Package 'mapme.biodiversity'

Title Efficient Monitoring of Global Biodiversity Portfolios

Version 0.0.1

Description Biodiversity areas, especially primary forest, serve a
multitude of functions for local economy, regional functionality of
the ecosystems as well as the global health of our planet. Recently,
adverse changes in human land use practices and climatic responses to
increased greenhouse gas emissions, put these biodiversity areas under
a variety of different threats. The present package helps to analyse a
number of biodiversity indicators based on freely available
geographical datasets. It supports computational efficient routines
that allow the analysis of potentially global biodiversity portfolios.
The primary use case of the package is to support evidence based
reporting of an organization's effort to protect biodiversity areas
under threat and to identify regions were intervention is most duly
needed.

License GPL (>= 3)

Depends R (>= 3.5.0)

Imports curl, data.table, dplyr, httr, magrittr, pbapply, rvest, sf,
stringr, terra, tibble, tidyr, tidyselect

Suggests DiagrammeR, exactextractr, ggplot2, knitr, lwgeom, rmarkdown,
SPEI, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

RoxygenNote 7.1.2

SystemRequirements GDAL (>= 3.0.0), PROJ (>= 4.8.0)

NeedsCompilation no

Author Darius A. Görgen [aut, cre],
Om Prakash Bhandari [aut]

Maintainer Darius A. Görgen <darius2402@web.de>

Repository CRAN

Date/Publication 2022-04-19 08:42:30 UTC

R **topics documented:**

| accessibility | *Calculate accessibility statistics* |

Description

Accessibility is the ease with which larger cities can be reached from a certain location. This function allows to efficiently calculate accessibility statistics (i.e. travel time to nearby major cities) for polygons. For each polygon, the desired statistic/s (mean, median or sd) is/are returned. The required resources for this indicator are:

- traveltime

Format

A tibble with a column for accessibility statistics (in minutes)

Details

The following arguments can be set:

stats_accessibility Function to be applied to compute statistics for polygons either one or multiple inputs as character. Supported statistics are: "mean", "median", "sd", "min", "max", "sum" "var".

engine The preferred processing functions from either one of "zonal", "extract" or "exactextract" as character.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2022,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources("traveltime",
    range_traveltime = c("5k_10k", "100k_200k", "500k_1mio", "1mio_5mio")
  ) %>%
 calc_indicators("accessibility", stats_accessibility = c("min", "max"), engine = "extract") %>%
 tidyr::unnest(accessibility))
```

`available_indicators` *Backlog function for available indicators*

Description

This function returns a list of either all available indicators and some additional metadata or for one or more requested indicators. It can be used by users of the package to inform themselves about the available data sets and to learn about potentially additional arguments that should be specified when requesting the indicator.

Usage

```
available_indicators(indicator = NULL)
```

Arguments

indicator   Defaults to NULL meaning that a list with all available indicators will be returned. If a character vector is specified only the information about the requested indicator will be returned.

Value

A list object.

Examples

```
names(available_indicators())
```

`available_resources` *Backlog function for available resources*

Description

This function returns a list of either all available resources and some additional metadata or for one or more requested resources. It can be used by users of the package to inform themselves about the available data sets and to learn about potentially additional arguments that should be specified when requesting the resource.

Usage

```
available_resources(resources = NULL)
```

Arguments

resources   Defaults to NULL meaning that a list with all available resources will be returned. If a character vector is specified only the information about the requested resource will be returned.

Value

A list object.

Examples

```
names(available_resources())
```

---

biome   *Calculate biomes statistics (TEOW) based on WWF*

Description

This function allows to efficiently retrieve the name of the biomes and compute the corresponding area from Terrestrial Ecoregions of the World (TEOW) - World Wildlife Fund (WWF) for polygons. For each polygon, the name and area of the biomes (in hectare) is returned. The required resources for this indicator are:

- ecoregions

Format

A tibble with a column for name of the biomes and corresponding area (in ha).

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2001,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources("ecoregions") %>%
  calc_indicators("biome") %>%
  tidyr::unnest(biome))
```

---

calc_indicators *Compute specific indicators*

Description

With `calc_indicators()` specific biodiversity indicators can be calculated. A requirment is that the ressources that are mandatory inputs for the requested indicators are available locally. Multiple indicators and their respective additional arguments can be supplied. You can check available indicators and their requirement via `available_indicators()`, but the function will also gracefully inform you about any misspecifications.

Usage

```
calc_indicators(x, indicators, ...)
```

Arguments

x	A biodiversity portfolio object constructed via `init_portfolio()`
indicators	A character vector indicating the requested indicators. All specified indicators must be supported by the package. You can use `available_indicators()` to get more information, e.g. additional required arguments and their default values, about the supported indicators
...	Additional arguments required for the requested indicators. Check `available_indicators()` to learn more about the supported indicators and their arguments.

Value

The sf portfolio object x with additional nested list column per requested indicator.

---

chirps *Climate Hazards Group InfraRed Precipitation with Station data (CHIRPS)*

Description

This resource is published by Funk et al. (2015) and represents a quasi-global (50°S-50°S) rainfall estimation at a monthly resolution starting with the year 1981 to the near-present. It has a spatial resolution of 0.05°. The data can be used to retrieve information on the amount of rainfall. Due to the availability of +30 years, anomaly detection and long-term average analysis is also possible. The routine will download the complete archive in order to support long-term average and anomaly calculations with respect to the 1981 - 2010 climate normal period. Thus no additionall arguments need to be specified.

Format

Global raster layers available for years 1981 to near-present.

Source

https://data.chc.ucsb.edu/products/CHIRPS-2.0/global_monthly/cogs/

References

Funk, C., Peterson, P., Landsfeld, M. et al. The climate hazards infrared precipitation with stations—a new environmental record for monitoring extremes. Sci Data 2, 150066 (2015). doi: 10.1038/sdata.2015.66

---

chirpsprec *Calculate precipitation statistics based on CHIRPS*

Description

This functions allows to calculate precipitation statistics based on the CHIRPS rainfall estimates. Corresponding to the time-frame of the analysis of the portfolio, monthly precipitation statistics are calculated. These include the total rainfall amount, rainfall anomaly against the 1981-2010 climate normal, and the Standardized Precipitation Index (SPI) which is available for scales between 1 and 48 months. Th function needs the SPEI package to be installed. The required resources for this indicator are:

- chirps

Format

A tibble with a column for years, months, absolute rainfall (in mm), rainfall anomaly (in mm) and one or more columns per selected time-scale for SPI (dimensionless).

Details

The following arguments can be set:

scales_spi An integer vector indicating the scales for which to calculate the SPI.

spi_previous_year An integer specifying how many previous years to include in order to fit the SPI. Defaults to 8 years.

engine The preferred processing functions from either one of "zonal", "extract" or "exactextract" as character.

Examples

```
if(Sys.getenv("NOT_CRAN") == "true"){
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2010,
    outdir = system.file("res", package = "mapme.biodiversity"),
```

```
      tmpdir = system.file("tmp", package = "mapme.biodiversity"),
      cores = 1,
      verbose = FALSE
  ) %>%
  get_resources("chirps") %>%
  calc_indicators("chirpsprec", engine = "exactextract", scales_spi = 3, spi_prev_years = 8) %>%
  tidyr::unnest(chirpsprec))
}
```

---

drought_indicator   *Calculate drought indicator statistics*

Description

This function allows to efficiently calculate the relative wetness in the shallow groundwater section with regard to the the 1948-2012 reference period. The values represent the wetness percentile a given area achieves at a given point in time in regard to the reference period. For each polygon, the desired statistic/s (mean, median or sd) is/are returned. The required resources for this indicator are:

- nasagrace

Format

A tibble with a column for each specified stats and a column with the respective date.

Details

The following arguments can be set:

stats_drought Function to be applied to compute statistics for polygons either one or multiple inputs as character "mean", "median" or "sd".

engine The preferred processing functions from either one of "zonal", "extract" or "exactextract" as character.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2022,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
```

```
  get_resources("nasagrace") %>%
 calc_indicators("drought_indicator", stats_drought = c("mean", "median"), engine = "extract") %>%
 tidyr::unnest(drought_indicator))
```

---

ecoregions   *Terrestrial Ecoregions of the World (TEOW) Polygon*

Description

This resource is part of the publication by Olson et al. (2004) "Terrestrial Ecosystems of the World
(TEOW) from WWF-US (Olson)". It depicts 867 terrestrial ecoregions around the world classified
into 14 different terrestrial biomes such as forests, grasslands, or deserts. The polygons represent
the ecoregions, defined as relatively large units of land or inland water sharing a large majority of
biodiversity. The datasets is made available from World Wildlife Fund (WWF) for the year 2001.

Format

Global terrestrial polygon resource depicting ecoregions.

Source

https://www.worldwildlife.org/publications/terrestrial-ecoregions-of-the-world

References

Olson, D. M., Dinerstein, E., Wikramanayake, E. D., Burgess, N. D., Powell, G. V. N., Underwood,
E. C., D'Amico, J. A., Itoua, I., Strand, H. E., Morrison, J. C., Loucks, C. J., Allnutt, T. F., Ricketts,
T. H., Kura, Y., Lamoreux, J. F., Wettengel, W. W., Hedao, P., Kassem, K. R. 2001. Terrestrial
ecoregions of the world: a new map of life on Earth. Bioscience 51(11):933-938.

---

elevation   *Calculate elevation statistics*

Description

This function allows to efficiently calculate elevation statistics for polygons. For each polygon, the
desired statistic/s (mean, median or sd) is/are returned. The required resources for this indicator
are:

- srtmdem

Format

A tibble with a column for elevation statistics (in meters)

Details

The following arguments can be set:

stats_elevation Function to be applied to compute statistics for polygons either one or multiple inputs as character. Supported statistics are: "mean", "median", "sd", "min", "max", "sum" "var".

engine The preferred processing functions from either one of "zonal", "extract" or "exactextract" as character.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2000:2020,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources("srtmdem") %>%
  calc_indicators("elevation",
    stats_elevation = c("mean", "median", "sd", "var"), engine = "extract"
  ) %>%
  tidyr::unnest(elevation))
```

---

emissions   *Calculate emission statistics*

Description

This functions allows to efficiently calculate emission statistics for areas of interest. For each year in the analysis timeframe, the forest losses from Hansen et al. (2013) are overlayed with the respective emission layer from Harris et al. (2021) and area-wise emission statistics are calculated for each year. The required resources for this indicator are:

- treecover2000
- lossyear
- greenhouse.

Format

A tibble with a column for years and emissions (in Mg)

Details

The following arguments can be set:

min_size The minum size of a forest patch to be considered as forest in ha.

min_cover The minimum cover percentage per pixel to be considered as forest.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2016:2017,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources(
    resources = c("treecover2000", "lossyear", "greenhouse"),
    vers_treecover = "GFC-2020-v1.8", vers_lossyear = "GFC-2020-v1.8"
  ) %>%
  calc_indicators("emissions", min_size = 1, min_cover = 30) %>%
  tidyr::unnest(emissions))
```

---

esalandcover   *ESA Copernicus Global Land Cover layer*

Description

This 100 meter spatial resolution land cover resource is published by Buchhorn et al. (2020) "Copernicus Global Land Cover Layers—Collection 2". The resource represents the actual surface cover of ground available annually for the period 2015 to 2019. The cell values range from 0 to 200, representing total of 23 discrete classifications from ESA.

Format

A global tiled raster resource available for years 2015 to 2019.

Source

<https://lcviewer.vito.be/download>

References

© European Union, Copernicus Land Monitoring Service (year), European Environment Agency (EEA)", f.ex. in 2018: "© European Union, Copernicus Land Monitoring Service 2018, European Environment Agency (EEA)

---

get_resources   *Download specific biodiversity resources*

Description

With get_resources() specific biodiversity data sets required for the calculation of indicators can be downloaded. The function supports the specification of several indicators and their respective additional arguments. You can check the required arguments via available_resources(), but the function will gracefully inform you about any misspecifications.

Usage

```
get_resources(x, resources, ...)
```

Arguments

x	A biodiversity portfolio object constructed via init_portfolio()
resources	A character vector indicating the requested resources. All specified resources must be supported by the package. You can use available_resources() to get more information, e.g. additional required arguments and their default values, about the supported resources.
...	Additional arguments required for the requested resources. Check available_resources() to learn more about the supported resources and their arguments.

Value

Primarily called for the side effect of downloading resources. Returns the sf portfolio object x with its attributes amended by the requested resources.

---

gmw   *Calculate mangrove extent based on Global Mangrove Watch (GMW)*

Description

This function allows to efficiently calculate area of mangrove from Global Mangrove Watch - World Conservation Monitoring Centre (WCMC) for polygons. For each polygon, the area of the mangrove (in hectare) for desired year is returned. The required resources for this indicator are:

- mangrove

Format

A tibble with a column for area of mangrove (in ha) and corresponding year.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "shell_beach_protected_area_41057_B.gpkg",
  package = "mapme.biodiversity"
) %>%
  read_sf() %>%
  init_portfolio(
    years = c(1996, 2016),
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources("mangrove") %>%
  calc_indicators("gmw") %>%
  tidyr::unnest(gmw))
```

---

greenhouse   *Forest greenhouse gas emissions*

Description

This resource is part of the publication by Harris et al. (2021) "Global maps of twenty-first century forest carbon fluxes.". It represents "the greenhouse gas emissions arising from stand-replacing forest disturbances that occurred in each modeled year (megagrams CO2 emissions/ha, between 2001 and 2020). Emissions include all relevant ecosystem carbon pools (aboveground biomass, belowground biomass, dead wood, litter, soil) and greenhouse gases (CO2, CH4, N2O)." The area unit that is downloaded here corresponds to the "megagrams of CO2 emissions/pixel" layer, in order to support the calculation of area-wise emissions.

Format

A global tiled raster resource available for all land areas.

Details

There are no arguments users need to specify. However, users should note that the spatial extent for this dataset does not totally cover the same extent as the `treecover2000` and `lossyear` resources by Hansen et al. (2013). A missing value (NA) will be inserted for greenhouse gas emissions for areas where no data is available.

Source

https://data.globalforestwatch.org/datasets/gfw::forest-greenhouse-gas-emissions/about

References

Harris, N.L., Gibbs, D.A., Baccini, A. et al. Global maps of twenty-first century forest carbon fluxes. Nat. Clim. Chang. 11, 234–240 (2021). https://doi.org/10.1038/s41558-020-00976-6

---

init_portfolio	*Initialization of a biodiversity portfolio object*

Description

This function expects an `sf` object as its first argument that contains only geometry of type `POLYGON` or `MULTIPOLYGON`. Each row of the object is considered a single asset in the portfolio for which biodiversity indicators will be calculated further down the processing chain. Some preliminary checks are conducted, e.g. that the CRS of the object is EPSG:4326 otherwise it will be transformed. Some portfolio wide parameters such as the output directory for downloaded data sets, a temporal directory for intermediate calculation of the number of cores available for the indicator calculation can be set by the user to have more fine-control of the workflow. However, these parameters are also set to sensible defaults and thus can be omitted during portfolio initialization.

Usage

```
init_portfolio(
  x,
  years,
  outdir = getwd(),
  tmpdir = tempdir(),
  cores = parallel::detectCores() - 1,
  add_resources = TRUE,
  aria_bin = NULL,
  verbose = TRUE
)
```

Arguments

x	The sf object to be transformed to a portfolio
years	Numeric vector for time frame of the analysis handed over as a vector of consecutive years
outdir	Character indicating the directory where resources will be written to. If the directory does not exist, we will attempt to create it. The output director cannot be equal to the temporary directory. Defaults to the current working directory.
tmpdir	Character vector to a directory used for intermediate files. Defaults to the output of `tempdir()`, e.g. the current R session's temporal directory. If a custom file path does not exist, we will attempt to create it. Cannot be equal to the output directory. Note, that for raster calculations we will set the temporal directory for the `terra` package here. Please make sure that enough disk space is available because some intermediate calculations can become quite large.

cores	An integer value indicating the number of cores on the host machine available for indicator calculation. It defaults to `parallel::detectCores() -1` cores, i.e. one core less than all available cores.
add_resources	Logical if existing resources in 'outdir' should be added to the portfolio. Defaults to TRUE. Setting it to FALSE can be of use e.g. if a previous download has terminated unexpectedly in order to resume.
aria_bin	A character vector to an aria2c executable for parallel downloads
verbose	Logical, defaults to TRUE, indicating if progress information is printed.

Value

The sf portfolio object x with amended attributes controlling the processing behavior further down the processing chain.

---

landcover *Calculate area of different landcover classes*

Description

The land cover data shows us how much of the region is covered by forests, rivers, wetlands, barren land, or urban infrastructure thus allowing the observation of land cover dynamics over a period of time. This function allows to efficiently calculate area of different landcover classes for polygons. For each polygon, the area of the classes in hectare(ha) is returned. The required resources for this indicator are:

- esalandcover

Format

A tibble with a column for area (in ha) per landcover class

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2016:2017,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources("esalandcover") %>%
  calc_indicators("landcover") %>%
  tidyr::unnest(landcover))
```

---

lossyear	*Year of forest loss occurence*

Description

This resource is part of the publication by Hansen et al. (2013) "High-Resolution Global Maps of 21st-Century Forest Cover Change". It represents "Forest loss during the period 2000–2020, defined as a stand-replacement disturbance, or a change from a forest to non-forest state. Encoded as either 0 (no loss) or else a value in the range 1–20, representing loss detected primarily in the year 2001–2020, respectively." Due to changes in the satellites products used in the compilation of the tree loss product, results before the year 2011 and afterwards are not directly comparable until reprocessing has finished. Users should be aware of this limitation, especially when the timeframe of the analysis spans over the two periods delimited by the year 2011.

Format

A global tiled raster resource available for all land areas.

Details

The following argument can be set:

vers_lossyear The version of the dataset to download. Defaults to "GFC-2020-v1.8". Check `mapme.biodiversity:::.available_gfw_versions()` to get a list of available versions

Source

https://data.globalforestwatch.org/documents/tree-cover-loss/explore

References

Hansen, M. C., P. V. Potapov, R. Moore, M. Hancher, S. A. Turubanova, A. Tyukavina, D. Thau, S. V. Stehman, S. J. Goetz, T. R. Loveland, A. Kommareddy, A. Egorov, L. Chini, C. O. Justice, and J. R. G. Townshend. 2013. "High-Resolution Global Maps of 21st-Century Forest Cover Change." Science 342 (15 November): 850–53.

---

mangrove	*Global Mangrove Extent Polygon*

Description

This resource is part of the publication by Bunting et al. (2018) "The Global Mangrove Watch—A New 2010 Global Baseline of Mangrove Extent". The polygons represent the mangrove, which is tropical coastal vegetation and considered the most significant part of the marine ecosystem. This resource is available for the period 1996- 2016 from Global Mangrove Watch (GMW), providing geospatial information about global mangrove extent.

Format

Global mangrove extent polygon available for years 1996, 2007-2010, 2015, and 2016.

Source

https://data.unep-wcmc.org/datasets/45

References

Bunting P., Rosenqvist A., Lucas R., Rebelo L-M., Hilarides L., Thomas N., Hardy A., Itoh T., Shimada M. and Finlayson C.M. (2018). The Global Mangrove Watch – a New 2010 Global Baseline of Mangrove Extent. Remote Sensing 10(10): 1669. doi:10.3390/rs10101669.

---

nasagrace	*NASA GRACE-based Drought Indicator layer*

Description

The resource is published by NASA GRACE Tellus. This data set reflects on potential drought conditions in the shallow groundwater section relative to a reference period spanning from 1948 to 2012. It is available as a global raster with a weekly temporal resolution starting with the year 2003. The value indicates the wetness percentile of a given pixel with regard to the reference period.

Format

Global raster layers available for years 2003 to present.

Source

https://nasagrace.unl.edu/globaldata/

---

popcount	*Calculate population count statistics*

Description

WorldPop, which was initiated in 2013, offers easy access to spatial demographic datasets, claiming to use peer-reviewed and fully transparent methods to create global mosaics for the years 2000 to 2020. This function allows to efficiently calculate population count statistics (e.g. total number of population) for polygons. For each polygon, the desired statistic/s (min, max, sum, mean, median, sd or var) is/are returned. The required resources for this indicator are:

- worldpop

Format

A tibble with a column for population count statistics

Details

The following arguments can be set:

stats_popcount Function to be applied to compute statistics for polygons either one or multiple
inputs as character. Supported statistics are: "mean", "median", "sd", "min", "max", "sum"
"var".

engine The preferred processing functions from either one of "zonal", "extract" or "exactextract"
as character.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2000:2010,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources("worldpop") %>%
 calc_indicators("popcount", stats_popcount = c("sum", "median"), engine = "extract") %>%
 tidyr::unnest(popcount))
```

---

read_portfolio   *Reading a portfolio object from disk*

Description

This function can be used to read a portfolio object that was previously written to disk via `write_portfolio()`
back into R as an `sf` object. It should specifically be directed against a GeoPackage which was the
output of `write_portfolio()`, otherwise the function is very likely to fail. All available indicators
will be read back into R as nested list columns reflecting the output once `calc_indicators()` has
been called.

Usage

```
read_portfolio(file, ...)
```

Arguments

file	A character vector pointing to a GeoPackage that has been previously written to disk via `write_portfolio()`
...	Additional arguments supplied to `st_read()`

Details

Important Note Portfolio-wide attributes that were specified via `init_portfolio()` will not be reconstructed. The reason is that users most likely exported to a GeoPackage in order to share their data, thus the file is very likely to be opened on a different machine / in a different working directory. Users can simply apply `init_portfolio()` on the object to re-set these attributes.

Value

An sf object object with nested list columns for every indicator table found in the GeoPackage source file.

---

soilgrids   *Soildgrids data layers*

Description

Soilgrids is a project combining global observation data with machine learning to map the spatial distribution of soil properties across the globe. It is produced at a spatial resolution of 250 meters and each parameters is mapped at different depths. In order to be able to assess prediction uncertainty, besides the mean and median prediction, the 0.05 and 0.95 percentile predictions are available. The following parameters are available:

bdod Bulk density of the fine earth fraction (kg/dm3)

cec Cation Exchange Capacity of the soil (cmol(c)/kg)

cfvo Volumetric fraction of coarse fragments > 2 mm (cm3/100cm3 (volPerc))

clay Proportion of clay particles < 0.002 mm in the fine earth fraction (g/100g)

nitrogen Total nitrogen (g/kg)

phh2o Soil pH (pH)

sand Proportion of sand particles > 0.05 mm in the fine earth fraction (g/100g)

silt Proportion of silt particles >= 0.002 mm and <= 0.05 mm in the fine earth fraction (g/100g)

soc Soil organic carbon content in the fine earth fraction (g/kg)

ocd Organic carbon density (kg/m3)

ocs Organic carbon stocks (kg/m²)

Format

A global tiled raster resource available for all land areas.

Details

Users can specify the following arguments:

layer The soil parameter as a single charachter

depth The requested depth as a single chrachter

stat The predicted statistic as a single charachter

Except for ocs, which is only available for a depth of "0-30cm", all other parameters are available at the following depths:

- "0-5cm"
- "5-15cm"
- "15-30cm"
- "30-60cm"
- "60-100cm"
- "100-200cm"

Each parameter and depth is available for the following statistics:

- "Q0.05"
- "Q0.50"
- "mean"
- "Q0.95"

Source

https://www.isric.org/explore/soilgrids

References

Hengl T, Mendes de Jesus J, Heuvelink GBM, Ruiperez Gonzalez M, Kilibarda M, et al. (2017) SoilGrids250m: Global gridded soil information based on machine learning. PLOS ONE 12(2): e0169748. doi: 10.1371/journal.pone.0169748

---

soilproperties   *Calculate Zonal Soil Properties*

Description

This indicator allows the extraction of zonal statistics for resource layers previously downloaded from soilgrids, thus in total supporting the calculation of zonal statistics for 10 different soil properties at 6 different depths for a total of 4 different model outputs (stat). Zonal statistics will be calculated for all soilgrid layers that have been previously made available vie get_resources(). The required resource for this indicator is:

- soilgrids

Format

A tibble with a column for the soilgrid layer, the depth and the model output statistic as well as additionall columns for all zonal statistics specified via `stats_soil`

Details

The following arguments can be set:

stats_soil Function to be applied to compute statistics for polygons either single or multiple inputs as character. Supported statistics are: "mean", "median", "sd", "min", "max", "sum" "var".

engine The preferred processing functions from either one of "zonal", "extract" or "exactextract" as character.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2022,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources("soilgrids",
    layers = c("clay", "silt"), depths = c("0-5cm", "5-15cm"), stats = "mean"
  ) %>%
 calc_indicators("soilproperties", stats_soil = c("mean", "median"), engine = "extract") %>%
 tidyr::unnest(soilproperties))
```

---

srtmdem   *SRTM 30m Digital Elevation Model (DEM) layer*

Description

This resource is published by Farr et al. (2007) "The Shuttle Radar Topography Mission". The layer represents the 30m global terrestrial digital elevation model from the NASA Shuttle Radar Topographic Mission (SRTM), available for download as 5 degree x 5 degree tiles. It is encoded as meter, representing the elevation at the particular grid cell.

Format

A global tiled raster resource available for all land areas.

Source

<https://srtm.csi.cgiar.org/srtmdata/>

References

NASA Shuttle Radar Topography Mission (SRTM)(2013). Shuttle Radar Topography Mission (SRTM) Global. Distributed by OpenTopography. https://doi.org/10.5069/G9445JDF. Accessed: 2022-03-17

---

teow	*Calculate terrestrial ecoregions statistics (TEOW) based on WWF*

Description

This function allows to efficiently retrieve the name of the ecoregions and compute the corresponding area from Terrestrial Ecoregions of the World (TEOW) - World Wildlife Fund (WWF) for polygons. For each polygon, the name and area of the ecoregions (in hectare) is returned. The required resources for this indicator are:

- ecoregions

Format

A tibble with a column for name of the ecoregions and corresponding area (in ha).

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2001,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources("ecoregions") %>%
  calc_indicators("teow") %>%
  tidyr::unnest(teow))
```

traveltime	*Accessibility to Cities layer*

Description

This resource is published by Weiss et al. (2018) "A global map of travel time to cities to assess inequalities in accessibility in 2015" on journal nature. Accessibility is the ease with which larger cities can be reached from a certain location. This resource represents the travel time to major cities in the year 2015. Encoded as minutes, representing the time needed to reach that particular cell from nearby city of target population range. The following ranges to nearby cities are available:

- "5k_10k"
- "10k_20k"
- "20k_50k"
- "50k_100k"
- "100k_200k"
- "200k_500k"
- "500k_1mio"
- "1mio_5mio"
- "50k_50mio"
- "5k_110mio"
- "20k_110mio"

Format

Global raster layer available for year 2015.

Details

The following argument should be specified by users:

range_traveltime A character vector indicating one or more ranges to download.

Source

https://figshare.com/articles/dataset/Travel_time_to_cities_and_ports_in_the_year_2015/7638134/3

References

Weiss, D. J., Nelson, A., Gibson, H. S., Temperley, W., Peedell, S., Lieber, A., . . . & Gething, P. W. (2018). A global map of travel time to cities to assess inequalities in accessibility in 2015. Nature, 553(7688), 333-336.

---

treecover   *Calculate treecover statistics*

Description

This functions allows to efficiently calculate treecover statistics for polygons. For each year in the analysis timeframe, the forest losses in preceeding and the current years are substracted from the treecover in the year 2000 and actual treecover figures within the polygon are returned. The required resources for this indicator are:

- treecover2000
- lossyear

Format

A tibble with a column for years and treecover (in ha)

Details

The following arguments can be set:

min_size The minum size of a forest patch to be considered as forest in ha.

min_cover The minimum cover percentage per pixel to be considered as forest.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2016:2017,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources(
    resources = c("treecover2000", "lossyear"),
    vers_treecover = "GFC-2020-v1.8", vers_lossyear = "GFC-2020-v1.8"
  ) %>%
  calc_indicators("treecover", min_size = 1, min_cover = 30) %>%
  tidyr::unnest(treecover))
```

---

treecover2000 *Treecover for the year 2000*

Description

This resource is part of the publication by Hansen et al. (2013) "High-Resolution Global Maps of 21st-Century Forest Cover Change". It represents "tree cover in the year 2000, defined as canopy closure for all vegetation taller than 5m in height. Encoded as a percentage per output grid cell, in the range 0–100." Due to changes in the satellites products used in the compilation of the treecover product, results before the year 2011 and afterwards are not directly comparable until reprocessing has finished. Users should be aware of this limitation, especially when the timeframe of the analysis spans over the two periods delimited by the year 2011.

Format

A global tiled raster resource available for all land areas.

Details

The following argument can be set:

vers_treecover The version of the dataset to download. Defaults to "GFC-2020-v1.8". Check mapme.biodiversity:::.available_gfw_versions() to get a list of available versions

Source

https://data.globalforestwatch.org/documents/tree-cover-2000/explore

References

Hansen, M. C., P. V. Potapov, R. Moore, M. Hancher, S. A. Turubanova, A. Tyukavina, D. Thau, S. V. Stehman, S. J. Goetz, T. R. Loveland, A. Kommareddy, A. Egorov, L. Chini, C. O. Justice, and J. R. G. Townshend. 2013. "High-Resolution Global Maps of 21st-Century Forest Cover Change." Science 342 (15 November): 850–53.

---

treeloss *Calculate treeloss statistics*

Description

This functions allows to efficiently calculate the treecover and emissions indicators in a single function call together. Since most of the pre-processing operations for treecover and emissions are the same, it is more efficient to calculate them in one run if users are actually interested in both statistics. Otherwise users are advised to use the respective single indicator functions. The required resources for this indicator are:

- treecover2000

- [lossyear](#)
- [greenhouse](#).

Format

A tibble with a column for years, treecover (in ha), and emissions (in Mg CO2)

Details

The following arguments can be set:

min_size The minum size of a forest patch to be considered as forest in ha.

min_cover The minimum cover percentage per pixel to be considered as forest.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2016:2017,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources(
    resources = c("treecover2000", "lossyear", "greenhouse"),
    vers_treecover = "GFC-2020-v1.8", vers_lossyear = "GFC-2020-v1.8"
  ) %>%
  calc_indicators("treeloss", min_size = 1, min_cover = 30) %>%
  tidyr::unnest(treeloss))
```

---

tri   *Calculate Terrain Ruggedness Index (TRI) statistics*

Description

Terrain Ruggedness Index is a measurement developed by Riley, et al. (1999). The elevation difference between the center pixel and its eight immediate pixels are squared and then averaged and its square root is taken to get the TRI value. This function allows to efficiently calculate terrain ruggedness index (tri) statistics for polygons. For each polygon, the desired statistic/s (mean, median or sd) is/are returned. The required resources for this indicator are:

- [srtmdem](#)

Format

A tibble with a column for terrain ruggedness index statistics (in meters). The range of index values and corresponding meaning: (1) 0 - 80 m :- level surface (2) 81-116 m :- nearly level surface (3) 117-161 m :- slightly rugged surface (4) 162-239 m :- intermediately rugged surface (5) 240-497 m :- moderately rugged surface (6) 498-958 m :- highly rugged surface (7) 959-4367 m:- extremely rugged surface

Details

The following arguments can be set:

stats_tri Function to be applied to compute statistics for polygons either single or multiple inputs as character. Supported statistics are: "mean", "median", "sd", "min", "max", "sum" "var".

engine The preferred processing functions from either one of "zonal", "extract" or "exactextract" as character.

References

Riley, S. J., DeGloria, S. D., & Elliot, R. (1999). Index that quantifies topographic heterogeneity. intermountain Journal of sciences, 5(1-4), 23-27.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2000:2020,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources("srtmdem") %>%
 calc_indicators("tri", stats_tri = c("mean", "median", "sd", "var"), engine = "extract") %>%
  tidyr::unnest(tri))
```

---

wcprec *Calculate precipitation statistics*

Description

This function allows to efficiently calculate precipitation statistics from Worldclim for polygons. For each polygon, the desired statistic/s (min, max, sum, mean, median, sd or var) is/are returned. The required resources for this indicator are:

- precipitation layer from worldclim

Format

A tibble with a column for precipitation statistics (in mm)

Details

The following arguments can be set:

stats_worldclim Function to be applied to compute statistics for polygons either one or multiple inputs as character. Supported statistics are: "mean", "median", "sd", "min", "max", "sum" "var".

engine The preferred processing functions from either one of "zonal", "extract" or "exactextract" as character.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2018,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources("precipitation") %>%
 calc_indicators("wcprec", stats_worldclim = c("mean", "median"), engine = "extract") %>%
  tidyr::unnest(wcprec))
```

---

wctmax   *Calculate maximum temperature statistics*

Description

This function allows to efficiently calculate maximum temperature statistics from Worldclim for polygons. For each polygon, the desired statistic/s (min, max, sum, mean, median, sd or var) is/are returned. The required resources for this indicator are:

  • maximum temperature layer from worldclim

Format

A tibble with a column for maximum temperature statistics (in °C)

Details

The following arguments can be set:

stats_worldclim Function to be applied to compute statistics for polygons either one or multiple inputs as character. Supported statistics are: "mean", "median", "sd", "min", "max", "sum" "var".

engine The preferred processing functions from either one of "zonal", "extract" or "exactextract" as character.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2018,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources("maxtemperature") %>%
 calc_indicators("wctmax", stats_worldclim = c("mean", "median"), engine = "extract") %>%
  tidyr::unnest(wctmax))
```

---

wctmin   *Calculate minimum temperature statistics based on WorldClim*

Description

This function allows to efficiently calculate minimum temperature statistics from Worldclim for polygons. For each polygon, the desired statistic/s (min, max, sum, mean, median, sd or var) is/are returned. The required resources for this indicator are:

- minimum temperature layer from [worldclim]

Format

A tibble with a column for minimum temperature statistics (in °C)

Details

The following arguments can be set:

stats_worldclim Function to be applied to compute statistics for polygons either one or multiple inputs as character. Supported statistics are: "mean", "median", "sd", "min", "max", "sum" "var".

engine The preferred processing functions from either one of "zonal", "extract" or "exactextract" as character.

Examples

```
library(sf)
library(mapme.biodiversity)
(aoi <- system.file("extdata", "sierra_de_neiba_478140_2.gpkg", package = "mapme.biodiversity") %>%
  read_sf() %>%
  init_portfolio(
    years = 2018,
    outdir = system.file("res", package = "mapme.biodiversity"),
    tmpdir = system.file("tmp", package = "mapme.biodiversity"),
    add_resources = FALSE,
    cores = 1,
    verbose = FALSE
  ) %>%
  get_resources("mintemperature") %>%
 calc_indicators("wctmin", stats_worldclim = c("mean", "median"), engine = "extract") %>%
 tidyr::unnest(wctmin))
```

---

worldclim   *WorldClim climatic variables (min temperature, max temperature, mean precipitation)*

Description

This resource is published by Fick et al. (2017) "WorldClim 2: new 1-km spatial resolution climate surfaces for global land areas". This resource represents multiple climatic variables from which we will be requiring minimum temperature, maximum temperature, and mean precipitation layers. The layers are available to download for the period 2000 - 2018 on monthly basis from WorldClim.

Format

Global raster layers available for years 2000 to 2018.

Details

Enlisted different resources can be reqested with their dedicated functions:

tmin Encoded as (°C), representing the minimum temperature per output grid cell.

tmax Encoded as (°C), representing the maximum temperature per output grid cell.

prec Encoded as (mm), representing the mean precipitation per output grid cell.

Source

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

---

worldpop *Population Count layer for year 2000-2020*

Description

This resource is published by open spatial demographic data and research organization called WorldPop. This resource represents the population count, 1 km spatial resolution layers available to download from the year 2000 to 2020. The dataset is called as WorldPop Unconstrained Global Mosaics. The encoded cell value represents the total number of people in that particular grid cell.

Format

Global raster layers available for years 2000-2020.

Source

<https://www.worldpop.org/geodata/listing?id=64>

---

write_portfolio *Writing a portfolio to disk*

Description

The function is used to save a processes biodiversity portfolio to disk. In order to ensure interoperability with other geospatial software the only supported format is the GeoPackage. If any other format is chosen, the function will automatically replace the supplied file extension with '.gpkg'. The metadata of a portfolio together with the geometry will be written to a table called 'metadata'. All available and supported indicators, which are expected to be present as a nested list columns will be written to their own respective tables. In order to allow re-joining the metadata with the indicators, it is expected that a column called '.assetid' which uniquely identifies all assets is present. Usually, users do not have to take care of this since the usual {mapme.biodiversity} workflow will ensure that this columns is present. Additional arguments to st_write() can be supplied.

Usage

```
write_portfolio(x, dsn, overwrite = FALSE, ...)
```

Arguments

x	A portfolio object processed with {mapme.biodiversity}
dsn	A file path for the output file. Should end with '.gpkg'
overwrite	A logical indicating if the output file should be overwritten if it exists
...	Additional arguments supplied to st_write()

Value

x, invisibly

Index

32