerical value indicating the inverse distance power for IDW. Defaults to 2.
`cal_reso`	A numerical value specifying the resolution to use on the calendar scale. Defaults to 10.
`verbose`	Logical value indicating whether progress should be printed to console. Defaults to FALSE.

Value

Returns a list holding an interpolated displacement curve for each isobase direction. Each displacement curve is represented by a data frame with the columns bce where negative values indicate years BCE and positive CE, lowerelev, representing the lower limit for the elevation of the shoreline for each year. upperelev, the upper limit for elevation of the shoreline for each year, and direction which indicates the direction of the isobases used when interpolating the curve.

Examples

```
# Create example point using the required coordinate system
# WGS84 / zone UTM32N (EPSG: 32632)
target_point <- sf::st_sfc(sf::st_point(c(579570, 6582982)), crs = 32632)

# Interpolate shoreline displacement curve to the target point location,
# setting the resolution on the calendar scale to 2000 years for speed.
target_curve <- interpolate_curve(target_point, cal_reso = 2000)
```

---

print.shoreline_date   *Print shoreline dates to console*

Description

Print the dates held in a shoreline_date object. Each date is printed with site name, elevation and highest density region. If the isobase direction is different or more are provided than the default, the directions and dates associated with these are printed separately.

Usage

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

Arguments

x	Object of class shoreline_date.
...	Additional arguments.

Value

Print the site names, elevations, non-default isobase directions and HDRs contained in a `shoreline_date` object to console.

Examples

```
target_point <- sf::st_sfc(sf::st_point(c(538310, 6544255)), crs = 32632)

# Reduce date resolution with cal_reso and elevation_reso for speed
target_date <- shoreline_date(site = target_point,
                              elevation = 70,
                              elev_reso = 1,
                              cal_reso = 400)

# Print to console
target_date
```

---

shoredate_hdr   *Find the highest density region of shoreline dates*

Description

Function to find 95% highest density region (HDR) for a provided shoreline date. Negative values denote years BCE while positive values denote CE.

Usage

```
shoredate_hdr(bce, probability, site_name, cal_reso, prob = 0.95)
```

Arguments

`bce`	A vector holding calendar years associated with a date.
`probability`	A vector holding the probability mass corresponding to each calendar year.
`site_name`	A vector holding the name of the site that has been dated.
`cal_reso`	Resolution on the calendar scale used when dating the site.
`prob`	A numerical value between 0 and 1 indicating the probability coverage of the HDR. Defaults to `0.95`.

Value

A list holding start and end points for segments of the highest density region of a shoreline date, the weighted mean date, the probability coverage and site name.

Examples

```
target_point <- sf::st_sfc(sf::st_point(c(538310, 6544255)), crs = 32632)

# Reduce date resolution with cal_reso and elevation_reso for speed
target_date <- shoreline_date(sites = target_point,
                              elevation = 80,
                              elev_reso = 1,
                              cal_reso = 400)

# shoredate_hdr() is already called under the hood with shoreline_date(),
# the result of which is printed when calling the shoreline_date object
target_date

# However, shoredate_hdr() can be applied separately by pulling the
# necessary data from the date
(shoredate_hdr(target_date[[1]][[1]]$date$bce,
               target_date[[1]][[1]]$date$probability,
               target_date[[1]][[1]]$site_name,
               target_date[[1]][[1]]$cal_reso))
```

---

shoredate_plot   *Plot shoreline dates*

Description

Function for plotting shoreline dates along with associated metadata.

Usage

```
shoredate_plot(
  shorelinedates,
  date_probability = TRUE,
  date_probability_scale = 10000,
  elevation_distribution = TRUE,
  displacement_curve = TRUE,
  site_name = FALSE,
  parameters = FALSE,
  isobase_direction = FALSE,
  highest_density_region = TRUE,
  hdr_label = TRUE,
  multiplot = FALSE,
  date_col = NA,
  date_fill = "darkgrey",
  displacement_col = "red",
  displacement_fill = "red",
  site_elevation_col = "#046c9a",
  site_elevation_fill = "#046c9a",
  hdr_col = "black",
```

```
    hdr_label_xadj = 0.2,
    hdr_label_yadj = 0.3,
    greyscale = FALSE
)
```

Arguments

shorelinedates   Object of class `shoreline_date`.

date_probability

          Logical value indicating whether the probability distribution of the shoreline date should be plotted. Defaults to `TRUE`.

date_probability_scale

          Numerical constant with which to scale the probability distribution of the date to make it fit the plot. Defaults to 10000.

elevation_distribution

          Logical value indicating whether the distribution describing the distance between site and shoreline should be displayed. Default is `TRUE`.

displacement_curve

          Logical value indicating whether the displacement curve should be displayed. Default is `TRUE`.

site_name   Logical value indicating whether the name of the site should be printed in the header of the plot. Defaults to `FALSE`.

parameters   Logical value indicating whether the parameters of the statistical function should be displayed. Default is `FALSE`.

isobase_direction

          Logical value indicating whether the direction of the isobases should be printed. Default is `FALSE`.

highest_density_region

          Logical value indicating whether the 95% highest density region should be displayed. Defaults to `TRUE`.

hdr_label   Logical value indicating whether the numeric values for the highest density regions should be displayed. Default is `TRUE`.

multiplot   Logical value indicating whether multiple dates should be plotted individually, or be collapsed into a single plot. The only other graphical option with `multiplot` set to `TRUE` is `highest_density_region`. Default is `FALSE`.

date_col   Character value specifying the outline colour of the probability distribution of the shoreline date. Defaults to `NA`.

date_fill   Character value specifying the fill colour of the probability distribution of the shoreline date. Defaults to `"darkgrey"`.

displacement_col

          Character value specifying the outline colour of the displacement curve. Defaults to `"red"`.

displacement_fill

          Character value specifying the fill colour of the displacement curve. Defaults to `"red"`.

site_elevation_col

        Character value specifying the outline colour of the distribution describing the likely distance between site and shoreline. Defaults to ″#046c9a″.

site_elevation_fill

        Character value specifying the fill colour of the distribution describing the likely distance between site and shoreline. Defaults to ″#046c9a″.

hdr_col   Character value specifying the colour of the line segment giving the highest density region of the shoreline date. Defaults to ″black″.

hdr_label_xadj   Numerical value between 0 and 1 specifying the position of the HDR label on the x-axis. Increasing the value moves the label further from the plot border. Defaults to 0.2.

hdr_label_yadj   Numerical value between 0 and 1 specifying the position of the HDR label on the y-axis. Increasing the value moves the label further from the plot border. Defaults to 0.3.

greyscale   Logical value indicating whether the plot should be in greyscale or not. If TRUE, overrides other colour parameters. Defaults to FALSE.

Details

shoredate_plot() returns a plot displaying the provided shoreline dates. A single plot is created for each date, where a range of settings can be adjusted to display or hide various parameters and results. Setting the parameter multiplot to TRUE returns a sparser version for multiple dates, where the only option is whether or not to display the highest density region in addition to each date. multiplot does not allow for multiple isobase directions. Negative values denote years BCE while positive values denote CE.

Value

Plot(s) displaying shoreline dates and associated metadata.

Examples

```
# Create example point with correct coordinate reference system
target_point <- sf::st_sfc(sf::st_point(c(538310, 6544255)), crs = 32632)

# Reduce date resolution with cal_reso and elevation_reso for speed
target_date <- shoreline_date(sites = target_point, elevation = 80,
                              elev_reso = 10,
                              cal_reso = 500)

shoredate_plot(target_date)
```

---

shoredate_sumplot   *Plot the summed probability distribution of multiple shoreline dates*

Description

Function to plot the sum of the probabilities of multiple shoreline dates as resulting from running sum_shoredates().

Usage

```
shoredate_sumplot(shoredates_sum, sample_size = TRUE)
```

Arguments

shoredates_sum	Object of class shoredates_sum.
sample_size	Logical indicating whether or not to display the number of summed dates on the plot. Defaults to TRUE.

Value

A line plot showing the provided summed probability distribution.

Examples

```
# Create example points
target_points <- sf::st_sfc(sf::st_point(c(538310, 6544255)),
                            sf::st_point(c(538300, 6544250)))

# Set correct CRS
target_points <- sf::st_as_sf(target_points, crs = 32632)

# Reduce date resolution with cal_reso and elevation_reso for speed
target_dates <- shoreline_date(target_points,
                               elevation = c(65, 70),
                               elev_reso = 10,
                               cal_reso = 750)

# Find summed probability
target_sum <- sum_shoredates(target_dates)

# Call to plot
shoredate_sumplot(target_sum)
```

---

shoreline_date *Shoreline date*

Description

A function for shoreline dating Stone Age sites based on their present-day elevation, their likely elevation above sea-level when in use and the trajectory of past shoreline displacement. Details and caveats pertaining to the implemented method is given in Roalkvam (2023).

Usage

```
shoreline_date(
  sites,
  elevation = NA,
  elev_reso = 0.01,
  cal_reso = 10,
  isobase_direction = 327,
  sum_isobase_directions = FALSE,
  model = "gamma",
  model_parameters = c(0.286, 20.833),
  elev_fun = "mean",
  upper_temp_limit = -2500,
  target_curve = NA,
  hdr_prob = 0.95,
  normalise = TRUE,
  sparse = FALSE,
  verbose = FALSE
)
```

Arguments

sites	Vector giving one or more site names, or, if displacement curves are to be interpolated, objects of class sf representing the sites to be dated. In the case of a spatial geometry, the first column is taken as the site name.
elevation	Vector of numeric elevation values for each site or a an elevation raster of class SpatRaster from the package terra from where the elevation values are to be derived.
elev_reso	Numeric value specifying the resolution with which to step through the distribution representing the distance between site and shoreline. Defaults to 0.01m.
cal_reso	Numeric value specifying the resolution to use on the calendar scale. Defaults to 10.
isobase_direction	
	A vector of numeric values defining the direction(s) of the isobases. Defaults to 327.

sum_isobase_directions

        Logical value indicating that if multiple isobase directions are specified in `isobase_direction`
the results should be summed for each site using `sum_shoredates`. Defaults to
FALSE.

model   Character vector specifying the statistical model with which to model the dis-
        tance from site to shoreline. Currently accepts either "none" or "gamma". De-
faults to "gamma".

model_parameters

        Vector of numeric values specifying the parameters for the statistical model de-
scribing the distance between site and shoreline. Defaults to c(0.286, 20.833),
denoting the shape and scale of the default gamma function, respectively.

elev_fun   Statistic to define site elevation if this is to be derived from an elevation raster.
        Uses `terra::extract()`. Defaults to mean.

upper_temp_limit

        Numerical value giving the upper temporal limit. Dates with a start date after
the limit are returned as NA. Defaults to -2500, i.e. 2500 BCE.

target_curve   Data frame holding pre-computed shoreline displacement curve. This has to
        have the same or higher resolution on the calendar scale as that specified with
`cal_reso`. [interpolate_curve()](#) will be run if nothing is provided to `target_curve`.
Defaults to NA.

hdr_prob   Numeric value specifying the coverage of the highest density region. Defaults
        to 0.95.

normalise   Logical value specifying whether the shoreline date should be normalised to
        sum to unity. Defaults to TRUE.

sparse   Logical value specifying if only site name and shoreline date should be returned.
        Defaults to FALSE. Note that of the functions for further treatment, sparse dates
are only compatible with [sum_shoredates()](#).

verbose   Logical value indicating whether progress should be printed to console. Defaults
        to FALSE.

Value

A nested list of class `shoreline_date` holding the shoreline date results and associated metadata
for each dated site for each isobase direction. The elements of each date is:

- `site_name` name of the site.
- `site_elev` elevation of the site.
- `date` data frame with the columns `bce` where negative values indicate years BCE and positive
  CE, as well as `probability`, which gives the probability mass for each year.
- `weighted_mean` the weighted mean date.
- `hdr_start` start values for the HDR ranges.
- `hdr_end` end values for the HDR ranges.
- `hdr_prob` probability level for the HDR.
- `dispcurve` data frame holding the displacement curve used for dating the site. This has the
  columns `bce`, giving years BCE/CE. `lowerelev`, the lower limit for the elevation of the shore-
  line for each year. `upperelev`, the upper limit for elevation of the shoreline for each year.

- dispcurve_direction direction of the isobases in use.

- model_parameters parameters for the statistical model.

- modeldat data frame holding the model distribution. The column offset denotes the vertical distance (m) from the shoreline, as specified by the elev_reso argument. px is the cumulative probability at each step of offset, and probs is the probability of each step found by subtracting the preceding value from each value of px.

- cal_reso resolution on the calendar scale.

References

Roalkvam, I. 2023. A simulation-based assessment of the relation between Stone Age sites and relative sea-level change along the Norwegian Skagerrak coast. *Quaternary Science Reviews* 299:107880. DOI: https://doi.org/10.1016/j.quascirev.2022.107880

Examples

```
# Create example point using the required CRS WGS84 UTM32N (EPSG: 32632)
target_point <- sf::st_sfc(sf::st_point(c(538310, 6544255)), crs = 32632)

# Date target point, manually specifying the elevation instead of providing
# an elevation raster. Reducing elev_reso and cal_reso for speed.
shoreline_date(sites = target_point,
               elevation = 80,
               elev_reso = 1,
               cal_reso = 400)
```

---

sum_shoredates   *Sum the probability of multiple shoreline dates*

Description

Function for finding the summed probability distribution of multiple shoreline dates.

Usage

```
sum_shoredates(
  shoreline_dates,
  cut_off = -2500,
  cut_off_level = 1,
  normalise = TRUE
)
```

Arguments

shoreline_dates
    Object of class `shoreline_date`.

cut_off   Calendar year specifying where dates should be cut off. Defaults to 2500 BCE.

cut_off_level Numerical value between 0 and 1 indicating the probability mass that has to fall after the cut-off for a date to be excluded. Defaults to 1, retaining all dates.

normalise   Logical value indicating whether the probability sum of the dates should be normalised to sum to unity. Defaults to TRUE.

Value

List of class `shoredate_sum` holding the elements:

- `sum` data frame with the columns bce where negative values indicate years BCE and positive CE, as well as `probability`, which gives the probability mass for each year.

- `dates_n` number of dates that make up the sum after applying any specified cut-off. One date per site per isobase direction.

Examples

```
target_points <- sf::st_sfc(sf::st_point(c(538310, 6544255)),
                            sf::st_point(c(538300, 6544250)))
target_points <- sf::st_as_sf(target_points, crs = 32632)

# Shoreline date, reducing resoltuion on elevation and calendar scales for
# speed.
target_dates <- shoreline_date(target_points,
                               elevation = c(65, 70),
                               elev_reso = 10,
                               cal_reso = 500)

sum_shoredates(target_dates)
```

---

target_plot   *Plot a map with target locations*

Description

Function to plot the centroids of sites to be dated and shoreline isobases of employed displacement curves on a basemap. Defaults to displaying a light-weight version of the spatial coverage in south-eastern Norway. However, spatial geometries covering other regions can also be provided or temporarily downloaded with the function.

Usage

```
target_plot(
  targets = NA,
  isobases = sf::st_read(system.file("extdata/isobases.gpkg", package = "shoredate",
    mustWork = TRUE), quiet = TRUE),
  basemap = sf::st_read(system.file("extdata/naturalearth_basemap.gpkg", package =
    "shoredate", mustWork = TRUE), quiet = TRUE),
  crs_epsg = 32632,
  naturalearth_basemap = FALSE,
  naturalearth_zoom = c(20000, 20000),
  target_labels = TRUE,
  scalebar = TRUE,
  scalebar_width = 0.4,
  scalebar_style = "ticks",
  scalebar_location = "br",
  base_fill = "grey",
  base_col = NA,
  target_shape = 21,
  target_col = "black",
  target_fill = "red",
  target_size = 2.25,
  isobase_line = c(Horten = "solid", Porsgrunn = "solid", Tvedestrand = "solid", Arendal
    = "solid"),
  isobase_col = c(Arendal = "black", Porsgrunn = "darkgreen", Tvedestrand = "blue",
    Horten = "darkorange"),
  greyscale = FALSE
)
```

Arguments

targets	Objects of class sf representing the sites to be dated. The first column beyond geom is taken as site name.
isobases	Spatial lines as object of class sf representing the shoreline isobases. Defaults to isobases with a direction of 327 within the spatial limit in SE Norway, but create_isobases() can be used to create isobases with other directions that can then be passed to target_plot().
basemap	Object of class sf representing a background map. Defaults to a light-weight basemap for the spatial limit in SE Norway.
crs_epsg	Numeric value specifying the EPSG code of the coordinate reference system (CRS) to be used. Geometries with a different CRS will be re-projected. Defaults to 32632, which is WGS 84 / UTM zone 32N (EPSG:32632).
naturalearth_basemap	
	Logical value specifying if a background map should be downloaded to be used as a basemap. Downloaded files are stored with base::tempdir() and deleted when the R session is closed. If TRUE, overrides the basemap argument. Defaults to FALSE.

naturalearth_zoom

> A vector of two numerical values specifying the amount of cropping that is done around provided `targets` when `naturalearth_basemap` is set to `TRUE`. Be aware of whether a projected or geographical CRS is specified in `crs_epsg`. Defaults to `c(20000, 20000)`.

target_labels   Logical value specifying whether the targets should be labelled in the plot. Takes the first column beyond the one holding the geometries to represent names. If this is not present the targets are labelled by row number. Defaults to `TRUE`.

scalebar   Logical specifying whether a scale bar should be added to the plot. Defaults to `TRUE`.

scalebar_width   Numerical value specifying the width of the scale bar by passing it to the `width_hint` argument of `ggspatial::annotation:scale()`. Defaults to `0.4`.

scalebar_style   Character value specifying the style of the scale bar by passing it to the `style` argument of `ggspatial::annotation:scale()`. Defaults to `"ticks"`.

scalebar_location

> Character value specifying the location of the scale bar on the plot by passing it to the `location` argument of `ggspatial::annotation:scale()`. Defaults to `"br"`.

base_fill   Character value specifying the fill colour of the basemap. Defaults to `"grey"`.

base_col   Character value specifying the outline colour of the basemap. Defaults to `NA`.

target_shape   Numerical value specifying the point shape that represent the centroids of the targets. Defaults to `21`.

target_col   Character value specifying the colour parameter for the points that represent the centroids of the targets. Defaults to `"black"`.

target_fill   Character value specifying the fill parameter for the points that represent the centroids of the targets. Defaults to `"red"`.

target_size   Numerical value specifying the size of the points that represent the centroids of the targets. Defaults to `2.25`.

isobase_line   Vector of character values specifying the linetype that is used to represent the isobases of the geologically derived displacement curves. Defaults to `c("Horten" = "solid", "Porsgrunn" = "solid", "Tvedestrand" = "solid", "Arendal" = "solid")`.'

isobase_col   Vector of character values specifying the colours used for the lines that represent the isobases of the geologically derived displacement curves. Defaults to `c("Arendal" = "black", "Porsgrunn" = "darkgreen", "Tvedestrand" = "blue", "Horten" = "darkorange")`.'

greyscale   Logical value indicating whether the plot should include colours or not. Overrides other graphical parameters When set to `TRUE`. Defaults to `FALSE`.

Value

A ggplot that displays a background map with the location of the shoreline isobases within the spatial coverage in south-eastern Norway, unless geometries for other regions are provided. If provided, the function also plots the position of target locations represented as centroids.

Examples

```
# Display the background map and default isobases
target_plot()
```

Index