

# Tarea 3 - Packages

September 16, 2017

## 1 Introducción

Esta tarea tiene como tema el desarrollo de un *package* para Python. El objetivo de esta tarea es desarrollar la librería científica denotada *mmpy* (de las iniciales de *Maths and Music in Python*) que sirve para trabajar con archivos de audio. La librería tiene las funcionalidades de leer archivos de audios de diferentes formatos, graficar diferentes representaciones (temporales y espectrales), calcular features, aplicar filtros y guardar a disco los resultados obtenidos. La forma de ejecutar la función *method* de esta librería es la siguiente:

```
In [ ]: import mmpy as mm

        mm.method()
```

## 2 Tarea

### 2.1 Package

Para el desarrollo de *mmpy.py* usted tendrá que desarrollar los siguientes pasos.

1. Programe el método *style()* que aplica un estilo de gráficos por defecto diseñado por usted.
2. Programe el método *load(path)* que lee el archivo de audio en *path* y devuelve un objeto de la clase *Audio*.
3. Programe la clase *Audio* con los siguientes métodos (donde los argumentos de los métodos los diseña usted):
  - *\_init\_*: Inicializa un objeto de la clase *Audio* con variables *ndarray* que representen el archivo de audio. También es posible inicializarlo con la representación *ndarray* directamente.
  - *plot\_time*: Graficar la representación del audio en el dominio temporal.
  - *plot\_freq*: Graficar la representación del audio en el dominio de frecuencias.
  - *set\_time*: Setear valores del audio en su representación temporal.
  - *set\_freq*: Setear valores del audio en su representación de frecuencias.
  - *reset*: Resetea el audio a sus valores iniciales.
  - *save*: Guardar a disco el audio representado.
  - *play*: Reproducir el audio representado.
4. Programe el método *get\_features(audio)* que devuelve un dataframe con diferentes features del audio.

5. Programe el método *widget(audio)* que genera un *ipywidget* que permite setiar rangos de valores, graficar, resetear y guardar el audio, en ambos dominios. Tiene el botón *Play* que sirve para reproducir el audio representado actualmente, y el botón *Save Plot* que guarda a disco los gráficos.
6. Programe el script *setup.py* que instala *mmpy* en el entorno de Python. Se deben especificar las dependencias requeridas por su *package*.

## 2.2 Jupyter Notebook

Para el testeo de su librería, usted deberá crear el Jupyter notebook *tarea3.ipynb* que pruebe varios ejemplos que utilicen todas las funciones desarrolladas anteriormente. Para esto va a generar 10 objetos *Audio* inicializados con un *ndarray*, obtenidos de combinaciones de las siguientes funciones:

- constante en un intervalo
- cuadrática
- exponencial
- seno y coseno
- ruido blanco

Los filtros son métodos que modifican los valores de audios en su representación temporal y/o de frecuencia, tales como sumar, multiplicar o convolucionar por alguna de las funciones listadas anteriormente. Implementar 5 filtros diferentes.

Para cada uno de los objetos *Audio* aplicar 2 filtros, graficar y calcular features de todas las representaciones, reproducir el audio original y el audio final. Realizar una descripción de cada uno de los ejemplos, filtros y resultados obtenidos.

Probar la interfaz gráfica con dos archivos de audio reales: la primera una canción instrumental y la segunda una canción con canto. Deberá guardar todos los gráficos originales y finales, incluirlos en el informe y realizar una descripción de los resultados obtenidos.

## 3 Entrega

El formato de entrega será subir a u-cursos el informe *tarea3.pdf*, el Jupyter notebook *tarea3.ipynb*, que debe ejecutar sin errores desde la primera celda a la última, donde cada celda debe resolver cada uno de los pasos descritos. Además se deben adjuntar los archivos de audio utilizados, *mmpy.py* y *setup.py*, el código debe estar comentado y el notebook debe estar escrito en forma de informe técnico, escribiendo una celda markdown antes de cada celda de código. Una vez que su notebook ejecute correctamente, deberá descargar la versión pdf del notebook y adjuntarla a su tarea. Este archivo corresponde a su informe de tarea, que deberá llamarse *tarea3.pdf*. Adjuntar el código del *package* como anexo del informe.