Exercise 1

In this exercise, you will implement a simple feature extraction. It is highly recommended to use Matlab’s “help” command.

(1P) Subtask 1.1
Please find and use the file “sound1.wav” (or “sound1.au”) on the course website. Use “wavread(…)” (or “auread(…)” respectively) to read in the file. Now make use of “sound(…)” to properly play the soundfile in Matlab.

(1P) Subtask 1.2
You have already heard about Pre-emphasis in the lecture. Now you should implement a function \( \text{pre}(s) \) which applies it to the audiostream \( s \). Use the following definition:

\[
f'_n = f_n - \alpha \cdot f_{n-1} \quad \text{with} \quad \alpha = 0.95
\]

(1) Which kind of filter is it? Is it a low- or high-pass filter?

(2P) Subtask 1.3
Implement the Matlab function:

```matlab
function M = windowing(s, shift, width)
```

which applies a Hamming window to signal segments (called frames) of width \( w \). How many frames \( f_n \) can be produced from a signal of length \( k \) assuming a shift of \( sh \) samples and a frame width of \( w \). \( M \) will be a Matrix \( \in \mathbb{R}^{f_n \times w} \).

(2P) Subtask 1.4
Use the function of 1.2 and apply it to the audiostream. Assume a shift of 10ms and a width of 25ms to create a spectrogram. Therefore use the Matlab function \( \text{spectrogram}(…) \) in combination with \( \text{view}(90, -90) \).

(\text{It is necessary to compute the amount of samples for the shift and width!})

Show the original audiostream and your spectrogram; Can you see some similarities?

---

\(^1\)You can use the Matlab function \( \text{filter}(…) \).
\(^2\)You can use the Matlab Funktion \( \text{hamming} \).
(1P) Subtask 1.5

You will find the Matlab function `mel` on our course website. Use it, assuming the following parameters:

\[
\begin{align*}
fl &= 133.3334 \text{Hz} \quad (2) \\
fh &= 6855.4976 \text{Hz} \quad (3) \\
fft_{\text{size}} &= 1024 \quad (4) \\
f_s &= 16k \text{Hz} \quad (5) \\
L &= 24 \quad (6) \\
fmel &= 1125 \text{Hz} \quad (7)
\end{align*}
\]

The result will be a Matrix \( W \in \mathbb{R}^{L \times \frac{\text{fft}_{\text{size}}}{2}} \) containing one filter per row. Plot the whole filterbank within the range 0:100.

(3P) Subtask 1.6

The goal of this subtask is to compute a MFCC-stream. After applying the preemphasis, the signal is windowed (for parameters see task 1.2). Now a FFT is applied to each frame, using \( \text{fft}_{\text{size}} \). Due to the symmetry, the second half of the resulting transformation can be dropped, thus reducing every frame to the size of \( \frac{\text{fft}_{\text{size}}}{2} \). Now the mel filterbank can be applied to the absolute values of the resulting fourier coefficients (This is a simple Matrix vector multiplication per frame!), reducing each frame to \( L \) values. Lastly, to compute another spectrogram, the \( \log_{10}(...) \) and \( \text{dct}(..., L) \) are applied. (A spectrogram can be visualized using `imagesc`!). Show the resulting spectrogram.

(2P) Subtask 1.7

Show the original signal, the spectrogram from task 1.4, as well as the spectrogram resulting from task 1.6.

What do you observe?