Exercise 1: Basics of speech processing and analysis

1. Exercise instructions

- Implement and return files **ex1_main.py** and **ex1_windowing.py**.
- Return your answers to MyCourses by 23:59 on Tuesday, September 21th, 2021.

2. Introduction

The goal of this exercise is to get acquainted with speech processing inside the Python environments. This includes reading, resampling, windowing, and visualization of speech signals.

3. Reading and resampling

For most of the exercises, we will work with the same audio file that you record by yourself. Please, record a short audio clip (about 5 seconds) in a clean environment and save by your firstname_speech.wav in the Sounds folder. You are provided with the skeleton functions. The objective of this section is to acquaint yourself with the process of reading, writing audio files, and resampling.

First, read the audio file you recorded and obtain its sampling frequency. Following this, check if the sampling frequency is Fs = 16kHz. If not, resample the audio signal to the frequency, desired frequency.

3.1 Useful functions (from scipy)

wavread, resample_poly

4. Windowing

Write a windowing function which takes the data, window length (frame length, as samples), hop size (as samples), and the type of window function as input and results in an output matrix of dimension n × m, with n corresponding to frame length and m corresponding to number of frames. The first frame starts at the beginning of the data.

We will use a frame length of 25ms, with 50% overlap. The basic window types should be rectangular, Hann, Hamming, Cosine (square root Hann) window types.

In the windowing function, to implement framing itself, first compute the number of frames for the signal under consideration using this formula:

$$number\ of\ frames = 1 + [\frac{(n-k)}{o}]$$

where n is the data length, k is frame length in samples and o is the number of overlapping samples. Use the floor function to round the number of frames. Following this, segment the data with right frame size and overlap and remember to apply the chosen window function to the generated signal frames. Note that if the end sample of the frame segment is larger than data length, zero-pad the remainder to achieve constant frame length.

4.1. Useful functions (all from numpy)

ones, hanning, hamming, sqrt, zeros, floor

5. Plotting and visualization

Visualization and plotting is an important step not only in the evaluation of methods but also in debugging and finding out problems with the methods. In this section, we will introduce some of the most common plotting and visualization schemes used in speech processing.

In the first figure, we shall plot the original audio signal (Subplot 1), a voiced frame in the time domain (Subplot 2), and the same voiced frame in the frequency domain (Subplot 3). It is important to make note of the plot axes. Divide the figure into 3 subplots, vertically. In the top-most subplot, plot the time domain original signal; the x-axis should be in seconds. In the middle subplot, plot the time domain signal of a voiced frame of your choice. Make sure that the x-axis is in milli-seconds. In the last subplot, plot the first half of the magnitude spectrum of the same voiced frame (i.e., from 0 Hz to Nyqvist frequency); represent x-axis as Hz, and y-axis as decibels. In the second part of this section, we will compute and plot the signal spectrogram. The x-axis should represent time and the y-axis should represent frequency.

For all the plots, label the axes and titles with appropriate strings.ex1_windowing.

Sample plots expected are shown in Figure_1_solution.png and Figure_2_solution.png.

5.1. Useful functions (matplotlib)

figure, subplot, plot, title, xlabel, ylabel, autoscale, imshow, yticks

6. Return

The results of the exercise must be presented in two documents.

ex1_windowing.py*:* It contains the windowing function (Keep it in a separate file as it will be used in future exercises).
ex1_main.py: Code for the rest of the tasks from the exercise.
Executing **ex1_main.py** has to run the whole exercise and plot the two required pictures.