

Appendix S1

August 16, 2016

Code contained within warbleR: an R package to streamline analysis of animal acoustic signals.

Install and load warbleR

```
# install and load package
install.packages("warbleR")
library(warbleR)
```

Download Xeno-Canto recordings and metadata

```
# Query Xeno-Canto for metadata using genus as a keyword
Phae <- querxc(qword = "Phaethornis", download = FALSE)

# Query Xeno-Canto for metadata using genus and species as keywords
Phae.lon <- querxc(qword = "Phaethornis longirostris", download = FALSE)

# Visualize geographic spread of Xeno-Canto recordings for given keyword
xcmaps(X = Phae.lon, img = FALSE)

# Filter recordings by vocalization type
Phae.lon.song <- Phae.lon[grepl("song", Phae.lon$Vocalization_type, ignore.case = TRUE),]

# Filter recordings by location
Phae.lon.song <- Phae.lon.song[grepl("Sarapiquí, Heredia", Phae.lon.song$Locality,
                                     ignore.case = FALSE),]

# Filter recordings by quality score
Phae.lon.song <- Phae.lon.song[Phae.lon.song$Quality == "A", ]

# Download desired recordings using filtered data frame as a query
querxc(X = Phae.lon.song, download = TRUE)
```

```
# Convert mp3 to wav format
# Simultaneously lower sampling rate for more speed in following analyses
mp32wav(samp.rate = 22.05)
```

Create long spectrograms for visual inspection and signal classification

```
# Create long spectrograms of whole recordings
lspec(flim = c(1.5, 11), ovlp = 10, sxrow = 6, rows = 15)
```

Detect or select signals within recordings

```
# Automatically select signals within recordings using amplitude, duration and
# frequency thresholds
Phae.ad <- autodetec(bp = c(2, 9), threshold = 20, mindur = 0.09, maxdur = 0.22,
                   ssmooth = 900, ls = TRUE, res = 100, flim= c(1, 12), wl = 300,
                   set =TRUE, sxrow = 6, rows = 15)

# Manually select signals within recordings using an interactive spectrogram created
# within the RStudio graphics device
manualoc(flim = c(1, 11), wl= 300)
```

Filter by signal-to-noise ratio

```
# Filter selections by signal to noise ratio
Phae.snr <- sig2noise(X = Phae.ad[seq(1, nrow(Phae.ad), 2), ], mar = 0.04)

# Filter 5 selections from each recording
Phae.hisnr <- Phae.snr[ave(-Phae.snr$SNR, Phae.snr$sound.files, FUN = rank) <= 5, ]
```

Measure acoustic parameters and acoustic similarity

```
# Measure 22 acoustic parameters across the whole batch of selections
params <- specan(Phae.hisnr, bp = c(1, 11), threshold = 15)

# Extract dominant frequency as a time series using dynamic time warping
tsLBH <- dfDTW(Phae.hisnr, length.out = 30, flim = c(1, 12), picsize = 2, res = 100,
              bp = c(2, 9))
```

```

# Filter a single selection per recording for following analyses
Phae.hisnr1 <- Phae.snr[ave(-Phae.snr$SNR, Phae.snr$sound.files,
                          FUN = rank) <= 1, ]

# Perform spectrogram cross-correlation to compare acoustic similarity between
# filtered signals
xcor <- xcorr(X = Phae.hisnr1, wl = 300, frange = c(2, 9), ovlp = 90, dens = 0.9,
              wn = "hanning", cor.method = "pearson")

# Visualize cross-correlation results
xcorr.graph(X = xcor, cex.cor = 1, cex.lab = 1, rel.cex = FALSE,
            labs = gsub("[^0-9]", "", Phae.hisnr1$sound.files))

```

Interactive singing analysis

```

# Load coordinated singing data contained in package
data(sim.coor.sing)

# Visualize overlap between individuals
coor.graph(X = sim.coor.sing, ovlp = TRUE, only.coor = FALSE, xl = 2, res = 80)

# Perform randomization test to evaluate overlapping or alternating patterns
coor.test(sim.coor.sing, iterations = 100, less.than.chance = TRUE)

```

Table S1. 22 acoustic parameters measured by *specan* , which can be utilized to compare signal structure.

Function	Description
<i>duration</i>	length of signal
<i>meanfreq</i>	mean frequency, kHz
<i>sd</i>	standard deviation of frequency
<i>median</i>	median frequency, kHz
<i>Q25</i>	first quantile of frequency
<i>Q75</i>	third quantile of frequency
<i>IQR</i>	interquantile range of frequency
<i>skew</i>	skewness
<i>kurt</i>	kurtosis
<i>sp.ent</i>	spectral entropy
<i>sfm</i>	spectral flatness
<i>mode</i>	mode frequency, kHz
<i>centroid</i>	frequency centroid
<i>peakf</i>	peak frequency (freq. with highest energy)
<i>meanfun</i>	average fundamental frequency
<i>minfun</i>	minimum fundamental frequency
<i>maxfun</i>	maximum fundamental frequency
<i>meandom</i>	average dominant frequency
<i>mindom</i>	minimum dominant frequency
<i>maxdom</i>	maximum dominant frequency
<i>dfrange</i>	dominant frequency range
<i>modindx</i>	modulation index