agasc / **Soundscape-analysis-with-R** Public

Code Issues Pull requests Actions Projects Wiki Security Insights

master ▾ ···

agasc ... on 22 Nov 2018

View code

Soundscape-analysis-with-R

Acoustic diversity indices.r

AcouIndexAlpha(wave, stereo=FALSE, min_freq = 2000, max_freq = 22000, anthro_min = 1000, anthro_max = 2000, bio_min=2000, bio_max=12000, wl=512, j=5, AcouOccupancy=TRUE, Bioac=TRUE, Hf=TRUE, Ht=TRUE, H=TRUE, ACI=TRUE, AEI_villa=TRUE, M=TRUE, NDSI=TRUE, ADI=TRUE, NP=TRUE)

This code will compute several acoustic diversity indices on a single wav file. This wav file can be stereo or mono. This function calls the packages "seewave" (Sueur et al. 2008a), "tuneR" (Ligges et al. 2016) and, "soundecology" (Villanueava-Rivera and Pijanowski 2016)

Arguments

wave

a R wave object (see the function "readWave" in the "tuneR" package)

stereo

if "FALSE" the function will consider only the left channel of a stereo wave file or the only channel of a mono wave file. If "TRUE" the function will calculate the indices on both channels of a stereo wave file separately. Set as "FALSE" by default.

min_freq

minimum frequency in hz. Set by default at 2000 Hz.

max_freq

maximum frequency in hz. Set by default at 22000 Hz.

anthro_min

minimum frequency of anthrophony used to calculate the Normalized Difference Soundscape Index (NDSI) from Kasten et al. (2012). Set by default at 1000 Hz

anthro_max

maximum frequency of anthrophony used only to calculate the Normalized Difference Soundscape Index (NDSI) from Kasten, et al. (2012). Set by default at 2000 Hz

bio_min

minimum frequency of biophony used only to calculate the Normalized Difference Soundscape Index (NDSI) from Kasten, et al. (2012). Set by default at 2000 Hz

bio_max

maximum frequency of biophony used only to calculate the Normalized Difference Soundscape Index (NDSI) from Kasten, et al. (2012). Set by default at 12000 Hz

wl

window length for the spectrogram analysis (even number of points). Set by default at 512 samples

j

temporal steps used only for the Acoustic Complexity Index (ACI) calculation from Pieretti et al. (2102). Set by default at 5 seconds

AcouOccupancy

if TRUE, will calculate the acoustic occupancy index. This index calculate the number of temporal sample in the spectrogram above a threshold of 50 DBA

Bioac

if TRUE, will calculate Bioacoustic index (Boelman et al. 2007) by calling the bioacoustic_index function from the soundecology package.

Hf

if TRUE, will calculate the frequency Entropy (Hf; Sueur et al. 2008b) by calling the "sh" function from the "seewave" package.

Ht

if TRUE, will calculate the temporal Entropy (Ht; Sueur et al. 2008b) by calling the "env" function from the "seewave" package.

H

if TRUE, will calculate the Entropy index (H; Sueur et al. 2008b) by calling the "sh" and "env" function from the "seewave" package.

ACI

if TRUE, will calculate the Acoustic Complexity Index (ACI; Pieretti et al. 2011) by calling the "acoustic_complexity" function from the "soundecology" package.,

AEI_villa

if TRUE, will calculate the Acoustic Eveness index (AEI; Villanueva-Rivera et al. 2011) by calling the "acoustic_evenness" function from the package "soundecology"

M

if TRUE, will calculate the Median of amplitude enveloppe (M; Depraetere et al. 2012) by calling the function "env" from the "seewave" package

NDSI

if TRUE, will calculate the Normalised Difference Soundscape Index (NDSI; Kasten et al. 2012) by calling the function "ndsi" from the "soundecology" package

ADI

if TRUE, will calculate the Acoustic diversity index (ADI in Pekin et al. 2013 or H' in Villanueva-Riviera et al. 2011) by calling the function "acoustic_diversity" in the "soundecology" package

NP

if TRUE, will calculate the Number of frequency Peaks (NP; Gasc et al. 2013) by calling the function "fpeaks" in the "seewave" package

Details

Value

This function will return a list of two tables. The first table called "Mono_left" is for the "left channel" or "mono channel" and the second table called "Mono_right" for the "right channel". In case of STEREO=FALSE, the second table will be filled with NA value. Column of this table will corresponds to the different indices you calculated. In case you want to measure the same set of indices on several wave files, this function can be included in a loop (see example below) and all table results can be combined in a final table with indices in column and wave files in lines.

Example

on one file

```
library(soundecology)

data(tropicalsound)

Result<-AcouIndexAlpha(tropicalsound, stereo=FALSE, min_freq = 2000, max_freq = 10000, anthro_min = 1000, anthro_max = 2000, bio_min=2000, bio_max=10000, wl=512, j=5, AcouOccupancy=TRUE, Bioac=TRUE, Hf=TRUE, Ht=TRUE, H=TRUE, ACI=TRUE, AEI_villa=TRUE, M=TRUE, NDSI=TRUE, ADI=TRUE, NP=TRUE)

Result_left<-Result$Mono_left

Result_left
```

on several files. Let's imagine you have a directory "Dir1" with several wave files (only wave files!).

```
setwd("Dir1")

WaveNames<-dir()

TableTotal<-NULL

for (i in 1: length(WaveNames))

{

wave<-readWave(WaveNames[i])

Result<-AcouIndexAlpha(wave, stereo=FALSE, min_freq = 2000, max_freq = 10000, anthro_min = 1000, anthro_max = 2000, bio_min=2000,

bio_max=12000, wl=512, j=5, AcouOccupancy=TRUE, Bioac=TRUE, Hf=TRUE, Ht=TRUE, H=TRUE, ACI=TRUE, AEI_villa=TRUE, M=TRUE, NDSI=TRUE,

ADI=TRUE, NP=TRUE)

TableTotal<-rbind(TableTotal,Result$Mono_left)
}

rownames(TableTotal)<-WaveNames

TableTotal
```

References

Boelman, N. T., G. P. Asner, P. J. Hart, and R. E. Martin (2007). Multi-trophic invasion resistance in Hawaii: Bioacoustics, field surveys, and airborne remote sensing. Ecological Applications 17:2137–2144.

Depraetere, M., S. Pavoine, F. Jiguet, A. Gasc, S. Duvail, and J. Sueur (2012). Monitoring animal diversity using acoustic indices: Implementation in a temperate woodland. Ecological Indicators 13:46–54.

Gasc, A., J. Sueur, S. Pavoine, R. Pellens, and P. Grandcolas (2013). Biodiversity sampling using a global acoustic approach: Contrasting sites with microendemics in New Caledonia. PLOS One 8:e65311. doi:10.1371/journal.pone. 0065311

README.md

Environmental Assessment Laboratory's Acoustic Library: An Archive for Studying Soundscape Ecology. Ecological Informatics 12: 50-67. doi: 10.1016/j.ecoinf.2012.08.001

Ligges,U., S. Krey, O. Mersmann, and S. Schnackenberg (2016). tuneR: Analysis of music. URL: http://r-forge.r-project.org/projects/tuner/.

Pieretti, N., A. Farina, and D. Morri (2011). A new methodology to infer the singing activity of an avian community: The Acoustic Complexity Index (ACI). Ecological Indicators 11: 868–873.

Sueur J., T. Aubin, and C. Simonis (2008a). Seewave: a free modular tool for sound analysis and synthesis. Bioacoustics, 18: 213-226

Sueur, J., S. Pavoine, O. Hamerlynck, and S. Duvail (2008b). Rapid acoustic survey for biodiversity appraisal. PLOS One 3:e4065.

Villanueva-Rivera, L.J., and B.C. Pijanowski (2016). soundecology: Soundscape Ecology. R package version 1.3.2. https://CRAN.R-project.org/package=soundecology

Villanueva-Rivera, L. J., B. C. Pijanowski, J. Doucette, and B. Pekin. 2011. A primer of acoustic analysis for landscape ecologists. Landscape Ecology 26: 1233-1246. doi: 10.1007/s10980-011-9636-9.

STLDataPrep.r

STLDataPrep(pumilio_URL,Colname,SiteNames,pathFlacFronten,Directory,DurationFile,STL_type,STLy_Hour,STLy_Nfiles,STLy_Month=NULL,STLd_Month,STLd_Ndays,STLd_Hours=NULL,FactorQuality=NULL)

The software Sonic Time Lapse Builder have been developped by Benjamin Gottesman and Mark Durham. It is creating a Audio file by concatainating several .wav files.To use Sonic Time Lapse Builder program, you will need to drag a directory containing .these files. This code will help you to prepare this directory from the Pumilio Database (see Villanueva-Rivera and Pijanowski 2012) in order to have clean time series of files for a seasonality (along month, by default it will consider all the months) or daily variation (along hour of the day, by default it will consider all hours). Author Amandine Gasc, 20160921.

Arguments

pumilio_URL

Address of the pumilio database

Colname

Character, the collection name in Pumilio example: "Arizona 2013"

SiteNames

Vector of character, names of the sites in Pumilio: c("H1-HI-R1","H1-HI-R2")

pathFlacFronten

Character, the path where you have the flac2wav software see ?wac2wav

Directory

Character, path of your temporary directory where you will have you files copied and transformed

DurationFile

Interger, the Duration final of the file in seconds, if empty you will concerve the original duration

STL_type

Character, either "STLd" for Sonic Time Lapse daily or "STLy" for sonic timelapse yearly

STLy_Hour

Character, representing ONE hours of the day, example: "00" for midnight, "09" for 9 a.m. or "22" for 10 p.m.

STLy_Nfiles

Integer, representing the number of files per month, equivalent of the number of days represented

STLy_Month==NULL

Vector of character representing the month: example: c("01","02","03") for a time lapse containing the month pf January to March. if NULL it will consider all the month from January to December.

STLd_Month

Character, representing ONE Month of the year, example: "01" for January or "10" for October

STLd_Ndays

Integer, representing the number of days, one folder will be created with the 24 hours of these days randomly sampled.

STLd_Hours==NULL

Vector of character representing the Hours: example: c("00","02","03") for a time lapse containing the Hours from midnight to 3 a.m. if NULL it will consider all the hours of the day.

FactorQuality

Vector of TRUE/FALSE follwing the filenames of the recordings to consider; For example you have information about wind, rain and/or technical issue: so information about how to exclude sounds from the selection

Example (not working for everyone, only to consult for argument formatting)

Example 1: STLy

pumilio_URL="http://soundscape01.rcac.purdue.edu/pumilio/"

Colname="Arizona 2013"

SiteNames<-c("H1-HI-R1","H1-HI-R2", "H1-HI-R3", "H1-NO-R1", "H1-NO-R2", "H1-NO-R3", "H2-HI-R1", "H2-HI-R2", "H2-HI-R3", "H2-NO-R1", "H2- NO-R2", "H2-NO-R3", "H3-LO-R1", "H3-LO-R2", "H3-LO-R3", "H3-ME-R1", "H3-ME-R2", "H3-ME-R3", "H3-NO-R1", "H3-NO-R2", "H3-NO-R3", "H4-HI-R1", "H4-HI-R2", "H4-HI-R3", "H4-NO-R1", "H4-NO-R2", "H4-NO-R3") # Set the sites names

pathFlacFronten<-"C:/Program Files (x86)/FLAC Frontend/tools"

Directory<-"C:/Users/gamandin/Desktop/DayJune_wind"

DurationFile<-10

STL_type = "STLy"

STLy_Hour = "06"

STLy_Nfiles = 5

STLy_Month<-c("03","04","05","06","07","08","09","10","11")

STLDataPrep(pumilio_URL=pumilio_URL, Colname=Colname,SiteNames=SiteNames,pathFlacFronten=pathFlacFronten,Directory=Directory,DurationFile=DurationFile,STL_type=STL_type,STLy_Hour=STLy_Hour,STLy_Nfiles=STLy_Nfiles,STLy_Month=STLy_Month)

Example 2: STLd

pumilio_URL="http://soundscape01.rcac.purdue.edu/pumilio/"

Colname="Arizona 2013"

SiteNames<-c("H1-HI-R1","H1-HI-R2", "H1-HI-R3", "H1-NO-R1", "H1-NO-R2", "H1-NO-R3", "H2-HI-R1", "H2-HI-R2", "H2-HI-R3", "H2-NO-R1", "H2-NO-R2", "H2-NO-R3", "H3-LO-R1", "H3-LO-R2", "H3-LO-R3", "H3-ME-R1", "H3-ME-R2", "H3-ME-R3", "H3-NO-R1", "H3-NO-R2", "H3-NO-R3", "H4-HI-R1", "H4-HI-R2", "H4-HI-R3", "H4-NO-R1", "H4-NO-R2", "H4-NO-R3") # Set the sites names

pathFlacFronten<-"C:/Program Files (x86)/FLAC Frontend/tools"

Directory<-"C:/Users/gamandin/Desktop/test"

DurationFile<-10

STL_type = "STLd"

STLd_Month="06"

STLd_Ndays=3

STLd_Hours=c("00","01","02","03","04")

STLDataPrep(pumilio_URL=pumilio_URL,Colname=Colname,SiteNames=SiteNames,pathFlacFronten=pathFlacFronten,Directory=Directory,DurationFile=DurationFile,STL_type=STL_type,STLd_Month=STLd_Month,STLd_Ndays=STLd_Ndays,STLd_Hours=STLd_Hours)

Example 3: STLd

pumilio_URL="http://soundscape01.rcac.purdue.edu/pumilio/"

Colname="Arizona 2013"

SiteNames<-c("H1-HI-R1","H1-HI-R2", "H1-HI-R3", "H1-NO-R1", "H1-NO-R2", "H1-NO-R3", "H2-HI-R1", "H2-HI-R2", "H2-HI-R3", "H2-NO-R1", "H2-NO-R2", "H2-NO-R3", "H3-LO-R1", "H3-LO-R2", "H3-LO-R3", "H3-ME-R1", "H3-ME-R2", "H3-ME-R3", "H3-NO-R1", "H3-NO-R2", "H3-NO-R3", "H4-HI-R1", "H4-HI-R2", "H4-HI-R3", "H4-NO-R1", "H4-NO-R2", "H4-NO-R3") # Set the sites names pathFlacFronten<-"C:/Program Files (x86)/FLAC Frontend/tools"

Directory<-"C:/Users/gamandin/Desktop/test"

DurationFile<-10

STL_type = "STLd"

STLd_Month="06"

STLd_Ndays=3

STLd_Hours=c("00","01","02","03","04")

FactorQuality<-ListInd_subWind_reorg[[1]][,1]

STLDataPrep(pumilio_URL=pumilio_URL,Colname=Colname,SiteNames=SiteNames,pathFlacFronten=pathFlacFronten,Directory=Directory,DurationFile=DurationFile,STL_type=STL_type,STLd_Month=STLd_Month,STLd_Ndays=STLd_Ndays,STLd_Hours=STLd_Hours)

References

Villanueva-Rivera, L. J., & Pijanowski, B. C. (2012). Pumilio: A Web-Based Management System for Ecological Recordings. Bulletin of the Ecological Society of America, 93(1), 71–81. https://doi.org/10.1890/0012-9623-93.1.71

STLDataPrepAverage

The software Sonic Time Lapse Builder have been developped by Benjamin Gottesman and Mark Durham. It is creating a Audio file by concataining several .wav files.

To use Sonic Time Lapse Builder program, you will need to drag a directory containing .these files. However, sometimes you will need to avergae the information from several sites. For example you might be interested in the Factor "Habitat" and have four sites in a forest and four sites in a prairie and you final goal is to produce one average time lapse for the forest habitat and one for the prairie. A code ("STLDataPrep.r") will help you to prepapre data from the Pumilio Database in order to have clean time series of files for a seasonality (along month, by default it will consider all the months) or daily variation (along hour of the day, by default it will consider all hours). This code will help you to average theses along different subdirectory based on a factor (can be among days of a site or among site of a factor) Author Amandine Gasc, 20160921.

Arguments

DirectoryFrom

character, directory path of the results from the function STL_DataPrepFromPumilio. Need to be one more directory level in it and the same number of files in each of these subdirectories.

DirectoryTo

character, directory path of the results you will obtained

TableFactorAveraged

a table of two columns, the first one is the directory name and the second one is the factor associated to it.

Example (not working for everyone, only to consult for argument formatting)

If you want to obtain a daily variation from different days of a same site

DirectoryFrom="C:/Users/gamandin/Desktop/DayJune/H1-NO-R2"

DirectoryTo="C:/Users/gamandin/Desktop/DayJuneAveraged/H1-NO-R2" TableFactorAveraged=cbind(Sites=dir("C:/Users/gamandin/Desktop/DayJune/H1-NO-R2"),Factor=c("201406","201406","201406","201406","201406"))

STLDataPrepAverage(DirectoryFrom=DirectoryFrom,DirectoryTo=DirectoryTo,TableFactorAveraged=TableFactorAveraged)

Example for different sites based on a factor disturbance level

DirectoryFrom="C:/Users/gamandin/Desktop/YearSTL"

DirectoryTo="C:/Users/gamandin/Desktop/YearSTL_averaged"

TableFactorAveraged=cbind(Sites=dir("C:/Users/gamandin/Desktop/YearSTL"),Factor=c("H1-HI","H1-HI","H1-HI","H1-NO","H1-NO","H1-NO","H2-HI","H2-HI","H2-HI","H2-NO","H2-NO","H2-NO","H3-ME","H3-ME","H3-ME","H3-NO","H3-NO","H3-NO","H4-HI","H4-HI","H4-HI","H4-NO","H4-NO","H4-NO"))

STLDataPrepAverage(DirectoryFrom=DirectoryFrom,DirectoryTo=DirectoryTo,TableFactorAveraged=TableFactorAveraged)

DistAverage

DistAverage<-
function(mat,factor_Site,factor_Day,factor_Hour,FactorToAverage,RemoveNA=TRUE)

The DistAverage function (Gasc et al. PLoS ONE 2013) has been developed in order to manipulate large distance matrices produced by the calculation of Beta acoustic indices such as Df (Df in Sueur et al, PLoS ONE 2008).

The function will calculate averaged matrices from an original pair-wise matrix by averaging on the "hour" (average similar hours), on the "sites" (average similar Sites) or on the days (average similar days).

Arguments

mat

mat is of class matrix and need to be an euclidean distance matrix. To check if this distance is euclidean you can type: is.euclidean(as.dist(mat)). In case of non euclidean distance, a transformation is possible such as: is.euclidean(lingoess(as.dist(mat)))

factor_Site

factor_Site need to be a vector of the length of the column or row of the matrix

factor_Day

factor_Day need to be a vector of the length of the column or row of the matrix

factor_Hour

factor_Hour need to be a vector of the length of the column or row of the matrix

FactorToAverage

FactorToAverage is a character that can be "factor_Site", "factor_Day" or "factor_Hour"

RemoveNA

RemoveNA can be "TRUE" or "FALSE", in case it is TRUE and if there are some NA in the final averaged matrix, a function will remove the NA value by a step by step method in order to remove the less data possible.

Example without NA

vec<-sample(seq(from=0.11,to=0.99,by=0.01), 36*36, replace = TRUE, prob = NULL);

mat<-as.matrix(as.dist(matrix(vec, 36, 36)));

colnames(mat)<-rownames(mat)<-c(paste(rep("site1",12),c(rep("day1",4),rep("day2",4),rep("day3",4)),c("h1","h2","h3","h4"),sep= "*"),*
paste(rep("site2",12),c(rep("day1",4),rep("day2",4),rep("day3",4)),c("h1","h2","h3","h4"),sep=""),
paste(rep("site3",12),c(rep("day1",4),rep("day2",4),rep("day3",4)),c("h1","h2","h3","h4"),sep="_ "));

MatrixAveraged<-DistAverage(mat=mat,factor_Site=substr(colnames(mat),1,5),factor_Day=substr(colnames(mat),7,10),factor_Hour=substr(colnames(mat),12,13),FactorToAverage="factor_Hour",RemoveNA=TRUE)

Example with NA

vec<-sample(seq(from=0.11,to=0.99,by=0.01), 36*36, replace = TRUE, prob = NULL);

vec[c(5:145,1200:1250)]<-NA;

mat<-as.matrix(as.dist(matrix(vec, 36, 36)));

colnames(mat)<-rownames(mat)<-c(paste(rep("site1",12),c(rep("day1",4),rep("day2",4),rep("day3",4)),c("h1","h2","h3","h4"),sep= "*"),*
paste(rep("site2",12),c(rep("day1",4),rep("day2",4),rep("day3",4)),c("h1","h2","h3","h4"),sep=""),
paste(rep("site3",12),c(rep("day1",4),rep("day2",4),rep("day3",4)),c("h1","h2","h3","h4"),sep="_ "));

MatrixAveraged<-DistAverage(mat=mat,factor_Site=substr(colnames(mat),1,5),factor_Day=substr(colnames(mat),7,10),factor_Hour=substr(colnames(mat),12,13),FactorToAverage="factor_Hour",RemoveNA=TRUE)

References

Gasc, A., Sueur, J., Pavoine, S., Pellens, R., & Grandcolas, P. (2013). Biodiversity Sampling Using a Global Acoustic Approach: Contrasting Sites with Microendemics in New Caledonia. PLoS ONE, 8(5), e65311. https://doi.org/10.1371/journal.pone.0065311

Releases

No releases published

Packages

No packages published

Languages

- ● **R** 100.0%