Package 'Rraven'

November 7, 2017

Type Package

Title connecting between R and Raven bioacoustic software

Version 1.0.0

Date 2017-04-28

Author Marcelo Araya-Salas

Maintainer Marcelo Araya-Salas <araya-salas@cornell.edu>

Description A tool to exchange data between R and Raven bioacoustic software (Cornell Lab of Ornithology). Functions work on data formats compatible with the R package warbleR.

License GPL (>= 2)

Imports pbapply, warbleR, utils, stats

Depends R (>= 3.2.1)

LazyData TRUE

URL https://github.com/maRce10/Rraven

BugReports https://github.com/maRce10/Rraven/issues

NeedsCompilation no

Suggests knitr

VignetteBuilder knitr

RoxygenNote 5.0.1

Repository CRAN

Date/Publication 2017-11-10 08:12:11

R topics documented:

---

exp_raven   *Export raven selections*

Description

exp_raven exports selection tables as Raven selection data in .txt format.

Usage

```
exp_raven(X, path = NULL, file.name = NULL, khz.to.hz = TRUE,
sound.file.path = NULL, single.file = TRUE)
```

Arguments

X	Data frame containing columns for sound file (sound.files), selection (selec), start and end time of signals ('start' and 'end') and low and high frequency ('low.freq' and 'high.freq', optional). See example data 'selec.table' in the [warbleR](#)) package.
path	A character string indicating the path of the directory in which to save the selection files. If not provided (default) the function saves the file into the current working directory.
file.name	Name of the output .txt file. If NULL then the sound file names are used instead. If multiple selection files are generated (see 'single.file') then the sound files names are added to the provided 'file.name'.
khz.to.hz	Logical. Controls if frequency variables should be converted from kHz (the unit used by other bioacoustic analysis R packages like [warbleR](#)) to Hz (the unit used by Raven). Default is TRUE.
sound.file.path	
	A character string indicating the path of the directory containing the sound file(s). Providing this information allows to open both sound file and selection table simultaneously. This can be done by using the 'File > Open selection table' option in Raven (or drag/drop the selection file into Raven). Default is NULL. This argument is required when exporting selections from multiple sound files.
single.file	Logical. Controls whether a single selection file (TRUE; default) or multiple selection files for each sound files (FALSE, hence, only applicable when several sound files are included in 'X') are generated. Note that 'sound.file.path' must be provided when exporting several sound files into a single selection file as the duration of the sound files is required.

Details

The function exports selection tables as the ones used by the bioacoustic analysis R package warbleR to Raven selection files in '.txt' format. This can be useful to obtain additional Raven measurements on existing selections by adding new measurements to the selection table once in Raven. Note that selection labels must be numeric and unduplicated when exporting them to Raven. If that is not the case the function will relabeled the selections and the previous selection labels will be retained in a new column('old.selec').

Value

The function saves a selection table in '.txt' format that can be directly opened in Raven. If several sound files are available users can either export them as a single selection file or as multiple selection files (one for each sound file). No objects are returned in the R environment.

Author(s)

Marcelo Araya-Salas (<araya-salas@cornell.edu>)

See Also

imp_raven; imp_syrinx

Examples

```
## Not run:
# First set temporary folder
setwd(tempdir())

# Load data
data(list = c("Phae.long1", "Phae.long2", "Phae.long3", "Phae.long4", "selec.table"))

# Select data for a single sound file
st1 <- selec.table[selec.table$sound.files == "Phae.long1.wav",]

# Export data of a single sound file
exp_raven(st1, file.name = "Phaethornis Rraven examples")

writeWave(Phae.long1, "Phae.long1.wav") #save sound files
writeWave(Phae.long2, "Phae.long2.wav")
writeWave(Phae.long3, "Phae.long3.wav")
writeWave(Phae.long4, "Phae.long4.wav")

exp_raven(X = selec.table, file.name = "Phaethornis warbleR examples",
sound.file.path = tempdir(), single.file = T)

## End(Not run)
```

---

extract_ts   *Extract time series parameters from data imported from Raven*

Description

extract_ts extracts time series parameters from data imported from Raven bioacoustic software.

Usage

```
extract_ts(X, ts.column, equal.length = FALSE, as.time.series = FALSE)
```

Arguments

X	Data frame imported from Raven. It should include at least columns for: sound file names, selection labels, a parameters encoded as a time series (e.g. several numbers separated by semicolon)
ts.column	Name of the column with the time series data to be extracted. Default is NULL.
equal.length	Logical. Controls whether time series are kept as in the original data (most of the time with unequal lengths) or numbers are interpolated to equalize series length (using the [approx](#) function). All series will be interpolated to match the length of the longest series in the data. Default is FALSE.
as.time.series	Logical. Controls if data is converted to the time series format (using the [as.ts](#) function). Default is FALSE.

Details

The function extracts parameters enconded as time series in Raven selection files. The resulting data frame can be directly input into functions for time series analysis of acoustic signals as [dfDTW](#).

Value

A data frame with columns for sound file name (sound.files), selection label (selec) and the time series for each selection.

Author(s)

Marcelo Araya-Salas (<araya-salas@cornell.edu>)

See Also

[imp_raven](#); [exp_raven](#)

Examples

```
## Not run:
# Load data
data("selection_file_ts")

# freq contour 95 dif lengths
extract_ts(X = selection_file_ts, ts.column = "Freq.Contour.95...Hz.")

# freq contour 95 equal lengths
extract_ts(X = selection_file_ts, ts.column = "Freq.Contour.95...Hz.", equal.length = T)

# freq contour 95 equal lengths
extract_ts(X = selection_file_ts, ts.column = "Peak.Freq.Contour..Hz.", equal.length = T)

## End(Not run)
```

---

imp_raven   *Import Raven selections*

Description

imp_raven imports Raven selection files simultaneously from many files. Files must be in '.txt' format.

Usage

```
imp_raven(path = NULL, sound.file.col = NULL, all.data = FALSE, recursive = FALSE,
 name.from.file = FALSE, ext.case = NULL, freq.cols = TRUE, waveform = FALSE)
```

Arguments

path   A character string indicating the path of the directory in which to look for the Raven selection (text) files. If not provided (default) the function searches into the current working directory.

sound.file.col A character string with the name of the column containing the sound files in the selection text files. Default is NULL. If provided, the output data frame will contained all columns needed for subsequent analysis in the acoustic analysis package warbleR. Duplicated rows, as when "waveform" and "spectrogram" information are included for the same selection, will be removed. All selection files must contain "Selection", "Begin.Time" and "End.Time" columns.

all.data   Logical. If TRUE all columns in the selection files are returned, keeping the name columns as in the raven files. Default is FALSE. Columns absent in some selection files will be filled with NA's.

recursive   Logical. If TRUE the listing recurse into sub-directories.

name.from.file Logical. If TRUE the sound file names are extracted from the selection text file
                name. It asssumes that selections files contained the suffix "Table.1.selections.txt"
                or "selections.txt". Note that by default it will assume that the extension file
                name is ".wav". This can be control using the argumet 'ext.wav'. Default is
                FALSE). Ignored if sound.file.col' is provided and/or all.data is TRUE).

ext.case   Character string of length 1 to specify whether sound file extensions are in upper
                or lower case. This should match the extension of the of the .wav files from
                which the selection were made. It must be either 'upper' or 'lower'. Only
                needed when 'name.from.file' is TRUE. Ignored if 'sound.file.col' is provided
                and/or all.data is TRUE.

freq.cols   Logical. If TRUE 'Low Freq' and 'High Freq' columns are also imported. Ig-
                nored if all.data is TRUE.

waveform   Logical to control if waveform view data should be included (this data is typ-
                ically duplicated in spectrogram view data). Default is FALSE (not to include
                it).

Details

The function import raven selection data from many files simultaneously. Files must be in .txt
format. Selection files including data from mulitple recordings can also be imported.

Value

A single data frame with information of the selection files. If 'all.data' argument is set to FALSE
the data frame contains the following columns: selec, start, end, and selec.file. If sound.file.col is
provided the data frame will also contain a 'sound.files' column. In addition, all rows with dupli-
cated data are removed. This is useful when both spectrogram and waveform views are included in
the Raven selection files. If all.data is set to TRUE then all columns in the Raven selection files are
returned.

Author(s)

Marcelo Araya-Salas (<araya-salas@cornell.edu>)

See Also

imp_syrinx

Examples

```
## Not run:
# First set temporary folder
setwd(tempdir())

data(selection.files)

write.table(selection.files[[1]],file = "100889-Garrulax monileger.selections.txt",
row.names = FALSE, sep= '\t')
```

```
write.table(selection.files[[2]],file = "1023-Arremonops rufivirgatus.selections.txt",
row.names = FALSE, sep= '\t')

 #providing the name of the column with the sound file names
rav.dat <- imp_raven(sound.file.col = "End.File", all.data = FALSE)

View(rav.dat)

#getting all the data
rav.dat2 <- imp_raven(all.data = TRUE)
View(rav.dat2)

## End(Not run)
```

---

imp_syrinx   *Import Syrinx selections*

Description

`imp_syrinx` imports Syrinx selection data from many files simultaneously. All files must be have the same columns.

Usage

```
imp_syrinx(path = NULL, all.data = FALSE, recursive = FALSE,
exclude = FALSE, hz.to.khz = TRUE)
```

Arguments

path	A character string indicating the path of the directory in which to look for the text files. If not provided (default) the function searches into the current working directory. Default is `NULL`.
all.data	Logical. If `TRUE` all columns in text files are returned. Default is `FALSE`. Note that all files should contain exactly the same columns in the same order.
recursive	Logical. If `TRUE` the listing recurse into sub-directories.
exclude	Logical. Controls whether files that cannot be read are ignored (`TRUE`). Default is `FALSE`.
hz.to.khz	Logical. Controls if frequency variables should be converted from Hz (the unit used by Syrinx) to kHz (the unit used by warbleR and other bioacoustic analyssis packages in R). Default if `TRUE`. Ignored if all.data is `TRUE`.

Value

A single data frame with information of the selection files. If all.data argument is set to `FALSE` the data frame contains the following columns: selec, start, end, and selec.file. If sound.file.col is provided the data frame will also contain a 'sound.files' column. If all.data is set to `TRUE` then all columns in selection files are returned.

Author(s)

Marcelo Araya-Salas (<araya-salas@cornell.edu>)

See Also

[imp_raven](#)

Examples

```
## Not run:
# First set temporary folder
setwd(tempdir())

#load data
data(selection.files)

write.table(selection.files[[3]],file = "harpyeagle.wav.txt",row.names = FALSE,
 col.names = FALSE, sep= "\t")

write.table(selection.files[[4]],file = "Phae.long4.wav.txt",row.names = FALSE,
col.names = FALSE, sep= "\t")

syr.dat <- imp_syrinx(all.data = FALSE)

View(syr.dat)

#getting all the data
syr.dat <- imp_syrinx(all.data = TRUE)

View(syr.dat)

## End(Not run)
```

---

relabel_cols   *Relabel columns to match the selection table format*

Description

relabel_cols relabels columns to match the selection table format (as in the R package [warbleR](#))

Usage

```
relabel_cols(X, extra.cols.name = NULL, extra.cols.new.name = NULL, khz.to.hz = FALSE,
waveform = FALSE)
```

Arguments

X	Data frame imported from Raven.
extra.cols.name	
	Character vector with the names of additional columns to be relabeled. Default is NULL. 'extra.cols.new.name' must be also provided.
extra.cols.new.name	
	Character vector with the new names for the additional columns to be relabeled. Default is NULL. 'extra.cols.name' must be also provided.
khz.to.hz	Logical. Controls if frequency variables ('high.freq' and 'low.freq') should be converted from kHz (the unit used by other bioacoustic analysis R packages like [warbleR](#)) to Hz (the unit used by Raven). Default is TRUE.
waveform	Logical to control if 'waveform' related data should be included (this data is typically duplicated in 'spectrogram' data). Default is FALSE (not to include it).

Details

This function relabels columns to match the selection table format to match then ones used by other bioacoustic analysis R packages like [warbleR](#).

Value

The function returns the input data frame with new column names for time and frequency 'coordinates' and sound files and selections.

Author(s)

Marcelo Araya-Salas (<araya-salas@cornell.edu>)

See Also

[imp_raven](#); [exp_raven](#)

Examples

```
## Not run:
 data("selection_file_ts")

# Select data for a single sound file
rcdf1 <- relabel_cols()

# plus 1 additional column
rcdf2 <- relabel_cols(selection_file_ts, extra.cols.name = "selec.file", "Raven selection file")

# plus 2 additional column
rcdf3 <- relabel_cols(selection_file_ts, extra.cols.name = c("selec.file", "View"),
 c("Raven selection file", "Raven view"))

## End(Not run)
```

---

run_raven   *Open sound files in Raven sound analysis software*

Description

run_raven opens several sound files in Raven sound analysis software

Usage

```
run_raven(raven.path = NULL, sound.files = NULL, path = NULL, at.the.time = 10,
import = FALSE, redo = FALSE, ...)
```

Arguments

raven.path	A character string indicating the path of the directory in which to look for the raven executable file (where Raven was installed).
sound.files	character vector indicating the files that will be analyzed. If NULL (default) then Raven will be run without opening any file.
path	A character string indicating the path of the directory in which to look for the sound files. If not provided (default) the function searches into the current working directory. Default is NULL.
at.the.time	Numeric vector of length 1 controling how many files will be open in Raven at the same time. Note that opening too many files at once could make Raven run out of memory. You need to close Raven every time the batch of files is analyzed, so the next batch is opened. Default is 10.
import	Logical. Controls if the selection tables generated should be returned as a data frame into the R environment. This only works if the selections are saved in the "Selections" folder in the Raven directory. This argument calls the [imp_raven](#) internally. Additional arguments can be passed to [imp_raven](#) to control the way the data is imported.
redo	Logical. Controls whether only the subset of files with no Raven selections (.txt file) in the Raven 'selections' folder are analyzed. Useful when resuming the analysis. Default is FALSE.
...	Additional arguments to be passed to [imp_raven](#) for customizing how selections are imported (ignored if import = FALSE).

Details

The function runs Raven bioacoustics analysis software (Cornell Lab of Ornithology), opening many files simultaneously. Raven will still run if no sound files are provided (i.e. sound.files = NULL). At the end of the analysis the data can be automatically imported back into R using the 'import' argument.

Value

If import = TRUE a data frame with the selections produced during the analysis will be return as an data frame. See [imp_raven](#) for more details on how selections are imported.

Author(s)

Marcelo Araya-Salas (<araya-salas@cornell.edu>)

See Also

[imp_raven](); [imp_syrinx](); [run_raven]()

Examples

```
## Not run:
# First set temporary folder
setwd(tempdir())

# save sound files
data(list = c("Phae.long1", "Phae.long2"))
writeWave(Phae.long1, "Phae.long1.wav", extensible = FALSE)
writeWave(Phae.long2, "Phae.long2.wav", extensible = FALSE)

raven.path <- "PATH_TO_RAVEN_DIRECTORY_HERE"

# run function
run_raven(raven.path = raven.path, sound.files = c("Phae.long1.wav", "Phae.long2.wav"),
 at.the.time = 2, import = T, name.from.file = T, ext.case = "upper", all.data = T)

#getting all the data
rav.dat<-run_raven(all.data = TRUE)
View(rav.dat)

# run function on all the wav files in the working directory
run_raven(raven.path = raven.path, sound.files = list.files(pattern = "\.wav$",
ignore.case = TRUE), at.the.time = 4, import = FALSE)


## End(Not run)
```

---

selection_file_ts	*Data frame of selections imported from Raven including a time series parameter.*

Description

A data frame containing the acoustic parameters measured on *Phaethornis longirostris* (Long-billed Hermit) songs from the example sound files included in the [warbleR]() package.

Usage

```
data(selection_file_ts)
```

Format

A data frame with 3 rows and 14 variables

Source

Marcelo Araya-Salas, warbleR

Index