

Project 2 - Task 2

Hazim Fitri P152419

2025-06-29

Contents

Crying Man Audio	6
------------------	---

```
library(seewave)
library(rpanel)

## Loading required package: tcltk

## Package 'rpanel', version 1.1-5: type help(rpanel) for summary information

library(tuneR)

# upload data of a lion roar
lion = readMP3('Data/lion_roar.mp3')
str(lion)

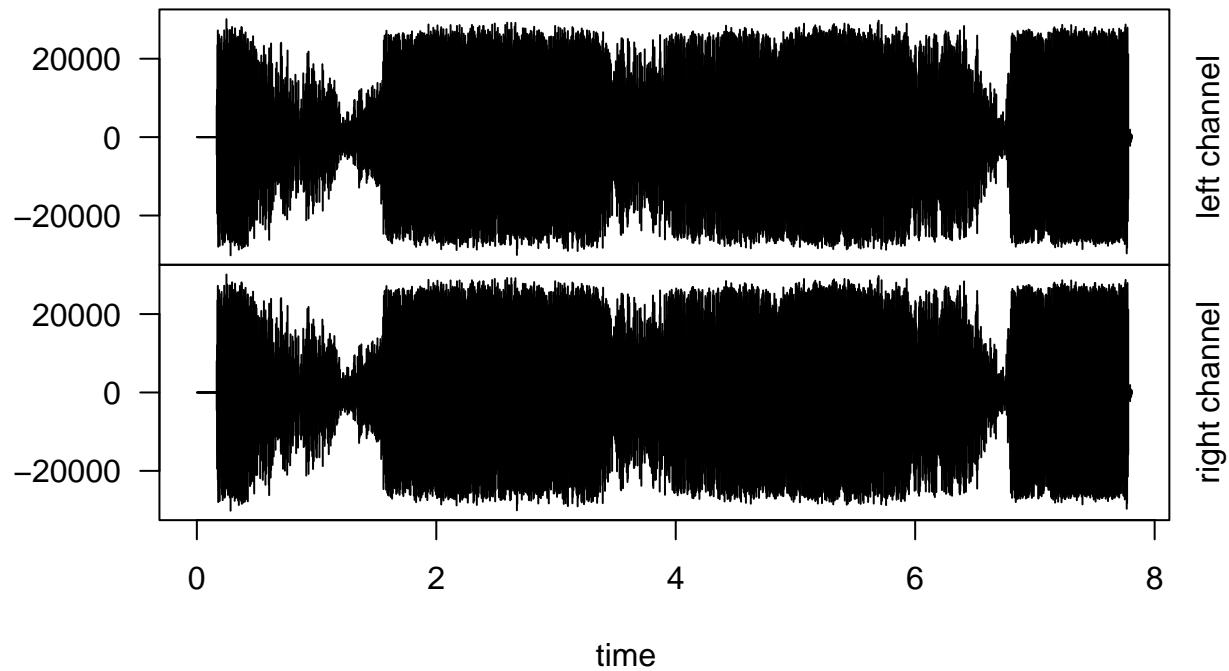
## Formal class 'Wave' [package "tuneR"] with 6 slots
##   ..@ left      : int [1:344448] 0 0 0 0 0 0 0 0 0 ...
##   ..@ right     : int [1:344448] 0 0 0 0 0 0 0 0 0 ...
##   ..@ stereo    : logi TRUE
##   ..@ samp.rate: num 44100
##   ..@ bit       : num 16
##   ..@ pcm       : logi TRUE

plot.frequency.spectrum <- function(X.k, xlims=c(0,length(X.k))) {
  plot.data <- cbind(0:(length(X.k)-1), Mod(X.k))

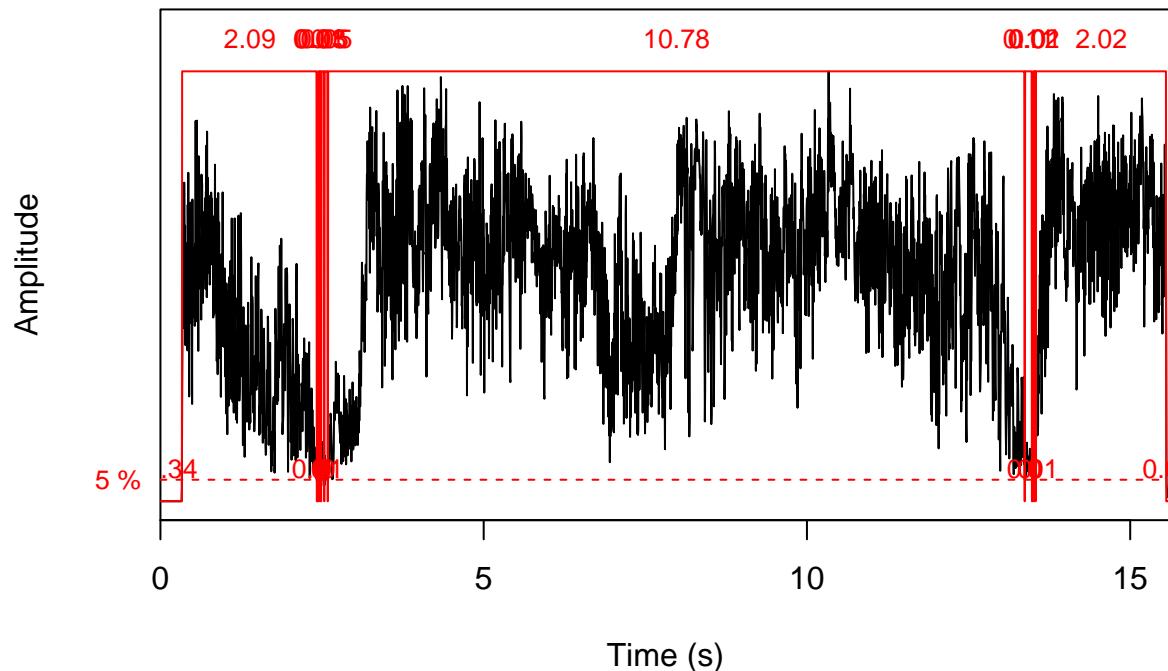
  # TODO: why this scaling is necessary?
  plot.data[2:length(X.k),2] <- 2*plot.data[2:length(X.k),2]

  plot(plot.data, t="h", lwd=2, main="",
       xlab="Frequency (Hz)", ylab="Strength",
       xlim=xlimits, ylim=c(0,max(Mod(plot.data[,2]))))
}

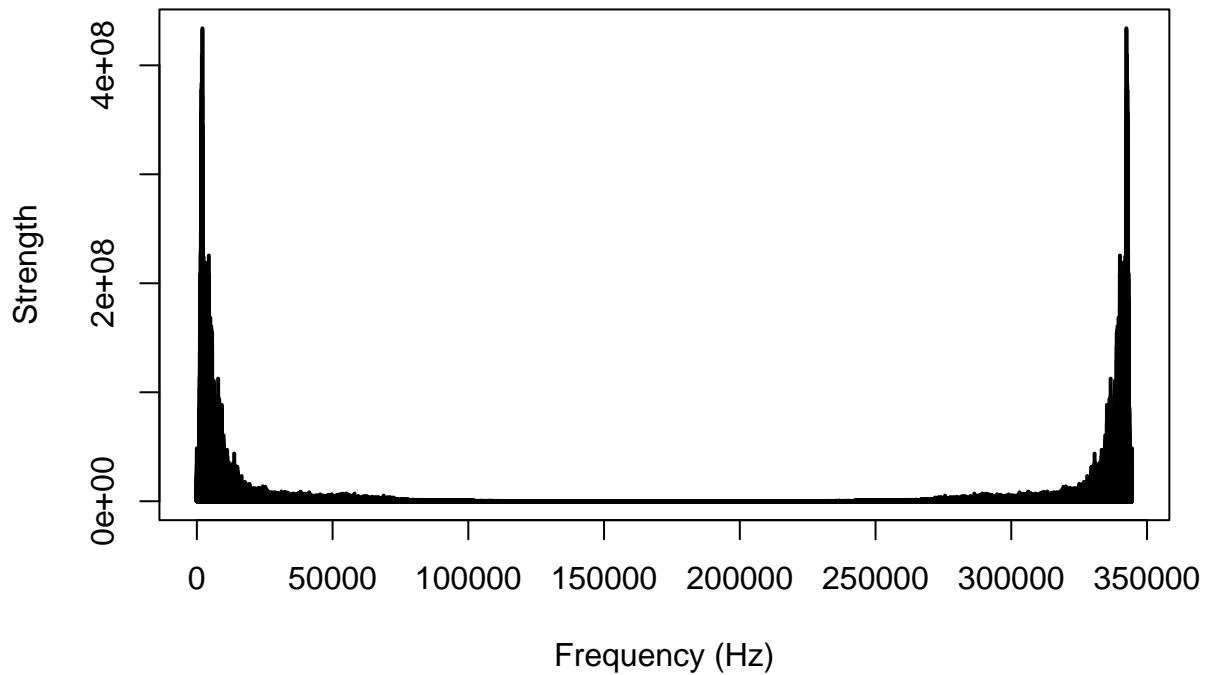
plot(lion)
```



```
timer(lion, f=22050, threshold=5, msSmooth=c(100,0))
```

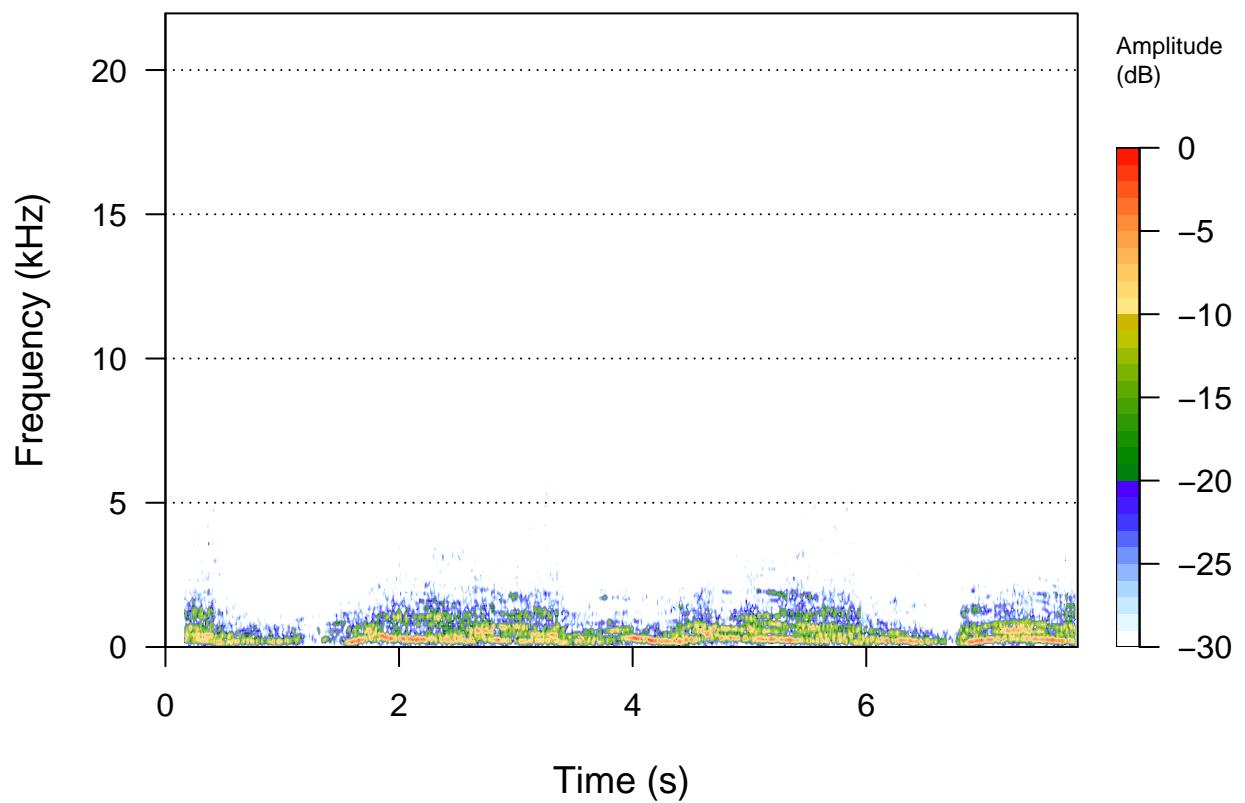


```
lion_r = fft(lion@left)
plot.frequency.spectrum(lion_r)
```

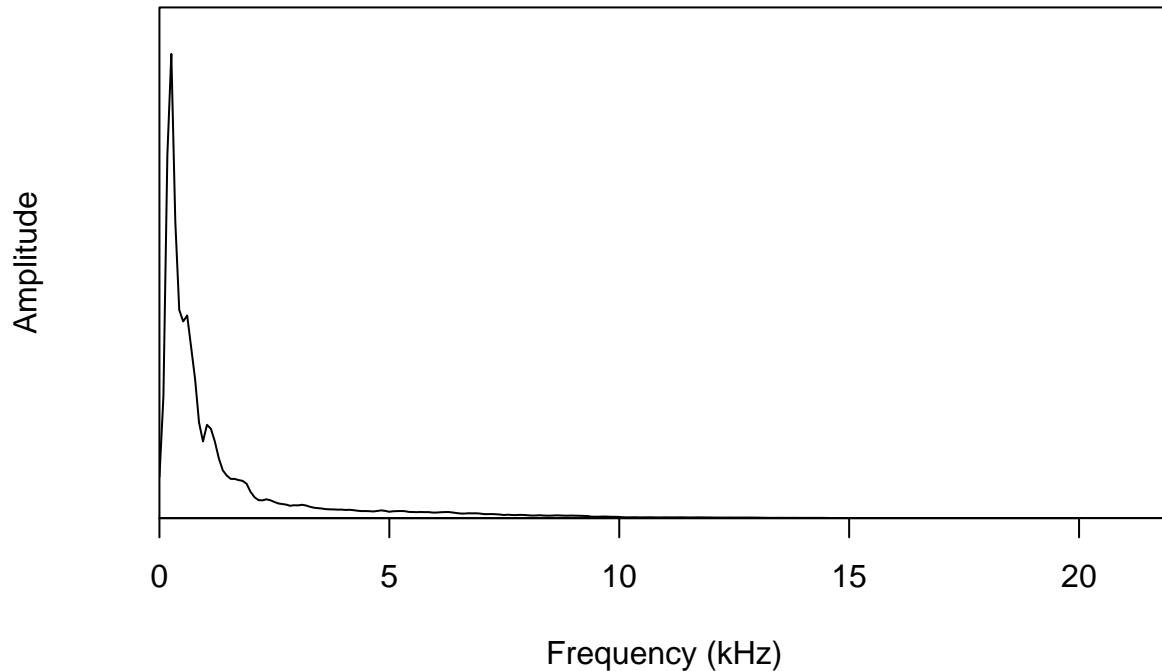


```
dynspec(lion, wl=1024, osc=T)
```

```
spectro(lion)
```



```
meanspec(lion)
```

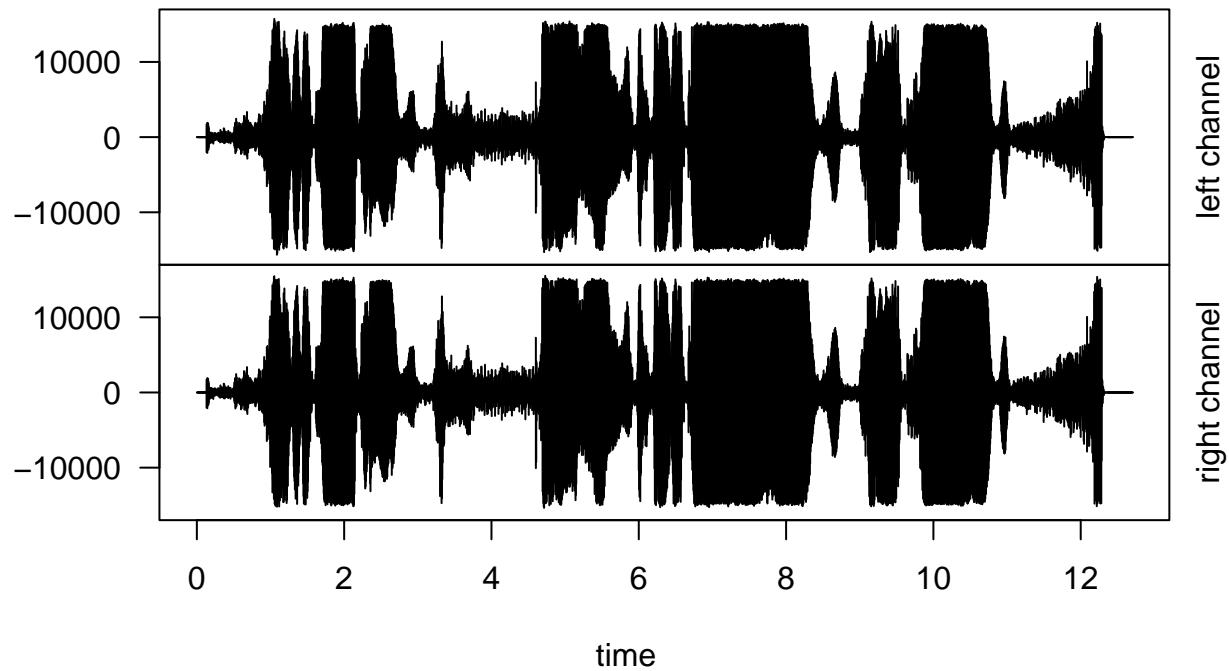


Crying Man Audio

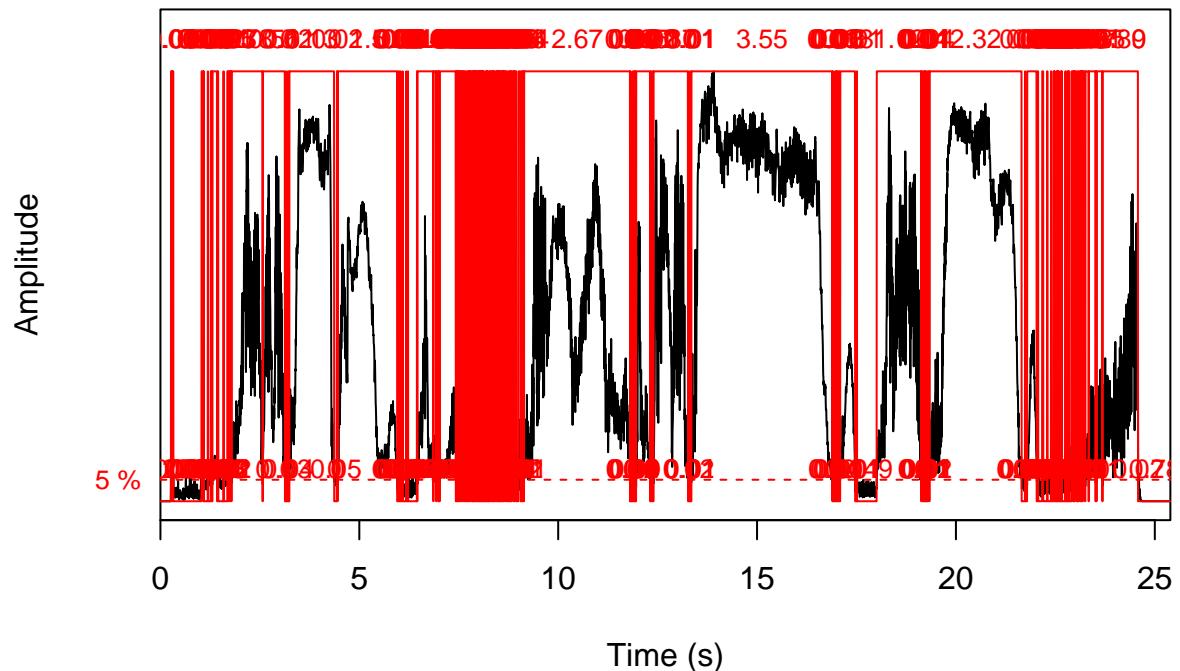
```
# upload data of a crying man
cry = readMP3('Data/crying_man.mp3')
str(cry)

## Formal class 'Wave' [package "tuneR"] with 6 slots
##   ..@ left      : int [1:559872] 0 0 0 0 0 0 0 0 0 ...
##   ..@ right     : int [1:559872] 0 0 0 0 0 0 0 0 0 ...
##   ..@ stereo    : logi TRUE
##   ..@ samp.rate: num 44100
##   ..@ bit       : num 16
##   ..@ pcm       : logi TRUE

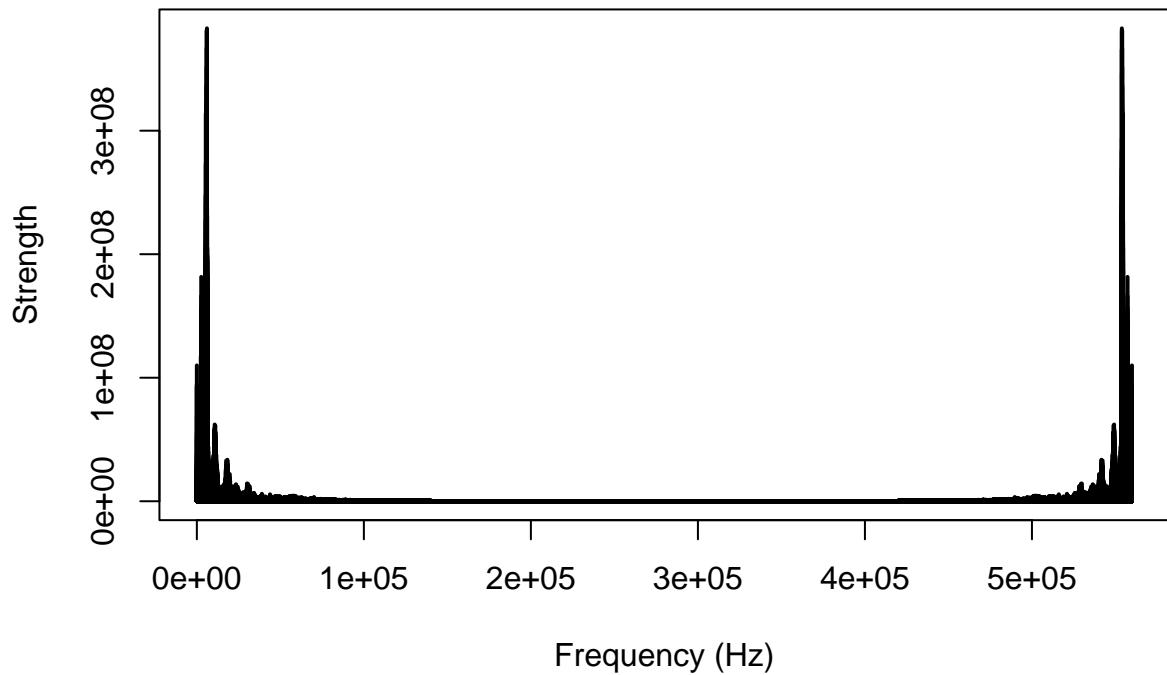
plot(cry)
```



```
timer(cry, f=22050, threshold = 5, msSmooth = c(100,0))
```

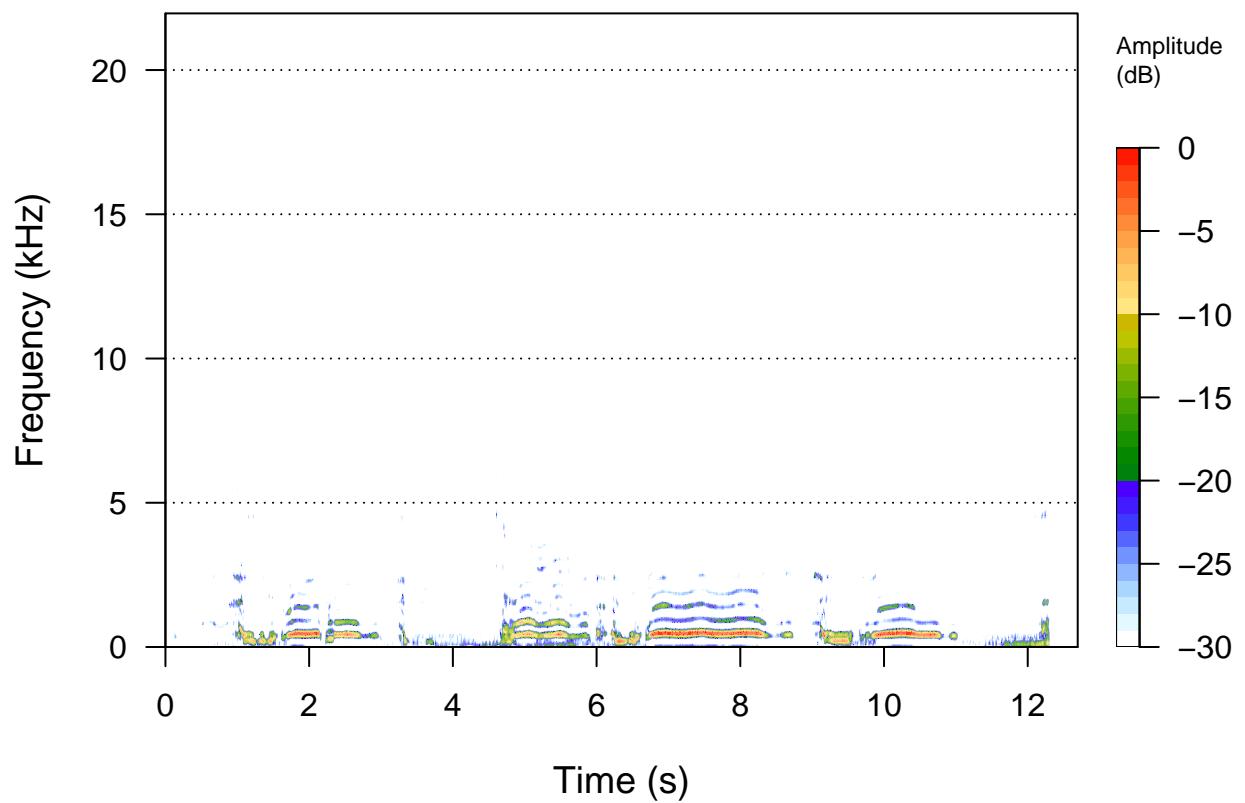


```
cry_m = fft(cry@left)
plot.frequency.spectrum(cry_m)
```



```
dynspec(cry)
```

```
spectro(cry)
```



```
meanspec(cry)
```

