Package ‘climate’

March 30, 2020

Title Interface to Download Meteorological (and Hydrological) Datasets

Version 0.9.3

Description Automatize downloading of meteorological and hydrological data from publicly available repositories:
OGIMET (<http://ogimet.com/index.phtml.en>),
University of Wyoming -
  atmospheric vertical profiling data (<http://weather.uwyo.edu/upperair>),
and Polish Institute of Meteorology and Water Management -
This package also allows for adding geographical coordinates for each observation.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Depends R (>= 3.1)

Imports RCurl, XML

Suggests testthat, knitr, rmarkdown, dplyr, tidyr, maps

URL https://github.com/bczernecki/climate

BugReports https://github.com/bczernecki/climate/issues

VignetteBuilder knitr

NeedsCompilation no

Author Bartosz Czernecki [aut, cre] (<https://orcid.org/0000-0001-6496-1386>),
  Arkadiusz Głogowski [aut] (<https://orcid.org/0000-0002-7587-8892>),
  Jakub Nowosad [aut] (<https://orcid.org/0000-0002-1057-3721>),
  IMGW-PIB [ctb] (source of the data)

Maintainer Bartosz Czernecki <nwp@amu.edu.pl>

Repository CRAN

Date/Publication 2020-03-30 08:50:09 UTC
### R topics documented:

- hydro_imgw .......................................................... 2
- hydro_imgw_annual ............................................... 3
- hydro_imgw_daily .................................................. 4
- hydro_imgw_monthly ............................................... 5
- imgw_hydro_abbrev ............................................... 6
- imgw_hydro_stations ............................................. 6
- imgw_meteo_abbrev ............................................... 7
- imgw_meteo_stations ............................................. 8
- meteo_imgw ........................................................ 8
- meteo_imgw_daily .................................................. 9
- meteo_imgw_hourly ............................................... 10
- meteo_imgw_monthly ............................................. 11
- meteo_noaa_co2 .................................................. 12
- meteo_ogimet ..................................................... 13
- nearest_stations_ogimet ...................................... 15
- sounding_wyoming ............................................... 16
- stations_ogimet .................................................. 18

---

**Index**

---

**hydro_imgw**

*Hydrological data from IMGW*

---

### Description

Downloading hourly, daily, and monthly hydrological data from the measurement stations available in the danepubliczne.imgw.pl collection

### Usage

```r
hydro_imgw(
  interval,
  year,
  coords = FALSE,
  value = "H",
  station = NULL,
  col_names = "short",
  ...
)
```

### Arguments

- **interval**: temporal resolution of the data ("daily", "monthly", or "semiannual_and_annual")
- **year**: vector of years (e.g., 1966:2000)
- **coords**: add coordinates of the stations (logical value TRUE or FALSE)
**hydro_imgw_annual**

- **value**: type of data (can be: state - "H" (default), flow - "Q", or temperature - "T")
- **station**: vector of hydrological stations danepubliczne.imgw.pl; can be given as station name with CAPITAL LETTERS (character) It accepts either names (characters in CAPITAL LETTERS) or stations' IDs (numeric)
- **col_names**: three types of column names possible: "short" - default, values with shorten names, "full" - full English description, "polish" - original names in the dataset
- **...**: other parameters that may be passed to the 'shortening' function that shortens column names

**Value**

A data.frame with columns describing the hydrological parameters (e.g. flow, water level) where each row represent a measurement, depending on the interval, at a given hour, month or year. If coords = TRUE additional two columns with geographic coordinates are added.

**Examples**

```r
x <- hydro_imgw("monthly", year = 1999)
head(x)
```

---

**hydro_imgw_annual**  
*Semi-annual and annual hydrological data*

---

**Description**

Downloading hydrological data for the semi-annual and annual period available in the danepubliczne.imgw.pl collection

**Usage**

```r
hydro_imgw_annual(
  year,
  coords = FALSE,
  value = "H",
  station = NULL,
  col_names = "short",
  ...
)
```

**Arguments**

- **year**: vector of years (e.g., 1966:2000)
- **coords**: add coordinates of the stations (logical value TRUE or FALSE)
- **value**: type of data (can be: state - "H" (default), flow - "Q", or temperature - "T")
hydro_imgw_daily

station  name or ID of hydrological station(s). It accepts names (characters in CAPITAL LETTERS) or stations' IDs (numeric)
col_names three types of column names possible: "short" - default, values with shorten names, "full" - full English description, "polish" - original names in the dataset

Examples

```r
ty = hydro_imgw_annual(year = 2000, value = "H", station = "ANNOPOL")
head(ty)
```

Description

Downloading daily hydrological data from the danepubliczne.imgw.pl collection

Usage

```r
hydro_imgw_daily(
  year,
  coords = FALSE,
  station = NULL,
  col_names = "short",
  ...
)
```

Arguments

- **year**  vector of years (e.g., 1966:2000)
- **coords**  add coordinates of the stations (logical value TRUE or FALSE)
- **station**  name or ID of hydrological station(s). It accepts names (characters in CAPITAL LETTERS) or stations' IDs (numeric)
- **col_names** three types of column names possible: "short" - default, values with shorten names, "full" - full English description, "polish" - original names in the dataset
- ... other parameters that may be passed to the 'shortening’ function that shortens column names
**Examples**

```r
daily <- hydro_imgw_daily(year = 2000)
head(daily)
```

---

**hydro_imgw_monthly**  
*Monthly hydrological data*

**Description**

Downloading monthly hydrological data from the danepubliczne.imgw.pl collection

**Usage**

```r
hydro_imgw_monthly(
  year,
  coords = FALSE,
  station = NULL,
  col_names = "short",
  ...
)
```

**Arguments**

- `year` vector of years (e.g., 1966:2000)
- `coords` add coordinates of the stations (logical value TRUE or FALSE)
- `station` name or ID of hydrological station(s). It accepts names (characters in CAPITAL LETTERS) or stations' IDs (numeric)
- `col_names` three types of column names possible: "short" - default, values with shorten names, "full" - full English description, "polish" - original names in the dataset
- `...` other parameters that may be passed to the 'shortening' function that shortens column names

**Examples**

```r
monthly <- hydro_imgw_monthly(year = 2000)
head(monthly)
```
Definitions of hydrological parameters used for shortening column names from the danepubliczne.imgw.pl collection

Description
The object contains 3 columns that are currently used for improving readability of the downloaded dataset: fullname, abbr_eng, and fullname_eng

Usage
imgw_hydro_abbrev

Format
The data contains a data.frame with ca. 20 elements described in three ways:
- fullname original column names as downloaded from the repository
- abbr_eng shorten column names with abbreviations derived from the most popular scheme used for meteorological parameters
- fullname_eng detailed description of downloaded meteorological variables

Examples
```r
data(imgw_hydro_abbrev)
head(imgw_hydro_abbrev)
```

Location of the hydrological stations from the danepubliczne.imgw.pl collection

Description
The object contains weather stations coordinates, ID numbers, and elevations

Usage
imgw_hydro_stations
Format

The data contains a data.frame with 1304 obs. of 3 variables:

- id Station ID
- X Longitude
- Y Latitude

The object is in the geographic coordinates using WGS84 (EPSG:4326).

Examples

data(imgw_hydro_stations)
head(imgw_hydro_stations)

imgw_meteo_abbrev

*Definitions of meteorological parameters used for shortening column names for the meteorological data from the danepubliczne.imgw.pl collection*

Description

The object contains 3 columns that are currently used for improving readability of the downloaded dataset: fullname, abbr_eng, and fullname_eng

Usage

imgw_meteo_abbrev

Format

The data contains a data.frame with ca. 250 elements described in three ways:

- fullname original column names as downloaded from the repository
- abbr_eng shorten column names with abbreviations derived from the most popular scheme used for meteorological parameters
- fullname_eng detailed description of downloaded meteorological variables

The object is created mostly to be used altogether with the meteo_shortening_imgw function

Examples

data(imgw_meteo_abbrev)
head(imgw_meteo_abbrev)
### meteo_imgw

**Description**

Downloading hourly, daily, and monthly meteorological data from the SYNOP / CLIMATE / PRECIP stations available in the danepubliczne.imgw.pl collection

**Usage**

```r
meteo_imgw(
  interval,
  rank,
  year,
  status = FALSE,
  coords = FALSE,
  station = NULL,
  col_names = "short",
  ...
)
```
Arguments

- **interval**: temporal resolution of the data ("hourly", "daily", "monthly")
- **rank**: rank of the stations ("synop", "climate", "precip")
- **year**: vector of years (e.g., 1966:2000)
- **status**: leave the columns with measurement and observation statuses (default status = FALSE - i.e. the status columns are deleted)
- **coords**: add coordinates of the station (logical value TRUE or FALSE)
- **station**: vector of hydrological stations danepubliczne.imgw.pl can be name of station CAPITAL LETTERS(character) It accepts names (characters in CAPITAL LETTERS) or stations’ IDs (numeric)
- **col_names**: three types of column names possible: "short" - default, values with shorten names, "full" - full English description, "polish" - original names in the dataset
- **...**: other parameters that may be passed to the 'shortening’ function that shortens column names

Value

A data.frame with columns describing the meteorological parameters (e.g. temperature, wind speed, precipitation) where each row represent a measurement, depending on the interval, at a given hour, month or year. If coords = TRUE additional two columns with geographic coordinates are added.

Examples

```r
x <- meteo_imgw("monthly", rank = "synop", year = 2018, coords = TRUE)
head(x)
```

---

**meteo_imgw_daily**

*Daily meteorological data*

Description

Downloading daily (meteorological) data from the SYNOP / CLIMATE / PRECIP stations available in the danepubliczne.imgw.pl collection

Usage

```r
meteo_imgw_daily(
  rank,
  year,
  status = FALSE,
  coords = FALSE,
  station = NULL,
  col_names = "short",
  ...
)
```
**Argument**

- **rank**
  - rank of the stations ("synop", "climate", or "precip")
- **year**
  - vector of years (e.g., 1966:2000)
- **status**
  - leave the columns with measurement and observation statuses (default status = FALSE - i.e. the status columns are deleted)
- **coords**
  - add coordinates of the station (logical value TRUE or FALSE)
- **station**
  - name of meteorological station(s). It accepts names (characters in CAPITAL LETTERS); stations’ IDs (numeric) are no longer valid
- **col_names**
  - three types of column names possible: "short" - default, values with shorten names, "full" - full English description, "polish" - original names in the dataset
  - other parameters that may be passed to the 'shortening' function that shortens column names

**Examples**

```r
daily <- meteo_imgw_daily(rank = "climate", year = 2000)
head(daily)
```

---

**Description**

Downloading hourly (meteorological) data from the SYNOP / CLIMATE / PRECIP stations available in the danepubliczne.imgw.pl collection

**Usage**

```r
meteo_imgw_hourly(
  rank,
  year,
  status = FALSE,
  coords = FALSE,
  station = NULL,
  col_names = "short",
  ...
)
```
**meteo_imgw_monthly**

**Arguments**

- **rank**
  - rank of the stations ("synop", "climate", or "precip")

- **year**
  - vector of years (e.g., 1966:2000)

- **status**
  - leave the columns with measurement and observation statuses (default status = FALSE - i.e. the status columns are deleted)

- **coords**
  - add coordinates of the station (logical value TRUE or FALSE)

- **station**
  - name or ID of meteorological station(s). It accepts names (characters in CAPITAL LETTERS) or stations' IDs (numeric)

- **col_names**
  - three types of column names possible: "short" - default, values with shorten names, "full" - full English description, "polish" - original names in the dataset

**Examples**

```r
hourly <- meteo_imgw_hourly(rank = "climate", year = 1984)
head(hourly)
```

---

**Description**

Downloading monthly (meteorological) data from the SYNOP / CLIMATE / PRECIP stations available in the danepubliczne.imgw.pl collection

**Usage**

```r
meteo_imgw_monthly(
  rank,
  year,
  status = FALSE,
  coords = FALSE,
  station = NULL,
  col_names = "short",
  ...
)
```
Arguments

- rank: rank of the stations ("synop", "climate", or "precip")
- year: vector of years (e.g., 1966:2000)
- status: leave the columns with measurement and observation statuses (default status = FALSE - i.e. the status columns are deleted)
- coords: add coordinates of the station (logical value TRUE or FALSE)
- station: name or ID of meteorological station(s). It accepts names (characters in CAPITAL LETTERS) or stations’ IDs (numeric)
- col_names: three types of column names possible: "short" - default, values with shorten names, "full" - full English description, "polish" - original names in the dataset
- ...: other parameters that may be passed to the 'shortening' function that shortens column names

Examples

```r
monthly <- meteo_imgw_monthly(rank = "climate", year = 1969)
head(monthly)

# a descriptive (long) column names:
monthly2 <- meteo_imgw_monthly(rank = "synop", year = 2018,
                          col_names = "full")
head(monthly2)

# please note that station names may change over time
# and thus 2 names are required in some cases:
df = meteo_imgw_monthly(rank = 'synop', year = 1991:2000,
                           coords = TRUE, station = c("POZNAŃ","POZNAŃ-ŁAWICA"))
```

---

**meteo_noaa_co2**

**CO2 Mauna Loa (NOAA) dataset**

Description

Carbon Dioxide (CO2) monthly measurements from Mauna Loa observatory. The source file is available at: ftp://aftp.cmdl.noaa.gov/products/trends/co2/co2_mm_mlo.txt with all further details.

Usage

```r
meteo_noaa_co2()
```
Details

Data from March 1958 through April 1974 have been obtained by C. David Keeling of the Scripps Institution of Oceanography (SIO) and were obtained from the Scripps website (scrippsco2.ucsd.edu).

The "average" column contains the monthly mean CO2 mole fraction determined from daily averages. The mole fraction of CO2, expressed as parts per million (ppm) is the number of molecules of CO2 in every one million molecules of dried air (water vapor removed). If there are missing days concentrated either early or late in the month, the monthly mean is corrected to the middle of the month using the average seasonal cycle. Missing months are denoted by -99.99. The "interpolated" column includes average values from the preceding column and interpolated values where data are missing. Interpolated values are computed in two steps. First, we compute for each month the average seasonal cycle in a 7-year window around each monthly value. In this way the seasonal cycle is allowed to change slowly over time. We then determine the "trend" value for each month by removing the seasonal cycle; this result is shown in the "trend" column. Trend values are linearly interpolated for missing months. The interpolated monthly mean is then the sum of the average seasonal cycle value and the trend value for the missing month. NOTE: In general, the data presented for the last year are subject to change, depending on recalibration of the reference gas mixtures used, and other quality control procedures. Occasionally, earlier years may also be changed for the same reasons. Usually these changes are minor. CO2 expressed as a mole fraction in dry air, micromol/mol, abbreviated as ppm.

Examples

```r
c02 <- meteo_noaa_co2()
head(c02)
plot(c02$yy_d, c02$co2_avg, type='l')
```

Description

Downloading hourly or daily (meteorological) data from the Synop stations available at https://www.ogimet.com/

Usage

```
meteo_ogimet(interval, date, coords = FALSE, station, precip_split = TRUE)
```

Arguments

- `interval`: 'daily' or 'hourly' dataset to retrieve - given as character
- `date`: start and finish date (e.g., date = c("2018-05-01", "2018-07-01")) - character or Date class object
- `coords`: add geographical coordinates of the station (logical value TRUE or FALSE)
station: WMO ID of meteorological station(s). Character or numeric vector.

precip_split: whether to split precipitation fields into 6/12/24h numeric fields (logical value TRUE (default) or FALSE); valid only for hourly time step.

Value

A data.frame of measured values with columns describing the meteorological parameters (e.g. air temperature, wind speed, cloudiness). Depending on the interval, at a given hour or day. Different parameters are returned for daily and hourly datasets.

1. station_ID - WMO station identifier
2. Lon - longitude
3. Lat - latitude
4. Date - date (and time) of observations
5. TC - air temperature at 2 metres above ground level. Values given in Celsius degrees
6. TdC - dew point temperature at 2 metres above ground level. Values given in Celsius degrees
7. TmaxC - maximum air temperature at 2 metres above ground level. Values given in Celsius degrees
8. TminC - minimum air temperature at 2 metres above ground level. Values given in Celsius degrees
9. ddd - wind direction
10. ffkmh - wind speed in km/h
11. Gustkmh - wind gust in km/h
12. P0hpa - air pressure at elevation of the station in hPa
13. PseahPa - sea level pressure in hPa
14. PTnd - pressure tendency in hPa
15. Nt - total cloud cover
16. Nh - cloud cover by high-level cloud fraction
17. HKm - height of cloud base
18. InsoD1 - insolation in hours
19. Viskm - visibility in kilometres
20. Snowcm - depth of snow cover in centimetres
21. pr6 - precipitation totals in 6 hours
22. pr12 - precipitation totals in 12 hours
23. pr24 - precipitation totals in 24 hours
24. TemperatureCAvg - average air temperature at 2 metres above ground level. Values given in Celsius degrees
25. TemperatureCMax - maximum air temperature at 2 metres above ground level. Values given in Celsius degrees
26. TemperatureCMin - minimum air temperature at 2 metres above ground level. Values given in Celsius degrees
27. TdAvgC - average dew point temperature at 2 metres above ground level. Values given in Celsius degrees
28. HrAvg - average relative humidity. Values given in %
29. WindkmhDir - wind direction
30. WindkmhInt - wind speed in km/h
31. WindkmhGust - wind gust in km/h
32. PresslevHp - Sea level pressure in hPa
33. Precmm - precipitation totals in mm
34. TotClOct - total cloudiness in octants
35. lowClOct - cloudiness by low level clouds in octants
36. SunD1h - sunshine duration in hours
37. PreselevHp - atmospheric pressure measured at altitude of station in hPa
38. SnowDepcm - depth of snow cover in centimetres

Examples

```r
# downloading data for Poznan-Lawica
poznan <- meteo_ogimet(interval = "daily", date = c("2019-06-01", "2019-07-08"),
                      station = 12330, coords = TRUE)
head(poznan)
```

nearest_stations_ogimet

*List of nearby synop stations for a defined geographical location*

Description

Returns a data frame of meteorological stations with their coordinates and distance from a given location based on the ogimet webpage. The returned list is valid only for a given day.

Usage

```r
nearest_stations_ogimet(
  country = "United+Kingdom",
  date = Sys.Date(),
  add_map = FALSE,
  point = c(0, 0),
  no_of_stations = 1,
  ...
)
```
sounding_wyoming

Arguments

country: country name; for more than two words they need to be separated with a plus character (e.g., "United+Kingdom")
date: optionally, a day when measurements were done in all available locations; current Sys.Date used by default
add_map: logical - whether to draw a map for a returned data frame (requires maps/mapdata packages)
point: a vector of two coordinates (longitude, latitude) for a point we want to find nearest stations to (e.g. c(0, 0))
no_of_stations: how many nearest stations will be returned from the given geographical coordinates
...
extra arguments to be provided to the graphics::plot() function (only if add_map = TRUE)

Value

A data.frame with number of nearest station according to given point columns describing stations parameters (e.g. ID station, distance from point, geographic coordinates) where each row represents a measurement, each station which has a measurements on selected date. If add_map = TRUE additional map of downloaded data is added.

Examples

nearest_stations_ogimet(country = "United+Kingdom", point = c(10, 50),
add_map = TRUE, no_of_stations = 60)

sounding_wyoming

Sounding data

Description

Downloading the measurements of the vertical profile of atmosphere sounding data

Usage

sounding_wyoming(wmo_id, yy, mm, dd, hh)
**sounding_wyoming**

**Arguments**

- **wmo_id**: international WMO station code (World Meteorological Organization ID); For Polish stations: Leba - 12120, Legionowo - 12374, Wrocław- 12425
- **yy**: year - single number
- **mm**: month - single number denoting month
- **dd**: day - single number denoting day
- **hh**: hour - single number denoting initial hour of sounding; for most stations this measurement is done twice a day (i.e. at 12 and 00 UTC), sporadically 4 times a day

**Value**

Returns two lists with values described at: weather.uwyo.edu; The first list contains:

1. PRES - Pressure (hPa)
2. HGHT - Height (metres)
3. TEMP - Temperature (C)
4. DWPT - Dew point (C)
5. RELH - Relative humidity (%)
6. MIXR - Mixing ratio (g/kg)
7. DRCT - Wind direction (deg)
8. SKNT - Wind speed (knots)
9. THTA = (K)
10. THTE = (K)
11. THTV = (K)

The second list contains metadata and calculated thermodynamic / atmospheric instability indices

A list of 2 data.frames where first data frame represents parameters of upper parts with columns describing the meteorogical parameters (e.g. temperature, air pressure) where each row represent a measurement, depending on the height. Secound data.frame present a description of the conditions under which the sounding was carried out.

**Source**

http://weather.uwyo.edu/upperair/sounding.html

**Examples**

```r
profile <- sounding_wyoming(wmo_id = 12120, yy = 2019, mm = 4, dd = 4, hh = 0)
head(profile)
plot(profile[[1]]$HGHT, profile[[1]]$PRES, type = 'l')
```
scraping a list of meteorological (Synop) stations for a defined country from the Ogimet webpage

**Description**

Returns a list of meteorological stations with their coordinates from the Ogimet webpage. The returned list is valid only for a given day.

**Usage**

```r
stations_ogimet(country = "United+Kingdom", date = Sys.Date(), add_map = FALSE)
```

**Arguments**

- `country`: country name; for more than two words they need to be separated with a plus character (e.g. "United+Kingdom")
- `date`: a day when measurements were done in all available locations
- `add_map`: logical - whether to draw a map with downloaded metadata (requires maps/mapdata packages)

**Value**

A data.frame with columns describing the synoptic stations in selected countries where each row represents a station. If `add_map = TRUE` additional map of downloaded data is added.

**Examples**

```r
stations_ogimet(country = "Australia", add_map = TRUE)
```
Index

*Topic **abbreviations**
  - imgw_hydro_abbrev, 6
  - imgw_meteo_abbrev, 7
*Topic **datasets**
  - imgw_hydro_abbrev, 6
  - imgw_hydro_stations, 6
  - imgw_meteo_abbrev, 7
  - imgw_meteo_stations, 8
*Topic **hydro**
  - imgw_hydro_abbrev, 6
*Topic **meteo**
  - imgw_hydro_stations, 6
  - imgw_meteo_abbrev, 7
  - imgw_meteo_stations, 8
*Topic **shortening**
  - imgw_hydro_abbrev, 6
  - imgw_meteo_abbrev, 7

```
graphics::plot(), 16
hydro_imgw, 2
hydro_imgw_annual, 3
hydro_imgw_daily, 4
hydro_imgw_monthly, 5

imgw_hydro_abbrev, 6
imgw_hydro_stations, 6
imgw_meteo_abbrev, 7
imgw_meteo_stations, 8

meteo_imgw, 8
meteo_imgw_daily, 9
meteo_imgw_hourly, 10
meteo_imgw_monthly, 11
meteo_noaa_co2, 12
meteo_OGIMET, 13

nearest_stations_OGIMET, 15

sounding_wyoming, 16
stations_OGIMET, 18
```