1 Exercise

This exercise is related to the second one of the previous Assignment.

1.1 (3P) Subtask

The goal of this subtask is to compute a MFCC-stream.
After applying the preemphasis, the signal is windowed (use the parameters of the related exercise).
Now a FFT is applied to each frame, using $fft\_size$. Due to the symmetry, the second half of the resulting transformation can be dropped, thus reducing every frame to the size of $fft\_size/2$.
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 $dct(..., L)$ are applied. ($A \ spectrogram \ can \ be \ visualized \ using \ imagesc!$).

Show the resulting spectrogram.

1.2 (2P) Subtask

Show the original signal, the spectrogram from task 1.4 of the related exercise, as well as the spectrogram resulting from task 1.1.

What do you observe?
2 Exercice

In this exercise, you will implement a simple k-Nearest Neighbour classifier!

2.1 (1P) Subtask

To measure distances for a kNN classifier, it sometimes is useful to try other norms than the euclidian norm. Therefore implement a function \( p\text{norm}(x, p) \) with \( x \in \mathbb{R}^n \) and \( p \geq 1 \):

\[
||x||_p = \left( \sum_{i=1}^{n} |x_i|^p \right)^{\frac{1}{p}}
\]  

(1)

2.2 (1P) Subtask

For visualisation purposes, write a function \( c\text{plot}(T, M) \), which plots the given datasets. The matrices have the following form:

- The first column is \( \in \mathbb{N}^n \) (\( n \) being the number of samples) and contains labels
- The second and third columns represent the samples’ \( x \) and \( y \) values and are \( \in [0, 12]^n \)

You can use Matlab’s \( \text{scatter}(\ldots) \) and \( \text{axis}(\ldots) \) functions. Your implementation is to plot the given datasets using 1 color per label. Make sure you plot the matrix \( M \) before \( T \), as \( M \) will be the dataset and \( T \) the training set.

2.3 (3P) Subtask

Now implement a function:

\[
\text{function } V = \text{classify}(T, M, k, p)
\]

which classifies the given samples \( M \in \mathbb{R}^{n \times f} \) with \( n \) being the number of samples, \( f \) the length of the feature vectors. \( V \in \mathbb{N}^n \) is a Vector containing the labeling your classifier produced. \( k \) is the number of neighbours to check for and \( p \) the parameter for the norm function. \( T \) is the already labeled training set and its structure is as described in 2.2. (Hint: In this task you can make use of Matlab’s \text{mode}(\ldots) function)

2.4 (2P) Subtask

Read in the file \textit{train1.dat (importdata(\ldots))}. Now generate a matrix \( M \in [0, 12]^{2500 \times 2} \) and classify it using your classifier. Use the euclidian distance and generate 4 plots (using \( \text{cplot}(\ldots) \)) in a single window for \( k = 1, 3, 5, 9 \).

What can you observe especially for \( k = 9 \)?