Package: ORFID (via r-universe)

August 23, 2024

Type Package

Version 1.0.2.9000

Title Manage and Summarize Data from Oregon RFID ORMR and ORSR Antenna
Readers

Description Automates and standardizes the import of raw data from
Oregon RFID (radio-frequency identification) ORMR (Oregon RFID
Multi-Reader) and ORSR (Oregon RFID Single Reader) antenna
readers. Compiled data can then be combined within multi-reader
arrays for further analysis, including summarizing tag and
reader detections, determining tag direction, and calculating
antenna efficiency.

Date 2022-12-06

Depends R (>= 4.0.0)

Imports magrittr, dplyr, readr, tidyr, ggplot2, stringr, rlang,
openxlsx

License MIT + file LICENSE

URL https://github.com/hugo-marques/ORFID

BugReports https://github.com/hugo-marques/ORFID/issues

LazyData TRUE

Encoding UTF-8

RoxygenNote 7.2.1

Suggests knitr, rmarkdown, testthat (>= 2.1.0), devtools

VignetteBuilder knitr

Repository https://hugo-marques.r-universe.dev

RemoteUrl https://github.com/hugo-marques/orfid

RemoteRef HEAD

RemoteSha b06851a3696ac07545b9719f6c35d6c65b528d9d

Contents

---

ant_efficiency	*Detection efficiency for directional Oregon RFID antenna data*

Description

Determines detection efficiency for each antenna in systems where multiple antennas are used along a linear migration route.

Usage

```
ant_efficiency(x, LOC_vec)
```

Arguments

x	data frame generated using [join_multireader_data](#).
LOC_vec	vector of antenna locations from first encountered to last encountered.

Details

ant_efficiency determines the detection efficiency of each antenna in a linear migration route. Direction is determined based on the order of locations from first encountered to last encountered, as specified in *LOC_vec*. Use [site_summary](#) to identify all locations present in the multi reader data, which must be included in *LOC_vec*.

Antenna efficiency is determined by identifying which tags were detected at antenna x and which tags were detected anywhere after/above antenna x. The efficiency of antenna x is then the number of shared tag detections divided by the total number of detections after x. Note that efficiency and shared detections cannot be determined for the final antenna as there are no subsequent detections. Reversing the order of *LOC_vec* can inform efficiency in systems with movement in multiple directions.

Value

Returns a tibble object.

Author(s)

Annika Putt <annika@instream.net>

See Also

`import_ORFID` for importing data files from Oregon RFID ORMR and ORSR antenna readers.

`join_multireader_data` for combining data from Oregon RFID ORMR and ORSR antenna readers into a multi-reader array.

`site_summary` for identifying all locations present in a multi-reader array.

Examples

```
# Create a list containing compiled reader data:
readers <- list(reader_us, reader_ds)

# Join data into a multi-reader array:
PIT_data <- join_multireader_data(readers)

# List readers:
unique(PIT_data$LOC)

# Determine antenna efficiency for animals moving from downstream to upstream:
ant_efficiency(PIT_data, c("downstream_A1", "upstream_A1"))

# Determine antenna efficiency for animals moving from upstream to downstream:
ant_efficiency(PIT_data, c("upstream_A1", "downstream_A1"))
```

---

direction_summary   *Summary of movement direction for Oregon RFID antenna data*

Description

Summarizes the time difference between the first and last movement directions for each unique tag ID. Input data are created by `tag_direction`, which determines the direction of movement for each detection event in `join_multireader_data`.

Usage

```
direction_summary(dir_df, include_stationary = FALSE)
```

Arguments

dir_df   output from `tag_direction`.

include_stationary

                if *TRUE*, all detections are summarized. If FALSE, only detections with a known
                movement direction (up or down) are included in the summary.

Details

`direction_summary` isolates the first and last direction of movement from `tag_direction` and
determines the time difference in seconds and days. Directions are U for upstream movement, D
for downstream movement, and S for no movement, or consecutive detection at the same location.

It is common for a tag to be detected multiple times at the same antenna, which will result in a
movement direction of S, or stationary. The user is encouraged to examine direction summaries for
include_stationary = TRUE and *include_stationary = FALSE* to become familiar with their data.
When *include_stationary = FALSE*, there may be fewer tag ids in the direction summary than in the
full data set.

Value

Returns a tibble object.

Author(s)

Annika Putt <annika@instream.net>

See Also

`import_ORFID` for importing data files from Oregon RFID ORMR and ORSR antenna readers.

`join_multireader_data` for combining data from Oregon RFID ORMR and ORSR antenna read-
ers into a multi-reader array.

`tag_direction` for determining movement direction of detections in systems with a linear migra-
tion route.

Examples

```
# Create a list containing compiled reader data:
readers <- list(reader_us, reader_ds)

# Join data into a multi-reader array:
PIT_data <- join_multireader_data(readers)

# List readers:
unique(PIT_data$LOC)

# Determine tag direction for animals moving from downstream to upstream:
```

```
dir <- tag_direction(PIT_data, c("downstream_A1", "upstream_A1"))

# Determine the time difference between first and last detections:
dir_summary <- direction_summary(dir)
```

---

export_ORFID	*Export compiled data from Oregon RFID antenna readers*

Description

Data compiled using [import_ORFID](#) or [join_multireader_data](#) are exported to the working directory as either a .csv or .xlsx file

Usage

```
export_ORFID(x, name, extension)
```

Arguments

x	data frame to be exported.
name	file name.
extension	file extension (".csv" or ".xlsx").

Details

Any data frame created using functions in the **ORFID** package can be exported using this function.

Value

A file is saved to the working directory.

Author(s)

Hugo Marques <biohmarques@gmail.com>

See Also

[import_ORFID](#) for importing data files from Oregon RFID ORMR and ORSR antenna readers.

[join_multireader_data](#) for combining data from Oregon RFID ORMR and ORSR antenna readers into a multi-reader array.

`field_names`	*View field names from Oregon RFID (radio-frequency identification) antenna readers*

Description

Function to return field names and descriptions from Oregon RFID single readers (ORSR) and multi-readers (ORMR).

Usage

```
field_names()
```

Details

Returns the field/column names and its details from ORSR and ORMR data, which can be used to determine which field names should be included in data downloads.

Value

A data frame with two variables: name and details

Author(s)

Hugo Marques <biohmarques@gmail.com>

Examples

```
field_names()
```

---

import_old_readers	*Import from previous generations of Oregon RFID antenna readers*

Description

Only data collected using Oregon RFID (radio-frequency identification) ORMR (Oregon RFID Multi-Reader) and ORSR (Oregon RFID Single Reader) antenna readers can be imported using `import_ORFID`. Data from previous generations of readers must be imported using `import_old_readers`. Only detections are retained during compilation (events are removed).

This function will only work with raw data downloaded directly from Oregon RFID stationary readers. The files must be delimited (tab, comma, or semicolon) and unedited by the user.

Usage

```
import_old_readers(file, delim, verbose = FALSE)
```

Arguments

`file`	.txt file generated by an Oregon RFID reader.
`delim`	field/column delimiter, which must be tab ('\t'), comma (',') or semi-colon (';').
`verbose`	If *TRUE*, a description of the compiled data is printed to the console.

Details

The field/column delimiter must be tab, comma or semi-colon for data compilation and further analysis. The function cannot be used for space delimited data.

Data compiled using [import_ORFID](#) and [import_old_readers](#) can be joined together using [join_multireader_data](#).

Note that corruption may occur in reader data files. Check your data files and compiled data carefully to ensure accuracy.

Value

Returns a tibble object.

Author(s)

Hugo Marques <biohmarques@gmail.com>

Examples

```
# Import a single comma-deliminated data file from an old ORFID reader
import_old_readers(file = system.file("extdata/orfid_old_data_file.txt",
package = "ORFID"), delim = ",")
```

---

import_ORFID   *Import data files from Oregon RFID antenna readers*

Description

Imports files from Oregon RFID (radio-frequency identification) ORMR (Oregon RFID Multi-Reader) and ORSR (Oregon RFID Single Reader) antenna readers. Only detections are retained during compilation (events are removed).

This function will only work with raw data downloaded directly from Oregon RFID stationary readers. The files must be delimited (tab, comma, or semicolon) and unedited by the user.

Usage

```
import_ORFID(file, delim, verbose = FALSE)
```

Arguments

`file`	.txt file generated by an Oregon RFID reader.
`delim`	field/column delimiter, which must be tab ('\t'), comma (',') or semi-colon (';').
`verbose`	If *TRUE*, a description of the compiled data is printed to the console.

Details

The field/column delimiter must be tab, comma or semi-colon for data compilation and further analysis. The function cannot be used for space delimited data.

The tag number column (*TAG*) is required for subsequent analyses, and the function will return a warning if *TAG* is not included in the data file.

Note that corruption may occur in reader data files. Check your data files and compiled data carefully to ensure accuracy.

Value

Returns a tibble object of data compiled from a single ORFID data file.

Author(s)

Hugo Marques <biohmarques@gmail.com>, Annika Putt <annika@instream.net>

Examples

```
# Import a single tab-deliminated data file from an ORFID reader
import_ORFID(file = system.file("extdata/orfid_data_file.txt", package = "ORFID"), delim = "\t")
```

---

import_ORFID_events   *Import event records from Oregon RFID antenna readers*

Description

Imports files from Oregon RFID (radio-frequency identification) ORMR (Oregon RFID Multi-Reader) and ORSR (Oregon RFID Single Reader) antenna readers. Only event records are retained during compilation (detections are removed). The function will not work with older generations of Oregon RFID antenna readers.

This function will only work with raw data downloaded directly from Oregon RFID stationary readers. The files must be delimited (tab, comma, or semicolon) and unedited by the user.

Usage

```
import_ORFID_events(file, delim, verbose = FALSE)
```

Arguments

`file`	.txt file generated by an Oregon RFID reader.
`delim`	field/column delimiter, which must be tab ('\t'), comma (',') or semi-colon (';').
`verbose`	If *TRUE*, a description of the compiled data is printed to the console.

Details

The field/column delimiter must be tab, comma or semi-colon for data compilation. The function cannot be used for space delimited data.

Value

Returns a tibble object.

Author(s)

Hugo Marques <biohmarques@gmail.com>

Examples

```
# Importing event records from a single ORFID reader
import_ORFID_events(file = system.file("extdata/orfid_data_file.txt",
package = "ORFID"), delim = "\t")
```

---

join_multireader_data   *Combine data from multiple antenna readers*

Description

Function to combine data from unique readers into an array, using reader data compiled using [import_ORFID](#).

Usage

```
join_multireader_data(x, verbose = FALSE)
```

Arguments

`x`	A list object containing data frames compiled using [import_ORFID](#).
`verbose`	If *TRUE*, a description of the compiled data is printed to the console.

Details

As in [bind_rows](#), the output of [join_multireader_data](#) will contain a column if that column appears in any of the data inputs. The function creates a unique factor variable, *LOC* (location), which is a combination of *SCD* (site code) and *ANT* (antenna).

Value

Returns a tibble object of distinct detections from multiple antenna readers.

Author(s)

Hugo Marques <biohmarques@gmail.com>

See Also

import_ORFID for importing data files from Oregon RFID ORMR and ORSR antenna readers.

Examples

```
# Create a list containing compiled reader data:
readers <- list(reader_1, reader_2)

# Join data into a multi-reader array:
PIT_data <- join_multireader_data(readers)
```

---

marker_tag   *Summarize marker tag detections from Oregon RFID antenna readers*

Description

Creates a filtered dataset containing only detection data from a user-specified marker tag (unique tag number).

Usage

```
marker_tag(x, tag, gap)
```

Arguments

x	antenna data compiled using import_ORFID or join_multireader_data.
tag	marker tag identification (character object).
gap	minimum time gap in seconds between detections (optional).

Details

A column, *GAP*, is created that calculates the time gap (in seconds) between subsequent detections. If *GAP* is omitted, all detection from the marker tag are retained. If a minimum time gap is specified, only detections with a larger time gap than the specified minimum are retained. This allows the user to identify periods when marker tags were not being detected as frequently as expected.

Value

Returns a tibble object.

Author(s)

Hugo Marques <biohmarques@gmail.com>

See Also

[import_ORFID](#) for importing data files from Oregon RFID ORMR and ORSR antenna readers.

[join_multireader_data](#) for combining data from Oregon RFID ORMR and ORSR antenna readers into a multi-reader array.

Examples

```
# Create a list containing compiled reader data:
readers <- list(reader_1, reader_2, reader_3)

# Join data into a multi-reader array:
PIT_data <- join_multireader_data(readers)

# Summarize marker tag data
marker_tag(PIT_data, "0000_000000004978")

# Summarize marker tag data when the time gap between detections was greater or equal to 10 minutes.
marker_tag(PIT_data, "0000_000000004978", gap = 600)
```

---

marker_tag_plot   *Plot marker tag detections from Oregon RFID antenna readers*

Description

Creates a time series plot containing only detection data from a user-specified marker tag (unique tag number).

Usage

```
marker_tag_plot(x, tag, gap)
```

Arguments

x	antenna data compiled using [import_ORFID](#) or [join_multireader_data](#).
tag	marker tag identification (character object).
gap	minimum time gap in seconds between detections (optional).

Details

Creates a plot object displaying marker tag detections. If a minimum time gap is specified, time gaps greater than the minimum specified are highlighted in red. This allows the user to identify periods when marker tags were not being detected as frequently as expected.

Value

Returns a **ggplot2** object. If the plot is saved as a named object, **ggplot2** functions, including [theme](#) commands can be used to customize plot aesthetics, including axis labels, grid lines, etc.

Author(s)

Hugo Marques <biohmarques@gmail.com>

See Also

[import_ORFID](#) for importing data files from Oregon RFID ORMR and ORSR antenna readers. [join_multireader_data](#) for combining data from Oregon RFID ORMR and ORSR antenna readers into a multi-reader array.

Examples

```
# Create a list containing compiled reader data:
readers <- list(reader_1, reader_2, reader_3)

# Join data into a multi-reader array:
PIT_data <- join_multireader_data(readers)

# Plot marker tag data and highlight gaps greater than 10 minutes.
marker_tag_plot(PIT_data, "0000_000000004978", gap = 600)
```

---

reader_1   *ORFID data samples*

Description

Compiled data to illustrate functions from **ORFID**. The variables are as follows:

Usage

```
reader_1
```

Format

A data frame with 8590 rows and 9 variables:

DTY Detection type (S = summary, I = individual, E = event)

ARR Arrival date and time

TRF Time reference (G = GNSS, N = network, U = unreferenced)

DUR Duration of the period between the arrival and the departure from the detection zone

TTY Tag type A=Animal (R = Read only, W = Writeable, P = Phantom)

TAG Tag ID number

SCD Site code

NCD Number of consecutive detections

EFA Effective amps

---

reader_2 *ORFID data samples*

Description

Compiled data to illustrate functions from **ORFID**. The variables are as follows:

Usage

```
reader_2
```

Format

A data frame with 8590 rows and 9 variables:

DTY Detection type (S = summary, I = individual, E = event)

ARR Arrival date and time

TRF Time reference (G = GNSS, N = network, U = unreferenced)

DUR Duration of the period between the arrival and the departure from the detection zone

TTY Tag type A=Animal (R = Read only, W = Writeable, P = Phantom)

TAG Tag ID number

SCD Site code

NCD Number of consecutive detections

EFA Effective amps

---

reader_3 *ORFID data samples*

Description

Compiled data to illustrate functions from **ORFID**. The variables are as follows:

Usage

```
reader_3
```

Format

A data frame with 1034 rows and 9 variables:

DTY Detection type (S = summary, I = individual, E = event)
ARR Arrival date and time
TRF Time reference (G = GNSS, N = network, U = unreferenced)
DUR Duration of the period between the arrival and the departure from the detection zone
TTY Tag type A=Animal (R = Read only, W = Writeable, P = Phantom)
TAG Tag ID number
SCD Site code
NCD Number of consecutive detections
EFA Effective amps

---

reader_ds   *ORFID data samples*

Description

Compiled data to illustrate directional functions from **ORFID**. The variables are as follows:

Usage

reader_ds

Format

A data frame with 824 rows and 16 variables:

DTY Detection type (S = summary, I = individual, E = event)
ARR Arrival date and time
TRF Time reference (G = GNSS, N = network, U = unreferenced)
DUR Duration of the period between the arrival and the departure from the detection zone
TTY Tag type A=Animal (R = Read only, W = Writeable, P = Phantom)
TAG Tag ID number
SCD Site code
NCD Number of consecutive detections
EFA Effective amps
TCH Tag technology (HDX, FDX, HF)
ANT Antenna number
EMP Number of empty scans preceding detection
TSS Tag signal strength
SPV Unknown
NOI Noise
CLS Tag class

---

reader_us   *ORFID data samples*

Description

Compiled data to illustrate directional functions from **ORFID**. The variables are as follows:

Usage

```
reader_us
```

Format

A data frame with 781 rows and 16 variables:

DTY Detection type (S = summary, I = individual, E = event)

ARR Arrival date and time

TRF Time reference (G = GNSS, N = network, U = unreferenced)

DUR Duration of the period between the arrival and the departure from the detection zone

TTY Tag type A=Animal (R = Read only, W = Writeable, P = Phantom)

TAG Tag ID number

SCD Site code

NCD Number of consecutive detections

EFA Effective amps

TCH Tag technology (HDX, FDX, HF)

ANT Antenna number

EMP Number of empty scans preceding detection

TSS Tag signal strength

SPV Unknown

NOI Noise

CLS Tag class

---

site_summary   *Summarize site information from Oregon RFID antenna reader data*

Description

Summarizes detection information for unique antenna sites within antenna reader data compiled using `import_ORFID` or `join_multireader_data`.

Usage

```
site_summary(x)
```

Arguments

x   antenna data compiled using `import_ORFID` or `join_multireader_data`.

Details

Creates a tibble grouped by *SCD* (site code; one row per unique *SCD*). The data frame contains the site code (*SCD*), the total number of records detected (*REC*), the number of unique tags detected (*TAG_ID*), and the time at which the first (*FIR*) and last (*LAS*) detections occurred on the array.

Value

Returns a tibble object.

Author(s)

Hugo Marques <biohmarques@gmail.com>

See Also

`import_ORFID` for importing data files from Oregon RFID ORMR and ORSR antenna readers.

`join_multireader_data` for combining data from Oregon RFID ORMR and ORSR antenna readers into a multi-reader array.

Examples

```
# Create a list containing compiled reader data:
readers <- list(reader_1, reader_2)

# Join data into a multi-reader array:
PIT_data <- join_multireader_data(readers)

# Summarize detection information for each unique site:
site_summary(PIT_data)
```

---

start_stop_plot   *Plot reader start and stop times from Oregon RFID antenna readers*

Description

Creates a time series plot containing start and stop times from Oregon RFID (radio-frequency identification) ORMR (Oregon RFID multi-reader) and ORSR (Oregon RFID single reader) antenna readers.

Usage

```
start_stop_plot(x)
```

Arguments

x   event data compiled using [import_ORFID_events](#)

Details

Creates a plot displaying reader start and stop times. Note that start and stop times can be very close together and difficult to distinguish depending on the period plotted. Filter event data to improve resolution.

Value

Returns a plot object.

Author(s)

Annika Putt <annika@instream.net>

See Also

[import_ORFID_events](#) for importing event data from Oregon RFID ORMR and ORSR antenna readers.

[join_multireader_data](#) for combining data from Oregon RFID ORMR and ORSR antenna readers into a multi-reader array.

---

tag_direction   *Movement direction for Oregon RFID antenna data*

Description

Determines direction of tag movement in systems where multiple antennas are used along a linear migration route.

Usage

```
tag_direction(x, LOC_vec)
```

Arguments

x   data frame generated using [join_multireader_data](#).

LOC_vec   vector of antenna locations from first encountered to last encountered.

Details

tag_direction determines the direction of movement for individual detection events in *x*. Direction is determined based on the order of locations from first encountered to last encountered, as specified in *LOC_vec*. Note that direction cannot be determined until the tag has been detected at multiple locations. Use [site_summary](#) to identify all locations present in the multi-reader data, which must be included in *LOC_vec*.

Value

Returns a tibble object. The column *DIR* displays direction, where *U* is upstream movement, *D* is downstream movement, and *S* is no movement, or a consecutive detection at the previous location.

Author(s)

Annika Putt <annika@instream.net>

See Also

[import_ORFID](#) for importing data files from Oregon RFID ORMR and ORSR antenna readers.

[join_multireader_data](#) for combining data from Oregon RFID ORMR and ORSR antenna readers into a multi-reader array.

[site_summary](#) for identifying all locations present in a multi reader array

Examples

```
# Create a list containing compiled reader data:
readers <- list(reader_us, reader_ds)

# Join data into a multi-reader array:
PIT_data <- join_multireader_data(readers)

# List readers:
unique(PIT_data$LOC)

# Determine tag direction for animals moving from downstream to upstream:
tag_direction(PIT_data, c("downstream_A1", "upstream_A1"))
```

---

tag_summary   *Summarize tags detected by Oregon RFID antenna readers*

Description

Summarizes detection information for unique tags within antenna reader data compiled using import_ORFID
or join_multireader_data.

Usage

```
tag_summary(x, verbose = TRUE)
```

Arguments

x	antenna data compiled using import_ORFID or join_multireader_data.
verbose	If *TRUE*, a data frame describing output columns is printed to the console.

Details

Creates a tibble grouped by TAG (one row per unique TAG). A data frame describing the summarized data is printed to the console.

Value

Returns a tibble object.

Author(s)

Hugo Marques <biohmarques@gmail.com>

See Also

import_ORFID for importing data files from Oregon RFID ORMR and ORSR antenna readers.

join_multireader_data for combining data from Oregon RFID ORMR and ORSR antenna readers into a multi-reader array.

Examples

```
# Create a list containing compiled reader data:
readers <- list(reader_1, reader_2)

# Join data into a multi-reader array:
PIT_data <- join_multireader_data(readers)

# Summarize detection information for each unique tag:
tag_summary(PIT_data)
```

Index