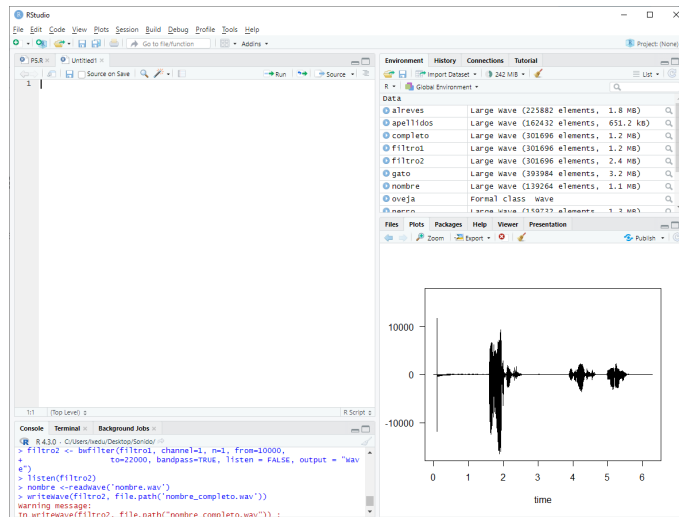
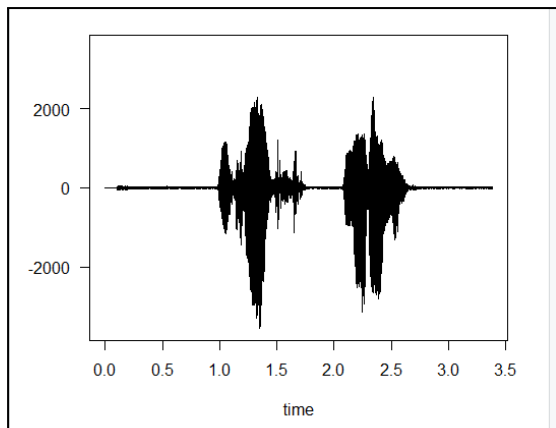


Práctica 5: Sonido

Para realizar esta práctica se realizará la instalación tanto de R como de Rstudio como se indica en el seminario. Una vez hecho esto, se trabajará con Rstudio.





Como siguiente, se obtienen las cabeceras de ambos archivos utilizando la función `str()`:

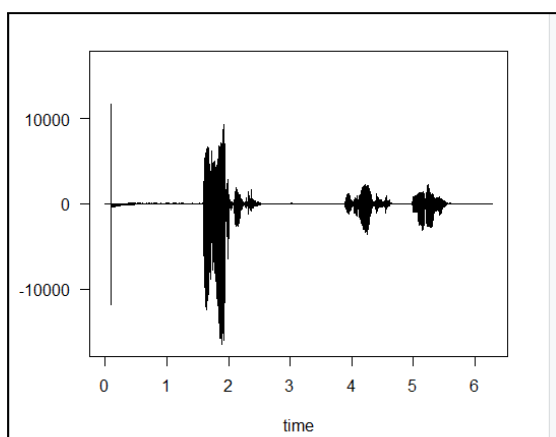
```
Formal class 'wave' [package "tuner"] with 6 slots
..@ left      : int [1:139264] 0 0 0 0 0 0 0 0 0 ...
..@ right     : int [1:139264] 0 0 0 0 0 0 0 0 0 ...
..@ stereo    : logi TRUE
..@ samp.rate: int 48000
..@ bit       : int 16
..@ pcm       : logi TRUE
> |
```

```
> plot(extractwave(apellidos, from = 1, to = 393984))
> str(apellidos)
Formal class 'wave' [package "tuner"] with 6 slots
..@ left      : int [1:162432] 0 0 0 0 0 0 0 0 0 ...
..@ right     : num(0)
..@ stereo    : logi FALSE
..@ samp.rate: num 48000
..@ bit       : num 16
..@ pcm       : logi TRUE
> |
```

Utilizando la función `paste()` se unen ambos sonidos, de tal forma que se forme un archivo en que parezca que se dicen nombres y apellidos de manera seguida:

```
filtro1 <- paste(apellidos, nombre, output="Wave")
```

La onda resultante sería:



Por último (en cuanto a este archivo) se le pasa un filtro de frecuencia y se guarda como nombre_completo.wav:

```
filtro2 <- bwfilter(filtro1, channel=1, n=1, from=10000,  
to=22000, bandpass=TRUE, listen = FALSE, output = "Wave")
```

```
writeWave(filtro2, file.path('nombre_completo.wav'))
```

En cuanto al otro fichero, se utilizará perro.wav. Primero se le aplica eco y después se le da la vuelta con las respectivas funciones:

```
perroECO <- echo(perro,f=22050,amp=c(0.8,0.4,0.2),delay=c(1,2,3),  
output="Wave")
```

```
alreves <- revw(perroECO, output='Wave')
```

Y para terminar la práctica, se guarda de la misma manera que el archivo anterior:

```
writeWave(alreves, file.path('alreves.wav'))
```