

Lab Python n° 4_2

Scipy & Digital Signals

Qui serait assez téméraire pour affirmer que nous connaissons et percevons toutes les forces, toutes les ondes et tous les moyens de communications ? « Hubert Reeves »

1. Ouverture d'un fichier .wav

A partir de la documentation de Scipy.io, trouver le moyen d'ouvrir un fichier .wav a partir de jupyter notebook.

Sous quel format le retrouvez vous ?

Quelle est la signification des données obtenues ?

Comment deviner la durée du morceau suivant ces données ?

Dans toute la suite du Lab :

La longueur du tableau contenant le signal est généralement très grande pour être traitée rapidement par un pc personnel. Tronquez les données avant de les traiter si jamais votre ordinateur ne répond pas rapidement.

2. Dessiner le Signal et son spectrogramme

Utilisez la bibliothèque Matplotlib et scipy afin de dessiner les courbes suivantes :

- Le signal obtenu.
- La transformée de fourrier du signal.
- Le spectrogramme du signal.

3. La fenêtre rectangulaire (Boxcar)

Créer manuellement un tableau représentant une fenêtre rectangulaire. Cette fenêtre est obtenue par la fonction :

$F(x) = 1$ si $a < x < b$, 0 sinon. Avec a et b deux entiers.

- Le tableau doit être de même taille que celui qui comporte votre signal.
- La taille de la fenêtre ($b-a$) doit être inférieure au quart de la longueur totale du signal (afin de bien observer les effets du fenêtre sur le signal et sa transformée de fourrier).

4. Troncature du signal

Multiplier le signal avec la fenêtre rectangulaire obtenue dans l'exercice précédent, élément par élément.

Plottez le résultat obtenu.

Faites la transformée de Fourier du signal tronqué.

Plottez le résultat obtenu (Appliquez la fonction [fftshift](#) du module [scipy.fftpack](#) à la transformée obtenue avant de dessiner la fonction. Lisez la documentation de [scipy.fftpack.fft](#) pour comprendre son utilité)

5. Transformée de fourrier de la fenêtre

Faites la transformée de Fourier de la fenêtre rectangulaire.

En utilisant, la fonction [convolve](#) de scipy, convolez la transformée de Fourier du signal (obtenue suite à la question n°2) avec la transformée de Fourier de la fenêtre rectangulaire.

Plottez le résultat obtenu. Que pouvez vous conclure en comparant la courbe obtenue dans cet exercice et dans le précédent ?

6. Filtres (Bonus)

La bibliothèque `scipy.signal` offre une multitude de filtres prédéfinis.

Appliquer un ou plusieurs filtres parmi les filtres disponibles sur le spectre du signal. Faites la transformée de fourrier inverse afin de retrouver votre signal dans le domaine temporel.

Grace à la fonction `wavfile.write`, vous pouvez retranscrire ce signal dans un fichier `.wav` et l'écouter.

Commentez l'effet du ou des filtre(s) utilisé. Pouvez-vous donner une explication mathématiques ?

Lien utiles :

<https://www.online-convert.com> : convertir mp3 en wav

<https://docs.scipy.org/doc/scipy-1.1.0/reference/signal.html> : documentation
scipy.signal

<https://docs.scipy.org/doc/scipy-1.1.0/reference/io.html> : documentation scipy.io