Package 'ISRaD'

September 16, 2020

Title Tools and Data for the International Soil Radiocarbon Database

Version 1.5.6

Description This is the central location for data and tools for the development, maintenance, analysis, and deployment of the International Soil Radiocarbon Database (ISRaD). ISRaD was developed as a collaboration between the U.S. Geological Survey Powell Center and the Max Planck Institute for Biogeochemistry. This R package provides tools for accessing and manipulating ISRaD data, compiling local data using the ISRaD data structure, and simple query and reporting functions for ISRaD. For more detailed information visit the ISRaD website at: <https://soilradiocarbon.org/>.

Depends R (>= 3.5.0)

Imports readxl,
   writexl,
   raster,
   rgdal,
   dplyr (>= 0.8),
   tidyr (>= 1.0),
   RCurl,
   ggplot2,
   maps,
   rcrossref,
   pangaear,
   stringr

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Suggests devtools,
   knitr,
   rmarkdown

R topics documented:

checkTemplateFiles   *Check ISRaD Template/Info files*

Description

Check that the template information file and the template file match appropriately.

Usage

```
checkTemplateFiles(outfile = "", verbose = TRUE)
```

Arguments

outfile	file to dump the output report. Defaults to an empty string that will print to standard output
verbose	if TRUE (default) will print output to specified outfile

Details

Used in compile() function, but primarily a development tool

Value

Nothing (run for side effects).

Examples

```
checkTemplateFiles()
```

compile *Compile ISRaD data product*

Description

Compiles template files into ISRaD database format.

Usage

```
compile(dataset_directory, write_report = FALSE, write_out = FALSE,
  return_type = c("none", "list")[2], checkdoi = FALSE,
  verbose = TRUE)
```

Arguments

dataset_directory

Directory where completed QAQCed template files are stored.

write_report   Boolean flag to write a log file of the compilation. File will be in the specified
dataset_directory at "database/ISRaD_log.txt". If a file with this name already
exists in this directory it will be overwritten.

write_out   Set to TRUE to write the compiled database file in .xlsx format in dataset_directory

return_type   A string that defines return object. Acceptable values are "none" or "list"; default
is "list".

checkdoi   Set to FALSE if you do not want to validate DOIs during QAQC. (Warning:
time consuming).

verbose   Set to TRUE to print results of function to console.

Examples

```
# Load example dataset Gaudinski_2001
entry <- ISRaD::Gaudinski_2001
# Save as .xlsx file
ISRaD.save.entry(
  entry = entry,
 template_file = system.file("extdata", "ISRaD_Master_Template.xlsx", package = "ISRaD"),
  outfile = file.path(tempdir(), "Gaudinski_2001.xlsx")
)
# Compile .xlsx file/s in dataset_directory into ISRaD database object
ISRaD.compiled <- compile(tempdir(),
  write_report = TRUE, write_out = TRUE,
  return_type = "list", checkdoi = FALSE, verbose = TRUE
)
```

---

convert_fm_d14c *convert_fm_d14c*

Description

convert fraction modern to d14c and d14c to fraction modern

Usage

```
convert_fm_d14c(fm = NA, d14c = NA, obs_date_y)
```

Arguments

fm	fraction modern (default NA)
d14c	delta 14C in per mille (default NA)
obs_date_y	year of observation/sample collection

Details

convenience function for radiocarbon unit conversions

Author(s)

J. Beem-Miller

Examples

```
convert_fm_d14c(fm = 0.97057, obs_date_y = 2005)
convert_fm_d14c(d14c = -35.86611, obs_date_y = 2005)
```

---

future14C *Future atmospheric 14C dataset for delta delta calculation*

Description

Data from: Sierra, C. "Forecasting atmospheric radiocarbon decline to pre-bomb values", Radiocarbon, Vol 60, Nr 4, 2018, p 1055–1066 DOI:10.1017/RDC.2018.33

Usage

```
future14C
```

Format

dataframe

Gaudinski_2001	*Gaudinski Harvard Forest example dataset*

Description

Data from Gaudinski, J., 2001, Belowground carbon cycling in three temperate forests of the eastern United States, University of California Irvine, Ph.D. thesis

Usage

```
Gaudinski_2001
```

Format

list

graven	*Graven dataset for delta delta calculation*

Description

Data from Graven et al 2017 https://www.geosci-model-dev.net/10/4405/2017/gmd-10-4405-2017.pdf

Usage

```
graven
```

Format

dataframe

ISRaD.extra	*ISRaD.extra*

Description

Fills in transformed and geospatial data where possible, generating an enhanced version of ISRaD.

Usage

```
ISRaD.extra(database, geodata_directory)
```

Arguments

database	ISRaD dataset object
geodata_directory	
	directory where geospatial data are found

Details

Fills fraction modern, delta 14C, delta-delta, profile coordinates, bulk density, organic C concentration, and SOC stocks from entered data; fills soil and climatic data from external geospatial data products

Value

New ISRaD_extra object with derived, transformed, and filled columns.

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
# Fill ISRaD.extra data
database.x <- ISRaD.extra(database,
  geodata_directory = system.file("extdata", "geodata_directory", package = "ISRaD")
)
```

ISRaD.extra.Cstocks   *ISRaD.extra.Cstocks*

Description

Calculates soil organic carbon stock

Usage

```
ISRaD.extra.Cstocks(database)
```

Arguments

database   ISRaD dataset object.

Details

Function first fills lyr_bd_samp, lyr_c_org, lyr_c_org, lyr_coarse_tot. Notes: 1) SOC stocks can only be calculated if organic carbon concentration and bulk density data are available, 2) SOC stocks are calculated for the fine earth fraction (<2mm).

Value

ISRaD_data object with filled columns "lyr_coarse_tot_filled", "lyr_bd_samp_filled", "lyr_c_inorg_filled", "lyr_c_org_filled", "lyr_soc_filled".

Author(s)

J. Beem-Miller

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
database.x <- ISRaD.extra.Cstocks(database)
```

ISRaD.extra.delta_delta

ISRaD.extra.delta_delta

Description

Calculates the difference between sample delta 14C and the atmosphere for the year of collection (delta-delta)

Usage

```
ISRaD.extra.delta_delta(database, future = TRUE)
```

Arguments

database	ISRaD dataset object
future	Project atmospheric radiocarbon into the future? TRUE/FALSE

Details

Creates new column for delta-delta value. Observation year and profile coordinates must be filled (use ISRaD.extra.fill_dates, and ISRaD.extra.fill_coords functions). The relevant atmospheric d14C data (northern or southern hemisphere or tropics) are determined by profile coordinates. Projection for 2016 to 2021 uses the four quarter average projected atmospheric radiocarbon concentration for Central Europe as estimated in Sierra (2019). Notes: Central Europe projection used for northern hemisphere as these projections perform better against observations than northern hemisphere projection; southern hemisphere and tropic atmospheric radiocarbon projection lagged by 2.5 per mille, as this is the mean lag observed from 2000 to 2015 in the Graven (2017) dataset.

Value

ISRaD_data object with new delta delta columns in relevant tables.

Author(s)

J. Beem-Miller and C. Hicks-Pries

References

Graven et al. 2017 <https://www.geosci-model-dev.net/10/4405/2017/gmd-10-4405-2017.pdf>; Sierra, C. "Forecasting atmospheric radiocarbon decline to pre-bomb values", Radiocarbon, Vol 60, Nr 4, 2018, p 1055–1066 DOI:10.1017/RDC.2018.33

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
# Fill profile coordinates
database.x <- ISRaD.extra.fill_coords(database)
# Fill dates
database.x <- ISRaD.extra.fill_dates(database.x)
# Fill delta 14C from fraction modern
database.x <- ISRaD.extra.fill_14c(database.x)
# Fill delta delta
database.x <- ISRaD.extra.delta_delta(database.x)
```

ISRaD.extra.fill_14c   *ISRaD.extra.fill_14c*

Description

Fills delta 14C from fraction modern if delta 14C not reported.

Usage

```
ISRaD.extra.fill_14c(database)
```

Arguments

database   ISRaD dataset object.

Details

Warning: xxx_obs_date_y columns must be filled for this to work!

Value

ISRaD_data object with filled delta 14C columns.

Author(s)

J. Beem-Miller & A. Hoyt

References

Stuiver and Polach, 1977

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
# Note that flx_14c values are NA
is.na(database$flux$flx_14c)
# Fill dates
database.x <- ISRaD.extra.fill_dates(database)
# Fill delta 14C from fraction modern
database.x <- ISRaD.extra.fill_14c(database.x)
# Column flx_14c in the "flux" table is now filled
is.na(database$flux$flx_14c)
```

```
ISRaD.extra.fill_coords
```
ISRaD.extra.fill_coords

Description

Fills profile coordinates from site coordinates if profile coordinates not reported.

Usage

```
ISRaD.extra.fill_coords(database)
```

Arguments

database   ISRaD dataset object.

Value

ISRaD_data object with filled profile coordinates.

Author(s)

J. Beem-Miller

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
# Fill profile coordinates
database.x <- ISRaD.extra.fill_coords(database)
```

```
ISRaD.extra.fill_dates
```
ISRaD.extra.fill_dates

Description

Fills frc_obs_date_y and inc_obs_date_y columns from lyr_obs_date_y if not reported.

Usage

```
ISRaD.extra.fill_dates(database)
```

Arguments

database   ISRaD dataset object.

Details

QAQC does not require frc_obs_date_y or inc_obs_date_y fields to be filled in. Therefore it is recommended to run this function prior to running the functions ISRaD.extra.fill_14c, ISRaD.extra.fill_fm, and ISRaD.extra.delta_delta, which require xxx_obs_date_y data.

Value

ISRaD_data object with filled frc_obs_date_y and inc_obs_date_y fields.

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
# Fill dates
database.x <- ISRaD.extra.fill_dates(database)
# Fraction table now has lyr_obs_date_y values in frc_obs_date_y field
```

ISRaD.extra.fill_fm   *ISRaD.extra.fill_fm*

Description

Fills fraction modern from delta 14C if fraction modern not reported.

Usage

```
ISRaD.extra.fill_fm(database)
```

Arguments

database   ISRaD dataset object.

Details

Warning: xxx_obs_date_y columns must be filled for this to work!

Value

ISRaD_data object with filled fraction modern columns.

Author(s)

J. Beem-Miller & A. Hoyt

References

Stuiver and Polach, 1977

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
# Fill dates
database.x <- ISRaD.extra.fill_dates(database)
# Fill fraction modern from delta 14C
database.x <- ISRaD.extra.fill_fm(database.x)
```

```
ISRaD.extra.geospatial
```
ISRaD.extra.geospatial

Description

Extracts data from a user-supplied raster file and adds data as a new variable at the profile level

Usage

```
ISRaD.extra.geospatial(database, geodata_directory,
  crs = "+proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0",
  fillWorldClim = TRUE)
```

Arguments

database	ISRaD dataset object
geodata_directory	
	Directory where geospatial data are found
crs	Coordinate reference system used for geospatial datasets
fillWorldClim	Option to fill climate data from the Worldclim dataset (downloads data from web)

Details

Generic function that uses geographic coordinates of profiles to extract data from one or more raster files. Raster data will be added as new variables at the profile level.

The new variable name will be a concatenation of "pro_", and the file name (excluding the file extension). The ISRaD file name convention for geospatial files uses a 6 component string, separated by "_". Missing components can be replaced with "x" ("x"s will be dropped before creating variable names). The 6 components are as follows:

1) Short description of the data type, e.g. "bd" for bulk density
2) Top layer depth or exact depth (numeric, cm)
3) Bottom layer depth (numeric, cm)
4) Year of data observation (numeric)
5) Data units (e.g. mmyr for mean annual precipitation)
6) Any relevant notes

Coordinate reference system can be specified with the "crs" argument; default is WGS84. Note that all files in geodata_directory must use the same crs.

Option "fillWorldClim" fills climate data from worldclim V1.4 at 2.5 resolution (http://www.worldclim.org/bioclim). Variable descriptions are as follows:
bio1 = Annual Mean Temperature,
bio2 = Mean Diurnal Range (Mean of monthly (max temp - min temp)),
bio3 = Isothermality (BIO2/BIO7) (* 100),
bio4 = Temperature Seasonality (standard deviation *100),
bio5 = Max Temperature of Warmest Month,

bio6 = Min Temperature of Coldest Month,

bio7 = Temperature Annual Range (BIO5-BIO6),

bio8 = Mean Temperature of Wettest Quarter,

bio9 = Mean Temperature of Driest Quarter,

bio10 = Mean Temperature of Warmest Quarter,

bio11 = Mean Temperature of Coldest Quarter,

bio12 = Annual Precipitation,

bio13 = Precipitation of Wettest Month,

bio14 = Precipitation of Driest Month,

bio15 = Precipitation Seasonality (Coefficient of Variation),

bio16 = Precipitation of Wettest Quarter,

bio17 = Precipitation of Driest Quarter,

bio18 = Precipitation of Warmest Quarter,

bio19 = Precipitation of Coldest Quarter

Value

Updated ISRaD_extra object with new columns at the profile level

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
# Fill profile coordinates
database <- ISRaD.extra.fill_coords(database)
# Run function
# Note that geospatial data in pkg is only for the Gaudinski_2001 dataset
# Users may supply their own geospatial data as long as it can be read by the raster package
database.x <- ISRaD.extra.geospatial(database,
  geodata_directory = system.file("extdata", "geodata_directory", package = "ISRaD"),
  fillWorldClim = TRUE
)
```

ISRaD.extra.geospatial.keys

ISRaD.extra.geospatial.keys

Description

Recode numeric values from categorical geospatial data products

Usage

```
ISRaD.extra.geospatial.keys(database, geodata_keys)
```

Arguments

database	ISRaD dataset object
geodata_keys	directory where geospatial data are found

Details

Generic function that reads .csv files paired with categorical raster data and recodes extracted data in the ISRaD_extra object. For the function to work, the .csv filenames must be identical to the corresponding raster filenames (except for the file extension). Additionally, the first column of the .csv file must contain the numeric identifier and the second column the corresponding character value.

Value

Updated ISRaD_extra object with recoded columns.

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
# Fill profile coordinates
database <- ISRaD.extra.fill_coords(database)
# Fill geospatial data
database.x <- ISRaD.extra.geospatial(database,
  geodata_directory = system.file("extdata", "geodata_directory", package = "ISRaD"),
  fillWorldClim = FALSE
)
# Recode numeric data to categorical
database.x <- ISRaD.extra.geospatial.keys(database.x,
  geodata_keys = system.file("extdata", "geodata_keys", package = "ISRaD")
)
```

ISRaD.flatten *ISRaD.flatten*

Description

Joins tables in ISRaD based on linking variables and returns "flat" dataframe/s

Usage

```
ISRaD.flatten(database, table)
```

Arguments

database   ISRaD dataset object: e.g. ISRaD_data, or ISRaD_extra

table   ISRaD table of interest ("flux", "layer", "interstitial", "fraction", "incubation").
             Must be entered with "".

Details

ISRaD.extra.flatten generates flat files (2 dimensional matrices) for user-specified ISRaD tables by joining higher level tables (metadata, site, profile, layer) to lower level tables (layer, fraction, incubation, flux, interstitial).

Value

A dataframe with nrow=nrow(table) and ncol=sum(ncol(meta),ncol(site),ncol(profile),...,ncol(table))

Author(s)

J. Beem-Miller

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
fractions <- ISRaD.flatten(database, "fraction")
layers <- ISRaD.flatten(database, "layer")
```

ISRaD.getdata   *ISRaD.getdata*

Description

Retrieves most recent version of ISRaD data from github

Usage

```
ISRaD.getdata(directory, dataset = "full", extra = FALSE,
  force_download = FALSE)
```

Arguments

directory	Location of ISRaD_database_files folder. If not found, it will be created.
dataset	Specify which data you want. Options are c("full", flux","interstitial","incubation","fraction","layer")
extra	TRUE/FALSE. If TRUE, the ISRaD_extra object will be returned. If FALSE, ISRaD_data will be returned. Default is FALSE.
force_download	TRUE/FALSE. If ISRaD_database files already exist in the specified directory, new data will not be downloaded by default. If force_download is set to TRUE, the newest data from github will be downloaded and overwrite any existing files.

Value

ISRaD data object

Examples

```
# Return full dataset ("full")
ISRaD_full <- ISRaD.getdata(tempdir(), dataset = "full", extra = FALSE)
# Return full dataset plus "extra" filled data
ISRaD_extra <- ISRaD.getdata(tempdir(), dataset = "full", extra = TRUE)
# Return only fraction data, including filled fraction data
ISRaD_fractions <- ISRaD.getdata(tempdir(), dataset = "fraction", extra = TRUE)
```

ISRaD.read.entry *ISRaD.read.entry*

Description

Reads ISRaD data object from Excel file in standard template format

Usage

```
ISRaD.read.entry(entry, template_file = system.file("extdata",
  "ISRaD_Master_Template.xlsx", package = "ISRaD"))
```

Arguments

entry   ISRaD data object.

template_file   Directory path and name of template file to use (defaults to the ISRaD_Master_Template
                file built into the package). Not recommended to change this.

Author(s)

J. Beem-Miller

Examples

```
# Load example dataset Gaudinski_2001
entry <- ISRaD::Gaudinski_2001
ISRaD.save.entry(
  entry = entry,
  template_file = system.file("extdata", "ISRaD_Master_Template.xlsx", package = "ISRaD"),
  outfile = file.path(tempdir(), "Gaudinski_2001.xlsx")
)
# Read in .xlsx file
ISRaD.read.entry(file.path(tempdir(), "Gaudinski_2001.xlsx"))
```

ISRaD.rep.count.all *ISRaD.rep.count.all*

Description

Generates a report of counts of observations at each level of the database

Usage

```
ISRaD.rep.count.all(database)
```

Arguments

database   ISRaD data object

Value

A [tibble](#) of observation counts, one column for each database table.

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
ISRaD.rep.count.all(database)
```

ISRaD.rep.count.frc   *ISRaD.rep.count.frc*

Description

Generates a report of fraction level observations, including fraction scheme and properties. Note that this function only counts rows, not 14C observations.

Usage

```
ISRaD.rep.count.frc(database)
```

Arguments

database   ISRaD data object

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
ISRaD.rep.count.frc(database)
```

ISRaD.rep.entry.stats *ISRaD.rep.entry.stats*

Description

Generates a report of metadata statistics for all entries

Usage

```
ISRaD.rep.entry.stats(database)
```

Arguments

database   ISRaD data object

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
ISRaD.rep.entry.stats(database)
```

ISRaD.rep.site.map   *ISRaD.rep.site.map*

Description

Generate a world map showing locations of all ISRaD sites

Usage

```
ISRaD.rep.site.map(database)
```

Arguments

database   ISRaD data object

Examples

```
# Obtain current ISRaD data
database <- ISRaD.getdata(tempdir(), dataset = "full", extra = FALSE)
# Generate a map of all ISRaD sites
ISRaD.rep.site.map(database)
```

ISRaD.report   *ISRaD.report*

Description

Generate basic summary reports of ISRaD data

Usage

```
ISRaD.report(database, report)
```

Arguments

database   ISRaD data object

report   Parameter to define which type of report you want. The default is "count.all" other options include "entry.stats", "count.frc", or "site.map".

Details

Wrapper for the simple reporting functions ISRaD.rep.count.all, ISRaD.rep.count.frc, ISRaD.rep.entry.stats, ISRaD.rep.site.map

Examples

```
# Load example dataset Gaudinski_2001
database <- ISRaD::Gaudinski_2001
# Report metadata statistics
ISRaD.report(database, report = "entry.stats")
# Report summary statistics for all levels of the database
ISRaD.report(database, report = "count.all")
# Generate a map of all ISRaD sites
ISRaD.report(database, report = "site.map")
```

ISRaD.save.entry   *ISRaD.save.entry*

Description

Saves ISRaD data object to .xlsx file

Usage

```
ISRaD.save.entry(entry, template_file = system.file("extdata",
  "ISRaD_Master_Template.xlsx", package = "ISRaD"), outfile)
```

Arguments

entry	ISRaD data object
template_file	Directory path and name of template file to use (defaults to the ISRaD_Master_Template file built into the package). Not recommended to change this.
outfile	File name and path for .xlsx output

Details

This function can be used to save a single entry (or a compiled database in the standard template format) to a .xlsx file. Replaces the function "ISRaD.save.xlsx" as that function depended on the package openxlsx, which was unstable at the time. This a simpler function and does not maintain the formating of the template file. The code for the original function is available in the ISRaD github repository in the [devScripts](#) directory.

Author(s)

J. Beem-Miller

QAQC   *QAQC*

Description

Checks template files for data coherence, formatting, and data entry errors

Usage

```
QAQC(file, writeQCreport = FALSE, outfile_QAQC = "",
  summaryStats = TRUE, dataReport = FALSE, checkdoi = TRUE,
  verbose = TRUE)
```

Arguments

file	File path for template file to be checked
writeQCreport	If TRUE, a text report of the QC output will be written to the outfile. Default is FALSE
outfile_QAQC	Filename of the output file (if writeQCreport is TRUE). Default is NULL, with the outfile being written to the directory where the template file is stored and named according to the file being checked.
summaryStats	Prints summary statistics. Default is TRUE.
dataReport	Prints list structure of database. Default is FALSE.
checkdoi	Set to FALSE if you do not want the QAQC check to validate DOIs (if TRUE this will be time consuming). Default is TRUE.
verbose	Set to TRUE to print results of function to console. Default is TRUE.

Details

This function can also be called from the ISRaD website (<http://soilradiocarbon.org>).

Examples

```
# Load example dataset Gaudinski_2001
entry <- ISRaD::Gaudinski_2001
# Save as .xlsx file
ISRaD.save.entry(
  entry = entry,
  template_file = system.file("extdata", "ISRaD_Master_Template.xlsx", package = "ISRaD"),
  outfile = file.path(tempdir(), "Gaudinski_2001.xlsx")
)
# Run QAQC
QAQC(file.path(tempdir(), "Gaudinski_2001.xlsx"))
```

Index