67355 Introduction to Speech Processing

Exercise 1 - 2024

The following exercise is structured from two parts: Theoretical and Practical.
The exercise should be done **in pairs** and is to be submitted via moodle by the deadline appearing under the subission box.
See submission guidelines for further instructions

1 Theoretical part (50 Points)

1.1 10 points

Prove: Multiplication over the time domain is equivalent to applying convolution over the frequency domain, i.e.

$$F[x_1(t) \cdot x_2(t)] = \frac{1}{2\pi} \left(X_1^F(\omega) * X_2^F(\omega) \right).$$

1.2 Laying the ground for Nyquist's sampling thm

Recall:

1. Impulse sampling with rate $\frac{1}{T}$ is in practice a discrete sampling $x_d(t)$ of a continuous signal $x(t)$

$$x_d(t) = \sum_{n=-\infty}^{\infty} x(t)\delta(t - nT) = x(t) \sum_{n=-\infty}^{\infty} \delta(t - nT).$$

2. Impulse train $s_T(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) = \sum_n \delta(t - nT)$; $\delta(x) = \{1 \text{ if } x = 0 \text{ else } 0\}$.
3. $X^F(\omega) = \int_{-\infty}^{\infty} x(t)e^{-i\omega t}dt$.
4. Convolution: $x(t) * y(t) = \int_{-\infty}^{\infty} x(\tilde{t})y(t - \tilde{t})d\tilde{t}$.
5. The notation X_d^F is the Fourier transform of x_d

In the next sections assume that the Fourier transform of an impulse train is $S_T^F(\omega) = \frac{2\pi}{T} \sum_n \delta\left(\omega - \frac{2\pi n}{T}\right)$,

1.2.1 10 points

Prove:

$$X_d^F(\omega) = \frac{1}{2\pi} \left(X^F(\omega) * S_T^F(\omega) \right).$$

1.2.2 10 points

Prove:

$$\sum_n \int_{-\infty}^{\infty} X^F(\tilde{\omega})\delta\left(\tilde{\omega} - \left(\omega - \frac{2\pi n}{T}\right)\right) d\tilde{\omega} = \sum_n X^F\left(\omega - \frac{2\pi n}{T}\right).$$

1.2.3 10 points

Prove that $X_d^F(\omega) = \frac{1}{T}\sum_n X^F\left(\omega - \frac{2\pi n}{T}\right)$.

1.2.4 10 points (Bonus)

Definition: A signal is called "band limited" if $\exists\omega_{max} \geq 0$ s.t. $\forall\omega, |\omega| > \omega_{max} : X^F(\omega) = 0$
Show that every band limited signal $x(t)$ bounded by ω_{max}, $x(t)$ could be reconstructed with sampling frequency

$$\frac{1}{T} = f_s > 2f_{max} \ ; \ f_{max} = \frac{\omega_{max}}{2\pi},$$

i.e. show that for the above, the following holds:

$$\left(\forall x_d(t), \forall|\omega| \leq \omega_{max} : X_d^F(\omega) = \frac{1}{T}X^F(\omega)\right) \Leftrightarrow f_s > 2f_{max},$$

To simplify things, assume that $\forall w \in [-\omega_{max}, \omega_{max}] : X^F(\omega) \neq 0$.

Hint: prove by negation.
Guidance:

1. Assume that $\exists f_s < 2 \cdot f_{max}$ s.t $\forall x_d(t), \forall|\omega| \leq \omega_{max} : X_d^F(\omega) = \frac{1}{T}X^F(\omega)$. Think of when this doesn't hold?
2. Prove the other way regularly.
3. Use section 1.2.3

1.3 10 points

Let $x(t) = sin(2\pi \cdot 1000 \cdot t) + sin(2\pi \cdot 5000 \cdot t)$. We sample $x(t)$ with sampling rate of $8[KHz]$.
What frequencies would appear in the Fourier transform of the discrete measured signal? Assume $\forall|\omega| > 2\pi \cdot 5000 :$ $X^F(\omega) = 0$.
Guidance: We've seen that in order to be able to reconstruct a certain frequency - we MUST have 2 measured points per cycle. Due to that, it must hold that all frequencies $> \frac{1}{2} \cdot f_s$ are 0 as we cannot reconstruct them. Think which frequencies may appear withing the range we are able to reconstruct.
Hints: (i) use what you proved in section 1.2.3; (ii) think what frequencies would have $X^F(\cdot) \neq 0$. Meaning consider the values where ω and n equals the signal frequencies; (iii) recall that Fourier transform is symmetric around zero (we also have negative frequencies).

2 Practical part (50 Points)

In Ex1.zip you will find several files. In each .py file there are blank functions for you to implement.
For your convenience, we have a working python environment you can use on the school's computers, found at */cs/course/2023/speech/miniconda3/envs/ISP/bin/python*. You can use this env by simply replacing your *python* command with this absolute path or by cloning the env. Alternatively, we offer a *pip freeze* of the environment in *our_env_pip_freeze.txt* so you can install the corresponding envs yourself.

Important note: it is under your responsibility to make sure that your code runs using the mentioned environment **on the university servers** as it is the environment we will be using to test your exercises.

Important note II: not following format in *ids.txt* or failing to pass *presubmit.py* will result in a 0 score for this part of the exercise.

The Points for these notebooks will be divided as follows:

- digit classifier - 30 pts.
- time stretch - 20 pts.

3 Submission Guidelines

- All theoretical questions are to be typed (i.e. not handwritten) and submitted as a single pdf.
- The header of your submitted solution pdf should include your IDs.
- Your submitted .tar file needs to include ids.txt file with a single line following the format: 'id1,id2'. We will use this file to grade your exercise using automated tests. Not having this file will result a score of 0 for the code part.
- You may write and use what ever helper functions and files you would like, we will be testing your code using the defined API.
- Please include figures for code-related questions in your pdf file.
- Please submit a single .tar file containing all relevant files (not other formats). The .tar file should contain only the .py files required and ids.txt (no other directories / audio files / etc.)