Package: hoboR (via r-universe)

June 7, 2026

Title Weather Station Data Summarization and Manipulation for HOBO Data Loggers

Version 1.1.0

Author Ricardo I. Alcala Briseno [aut, cre], Adam Carson [ctb], Yung-Hsiang Lan [ctb], Ebba Peterson [ctb], Niklaus J. Grunwald [ctb], Jared M. LeBoldus [ctb]

Maintainer Ricardo I. Alcala Briseno <ria5282@psu.edu>

Description Processing of CSV files generated by HOBO weather stations and data loggers. The package automatically imports multiple HOBO data records, removes duplicate records, identifies impossible values, subsets user-defined time ranges, and summarizes environmental data.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Depends R (>= 4.3.0)

Imports dplyr, lubridate, reshape, reshape2, ggplot2, tidyr, scales

Suggests testthat (>= 3.0.0)

URL https://www.r-project.org, https://leboldus-lab.github.io/hoboR/

BugReports https://github.com/LeBoldus-Lab/hoboR/issues

Config/testthat/edition 3

Config/pak/sysreqs libicu-dev

Repository https://leboldus-lab.r-universe.dev

Date/Publication 2026-06-07 17:08:38 UTC

RemoteUrl https://github.com/leboldus-lab/hobor

RemoteRef HEAD

RemoteSha 55d25422125b23521c045195b4581e6e45e3b0ca

Contents

`calibrator` *Calibrator HOBO data in CSV format*

Description

Calculates the difference between HOBO devices under controlled conditions. This additive function calculates the difference among hobo loggers using a base correction to HOBO loggers.

Usage

```
calibrator(list.data, columns = c(2, 7, 12), times, round = 7)
```

Arguments

`list.data`	A list containing the HOBO CSV files.
`columns`	The columns to be used for calibration.
`times`	The times in a vector of dates to be included in the calibration process.
`round`	The number of decimal places to round the results to.

Value

a data frame with the differences for data correction, to use with corrector

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata/calibration", package = "hoboR")

folder=paste0(rep("canopy", 5), 1:5)

pathtoread = data = list()

for (i in seq_along(folder)){
  pathtoread[[i]] <- paste0(path, "/",folder[i])
  # Loading all hobo files
  data[[i]] <- hobinder(as.character(pathtoread[i]), header = TRUE, skip = 0,
   channels = "ON" ) # channels is a new feature
}

# Make sure you enter the date & time format with zeros,
# for example 08:00 instead of 8:00 for 8am.
times <- c("2022-03-22 01:00", "2022-03-22 02:00", "2022-03-22 03:00",
           "2022-03-22 04:00", "2022-03-22 05:00", "2022-03-22 06:00",
           "2022-03-22 07:00", "2022-03-22 08:00", "2022-03-22 09:00")

variables <- c(3, 8, 13) # Select the weather variables

calibrationmeans <- calibrator(data, columns= variables, times = times)
```

`correction` *Correction test for HOBO data from calibrator*

Description

Additive function to calculate the difference among hobo loggers to calibrate using a base correction to the data

Usage

```
correction(data, w.var = "FULL", calibrate = calibrate)
```

Arguments

`data`	a list of CVS data containing the hobo
`w.var`	a column to correct the weather variable e.g., Temperature, RH (relative humidity), or FULL, will use the output of calibrator
`calibrate`	a value to correct the weather variable, must be numeric or USEFILE, will use the output of calibrator

Value

a data frame with the differences for data correction, to use with corrector

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata/calibration", package = "hoboR")

folder=paste0(rep("canopy", 5), 1:5)

pathtoread = data = list()

for (i in seq_along(folder)){
  pathtoread[[i]] <- paste0(path, "/",folder[i])
  # Loading all hobo files
  data[[i]] <- hobinder(as.character(pathtoread[i]), header = TRUE, skip = 0,
   channels = "ON" ) # channels is a new feature
}

# Double-check you enter the same date format
times <- c("2022-03-22 01:00", "2022-03-22 02:00", "2022-03-22 03:00",
            "2022-03-22 04:00","2022-03-22 05:00", "2022-03-22 06:00",
            "2022-03-22 07:00", "2022-03-22 08:00","2022-03-22 09:00")

variables <- c(3, 8, 13) # Select the weather variables

meanvars <- calibrator(data, columns = variables, times = times)

calibratedfiles <- correction(data = data,
                              w.var = "FULL",
                              calibrate = meanvars)
```

`correction.test` *Correction Test for HOBO data from calibrator*

Description

This function calculates the difference among HOBO loggers, finding the variance and using it as a base correction. It's designed to adjust HOBO data based on calibration files and specified thresholds.

Usage

```
correction.test(
  list.data,
  calibrationfile,
  w.var = c(3, 8, 13),
  times = times,
  threshold = c(1, 5, 10)
)
```

Arguments

list.data	A list of CSV data frames containing the HOBO data.
calibrationfile	A data frame representing the calibration file.
w.var	A vector of column indices to be used in the correction.
times	A vector of times for which the data is relevant.
threshold	A vector of threshold values for passing the correction test. The smaller the value the highest precision.

Value

A data frame with the differences for data correction, to be used with a corrector.

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata/calibration", package = "hoboR")

folder=paste0(rep("canopy", 5), 1:5)

pathtoread = data = list()

for (i in seq_along(folder)){
  pathtoread[[i]] <- paste0(path, "/",folder[i])
  # Loading all hobo files
  data[[i]] <- hobinder(as.character(pathtoread[i]), header = TRUE, skip = 0,
   channels = "ON" ) # channels is a new feature
}

# Double-check you enter the same date format
times <- c("2022-03-22 01:00", "2022-03-22 02:00", "2022-03-22 03:00",
            "2022-03-22 04:00","2022-03-22 05:00", "2022-03-22 06:00",
            "2022-03-22 07:00", "2022-03-22 08:00","2022-03-22 09:00")

variables <- c(3, 8, 13) # Select the weather variables

meanvars <- calibrator(data, columns = variables, times = times)

correction.test(list.data = data, calibrationfile =  meanvars,
                w.var = variables,
                times = times,
                threshold = c(1, 5, 10))
```

count_NAs *HOBO count NA's This function counts the number of NA's in your data set*

Description

HOBO software

Usage

```
count_NAs(data, w.var)
```

Arguments

data	dataframe with suspected NA's
w.var	weather variables to test

Value

Returns the numbers of NAs for the impossible values

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata", package = "hoboR")

csvfiles <- hobinder(path, header = TRUE, skip = 1, channels = "OFF")

cleancsv <- hobocleaner(csvfiles)

NAdata <- sensorfailures(cleancsv, condition = "<", threshold = c(0, 0),
                            w.var = c("Rain", "Wetness"))

count_NAs(NAdata, "Temp")
```

hobinder *Reads HOBO data in CSV format*

Description

Two functions that read the original data downloaded from HOBO software adding file names as metadata for each .csv file and cleans the data from duplicates creating a continuous file from all .csv's

Usage

```
hobinder(path, channels = "OFF", ...)
```

Arguments

path	select the path to the directory with the csv files
channels	turn on or off additional channels in HOBO data logger, default "OFF"
...	arguments to be passed to methods

Value

large csv file

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata", package = "hoboR")

csvfiles <- hobinder(path, header = TRUE, skip = 1, channels = "OFF")

head(csvfiles)
```

hobocleaner *Reads HOBO data in CSV format*

Description

Functions that cleans the original data downloaded from the HOBO software, cleans the data and removes duplicates creating a continuous file from all .csv's

Usage

```
hobocleaner(file, format = "ymd", na.rm = TRUE)
```

Arguments

file	CSV from hobinder
format	Select the time format, month, day, and year (mdy), year, month, and day (ymd) or year two digits, month and day (yymd)
na.rm	TRUE or FALSE to remove NAs, TRUE is default

Value

formatted data frame and duplicate values removed

Author(s)

Ricardo I Alcala Briseno, <ria5292@psu.edu>

Examples

```
path <- system.file("extdata", package = "hoboR")

csvfiles <- hobinder(path, header = TRUE, skip = 1, channels = "OFF")

cleancsv <- hobocleaner(csvfiles)

head(cleancsv)
```

---

hobocorrelations *Correlation plot for variables within a time range*

Description

This function provides a time point for a specified number of days. HOBO software

Arguments

data	A data frame with the HOBO data and a Date column
summariseby	Provide the interval date to present (e.g., "month")
by	Summary function for aggregation (e.g., "mean")
na.rm	Logical, whether to remove NAs from the result

Value

A ggplot object representing the correlation heatmap

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata", package = "hoboR")

csvfiles <- hobinder(path, header = TRUE, skip = 1, channels = "OFF")

cleaned <- hobocleaner(csvfiles, format = "ymd")

hobocorrelations(cleaned, summariseby = "month", by = "mean", na.rm = FALSE)
```

`hoborange` *Calculate Date Range Temperature Using HOBO Data*

Description

This function calculates the means for weather data collected by HOBO loggers over a specified date range. It's designed for use with data exported from HOBO software.

Usage

```
hoborange(
  data,
  start = "2022-08-04 09:05",
  end = "2022-10-04 09:05",
  na.rm = TRUE
)
```

Arguments

`data`	A data frame containing the HOBO data, including a `Date` column in POSIXct format.
`start`	The start of the date range in "yyyy-mm-dd HH:MM" format.
`end`	The end of the date range in "yyyy-mm-dd HH:MM" format.
`na.rm`	A logical value indicating whether NA values should be removed before calculation.

Value

A subset of the original data frame limited to the specified date range.

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata", package = "hoboR")

csvfiles <- hobinder(path, header = TRUE, skip = 1, channels = "OFF")

cleancsv <- hobocleaner(csvfiles)

site.ranges <- hoborange(cleancsv, start = "2022-08-04 09:05",
                         end = "2022-10-04 09:05")
```

`hobotime` *Summarise HOBO data by time intervals*

Description

This function calculates hobo weather by minutes HOBO software

Usage

```
hobotime(data, summariseby = "5 mins", na.rm = TRUE, na.action = na.omit)
```

Arguments

`data`	a data frame with the hobo data and a `Date` column
`summariseby`	a time interval in minmutes
`na.rm`	logical vector TRUE or FALSE
`na.action`	na.omit remove rows with NA's, na.pass keeps NA's

Value

a data frame summarized by minutes

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata", package = "hoboR")

data <- hobinder(path, header = TRUE, skip = 1, channels = "OFF")

cleandata <- hobocleaner(data, format = "ymd")

# this function needs clean data
subset <- hobotime(cleandata,  summariseby = "5 mins", na.rm = TRUE)

head(subset)
```

impossiblevalues *HOBO impossible values*

Description

Functions that gets the mean by date of the cleaned data downloaded from the HOBO software

Usage

```
impossiblevalues(data, showrows = 10, ...)
```

Arguments

data	Cleaned hobo data frame from original csv or hobocleaner and hobotime
showrows	Number of rows to show for maximum values, default is 10
...	arguments to be passed to methods

Value

Gives the rows with impossible values

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata", package = "hoboR")

csvfiles <- hobinder(path, header = TRUE, skip = 1, channels = "OFF")

cleancsv <- hobocleaner(csvfiles)

impossiblevalues(cleancsv, showrows = 10)
```

meanhobo *mean HOBO data in CSV format*

Description

Functions that gets the summary statistics by summarizing by date the cleaned data downloaded from the HOBO software

Arguments

data	cleaned hobo data frame from hobocleaneri
summarisedby	select a time interval 60 min, 24 hours, 1 day
na.rm	TRUE or FALSE to remove NAs, TRUE is default
minmax	TRUE or FALSE to retain min and max temperatures

Value

smaller data frame with means and standard deviation

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata", package = "hoboR")

data <- hobinder(path, header = TRUE, skip = 1, channels = "OFF")

cleandata <- hobocleaner(data, format = "ymd")

# this function needs clean data#needs clean data
hobodata <- meanhobo(cleandata, summariseby = "5 mins",
                                    na.rm = TRUE, minmax = TRUE)
```

---

sampling.rates *Calculates sampling rates from a CSV format*

Description

This function calculates incidence and rates for baiting Phytophthora collected on dates for baited and removed leaves HOBO software

Usage

```
sampling.rates(samples, n, round)
```

Arguments

samples	a csv with the format
n	Mandatory. Specifies the number of replicates of the experiment
round	Optional. Specifies the number of decimal places for rounding the output incidence calculated from a csv table designed for baiting Phytophthora

Value

smaller data frame with incidence and rates

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata/sampling", package = "hoboR")
#'
samples <- read.csv(paste0(path, "/", "sampling.csv"))

samp.rates <- sampling.rates(samples, n = 9, round = 2)
```

`sensorfailures` *HOBO Remove Values This function is designed to identify and address values that are likely to be out of range, such as temperatures exceeding the maximum recorded in your study area or relative humidity (RH) values exceeding 100%, which are not feasible. The* `sensorfailure()` *function allows you to detect and flag these anomalous readings, facilitating the cleansing of your dataset by replacing all out-of-range values with NA's, maintaining the integrity and accuracy of of your data analysis.*

Description

HOBO software

Usage

```
sensorfailures(
  data,
  condition = ">",
  threshold = c(34, 8),
  w.var = c("Temp", "Rain")
)
```

Arguments

`data`	Convert to NA the impossible values from HOBO data frame from `original csv` OR `hobocleaner` OR `hobotime`
`condition`	The condition for removal, one of (">", "<", "==", ">=", "<=")
`threshold`	Numeric vector specifying the threshold values for removal
`w.var`	weather variables, can be a single or multiple variables

Value

Returns the data with NAs for the impossible values

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata", package = "hoboR")

csvfiles <- hobinder(path, header = TRUE, skip = 1, channels = "OFF")

data <- hobocleaner(csvfiles)

sensorfailures(data, condition = ">",
                threshold = c(50, 3000, 101),
                w.var = c("Temp", "Rain", "Wetness"))

NAdata <- sensorfailures(data, condition = "<",
                threshold = c(0, 0),
                w.var = c("Rain", "Wetness"))
```

---

testhobolist *Test HOBO data for calibration*

Description

Check if the times date are present in the list of HOBO files

Usage

```
testhobolist(data, times)
```

Arguments

data	a list of CVS data containing hobo data
times	a series of times <- c("2022-03-22 01:00", "2022-03-22 02:00", "2022-03-22 03:00")

Value

a data frame with the total entries and the count of entries present in each data set

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata/calibration", package = "hoboR")

folder=paste0(rep("canopy", 5), 1:5)

pathtoread = data = list()

for (i in seq_along(folder)){
  pathtoread[[i]] <- paste0(path, "/",folder[i])
  # Loading all hobo files
  data[[i]] <- hobinder(as.character(pathtoread[i]), header = TRUE, skip = 0,
   channels = "ON" ) # channels is a new feature
}

# Make sure you enter the date & time format with zeros,
# for example 08:00 instead of 8:00 for 8am.
times <- c("2022-03-22 01:00", "2022-03-22 02:00", "2022-03-22 03:00",
           "2022-03-22 04:00", "2022-03-22 05:00", "2022-03-22 06:00",
           "2022-03-22 07:00", "2022-03-22 08:00", "2022-03-22 09:00")

data <- testhobolist(data, times)
```

---

timestamp *Timestamp for Specific Intervals*

Description

This function provides a time point for a specified number of days. HOBO software

Usage

```
timestamp(
  data,
  stamp = "yyyy/mm/dd: ss",
  by = "24 hours",
  days = 100,
  na.rm = TRUE,
  plot = TRUE,
  var = "Temp"
)
```

Arguments

data	A data frame with the HOBO data and a Date column
stamp	Provide a date
by	Provide the interval date to present (e.g., "24 hours")
days	Number of days for the interval

na.rm	Logical, whether to remove NAs from the result
plot	Logical, whether to generate a plot
var	Variable to plot (default is "Temp")

Value

A data frame summarized by minutes

Author(s)

Ricardo I Alcala Briseno, <ria5282@psu.edu>

Examples

```
path <- system.file("extdata", package = "hoboR")

data <- hobinder(path, header = TRUE, skip = 1, channels = "OFF")

cleandata <- hobocleaner(data, format = "ymd")

datastamp <- timestamp(cleandata, stamp ="2022-08-04 12:00",
                       by = "1 hour", days = 60, na.rm = TRUE,
                       plot = TRUE, var = "Temp")
```

Index