

# Práctica 5. Experimentación con el sistema de salida de sonido

*Duración: 2 sesiones*

## 1. Objetivos de la práctica

Los objetivos concretos de esta práctica son:

- Identificar y representar gráficamente la forma de onda de señales de sonido.
- Conocer la estructura de un fichero típico de sonido (ficheros WAV).
- Entender y operar con los parámetros principales de una señal de sonido.

Para realizar esta práctica se puede programar en lenguaje R para manejar el sonido, utilizando el entorno de programación RStudio, o bien otro lenguaje, como Python, C++ o Java.

## 2. Manejo de archivos de sonido

A partir de lo estudiado en el seminario, y si se opta por usar el lenguaje R, se utilizarán los paquetes `tuneR` y `seewave` para trabajar con varios archivos de sonido (tanto en formato WAV como en MP3). En el caso de usar otro lenguaje, se hará uso de las librerías que sean necesarias.

Para R, se propone usar las siguientes funciones ya estudiadas en el seminario:

- `readWave()`
- `readMP3()`
- `str()`
- `listen()`
- `plot()`
- `cutw()`
- `pastew()`
- `deletew()`
- `bwfilter()`
- `echo()`
- `revw()`

## Cuestiones a resolver

El objetivo principal de esta práctica es aprender a reproducir, crear, modificar y manejar sonidos usando el lenguaje elegido. Se deberá crear un programa para realizar las siguientes acciones:

1. Leer dos ficheros de sonido (WAV o MP3) de unos pocos segundos de duración cada uno. En el primero debe escucharse el nombre de la persona que realiza la práctica. En el segundo debe escucharse el apellido.
2. Dibujar la forma de onda de ambos sonidos.
3. Obtener la información de las cabeceras de ambos sonidos.
4. Unir ambos sonidos en uno nuevo.
5. Dibujar la forma de onda de la señal resultante.
6. Pasarle un filtro de frecuencia para eliminar las frecuencias entre 10000Hz y 20000Hz
7. Almacenar la señal obtenida como un fichero WAV denominado "mezcla.wav".
8. Cargar un nuevo archivo de sonido, aplicarle eco y a continuación darle la vuelta al sonido. Almacenar la señal obtenida como un fichero WAV denominado "alreves.wav".

Como resultado de hacer la práctica **se mostrará** al profesor el funcionamiento del script desarrollado (comprobar que los sonidos se representan y suenan como deben).

En el documento a entregar se describirá cómo se ha creado el programa y se mostrarán capturas de pantalla de las formas de onda obtenidas.

## Normas de entrega

La práctica/seminario podrá realizarse de manera individual o por grupos de hasta 2 personas.

Se entregará como un archivo de texto en el que se muestre la información requerida. También se puede utilizar la sintaxis de Markdown para conseguir una mejor presentación e incluso integrar imágenes o capturas de pantalla. La entrega se realizará subiendo los archivos necesarios al repositorio "PDIH" en la cuenta de GitHub del estudiante, a una carpeta llamada "P5".

Toda la documentación y material exigidos se entregarán en la fecha indicada por el profesor. No se recogerá ni admitirá la entrega posterior de las prácticas/seminarios ni de parte de los mismos.

La detección de prácticas copiadas implicará el suspenso inmediato de todos los implicados en la copia (tanto de quien realizó el trabajo como de quien lo copió).

Las faltas de ortografía se penalizarán con hasta 1 punto de la nota de la práctica/seminario.

## Referencias

<https://www.r-bloggers.com/intro-to-sound-analysis-with-r/>  
<https://hansenjohnson.org/post/spectrograms-in-r/>

<http://samcarcagno.altervista.org/blog/basic-sound-processing-r/>  
[https://cran.r-project.org/web/packages/soundgen/vignettes/acoustic\\_analysis.html](https://cran.r-project.org/web/packages/soundgen/vignettes/acoustic_analysis.html)  
<http://rug.mnhn.fr/seewave/>  
[https://rstudio-pubs-static.s3.amazonaws.com/92070\\_d44c3e320fae4b21b12c6101f347b28c.html#1](https://rstudio-pubs-static.s3.amazonaws.com/92070_d44c3e320fae4b21b12c6101f347b28c.html#1)

<https://towardsdatascience.com/music-genre-classification-with-python-c714d032f0d8>  
<https://towardsdatascience.com/music-in-python-2f054deb41f4>  
<https://realpython.com/playing-and-recording-sound-python>  
<https://www.tutorialspoint.com/read-and-write-wav-files-using-python-wave>  
<https://akuum.org/questions/18625085/how-to-plot-a-wav-file>  
<https://realpython.com/playing-and-recording-sound-python>  
<https://stackoverflow.com/questions/2890703/how-to-join-two-wav-files-using-python>  
<https://stackoverflow.com/questions/24920346/filtering-a-wav-file-using-python>

## Anexo. Grabar un sonido

Para generar los archivos de sonido WAV con el nombre y/o apellido se pueden usar herramientas de línea de comandos (p.ej. en Linux o macOS):

**en macOS:**

```
say -o fich.wav --data-format=LEF32@22050 "Nombre Apellido"
```

**en Linux:**

```
espeak "Nombre Apellido" -w fich.wav
```