Compte rendu du TP3 - Fourier

Louis Allain

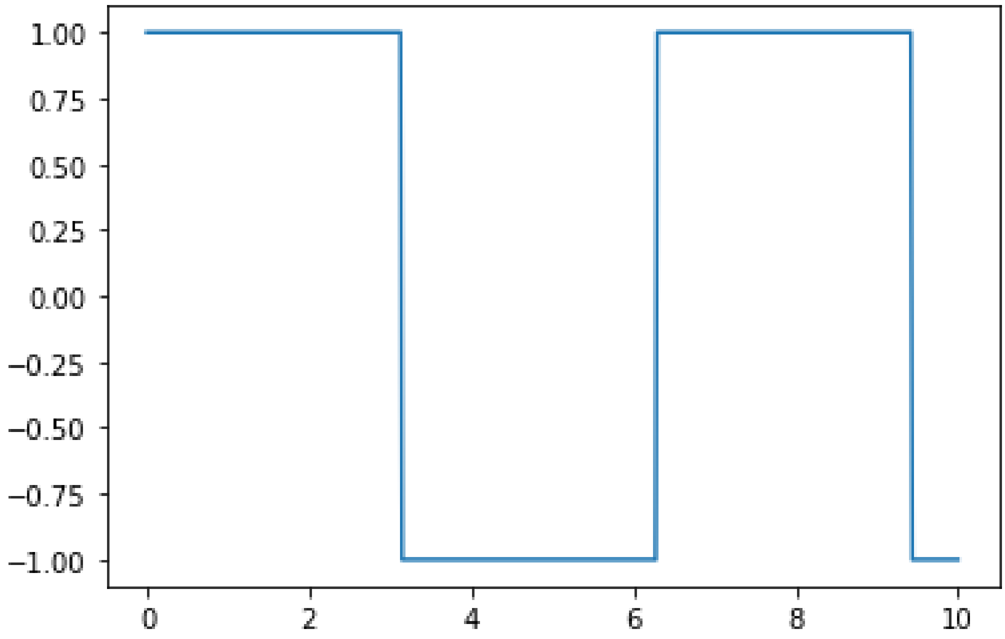
October 8, 2019

1 Séries de Fourier

Réponses aux question :

1. La période du signal est $T_0 = \frac{2}{\pi}$.
   La pulsation du signal $\omega_0 = \frac{2\pi}{T} = 1$ où $T = 2\pi$.
   La fréquence du signal $f_0 = \frac{1}{T_0} = \frac{1}{2\pi}$.

2. On doit prendre f_s, la fréquence d'échantillonnage tel que $f_s \geq 2.Nf_0$ où Nf_0 est la fréquence la plus grande de la dernière harmonique que l'on choisi.



3. Réécriture de s à partir de la composition de s en série de Fourier :

   $$s(t) = \frac{4}{\pi}\sin(2\pi(f_0)t) + \frac{4}{3\pi}\sin(2\pi(3f_0)t) + \frac{4}{5\pi}\sin(2\pi(5f_0)t) + ... + \frac{4}{N\pi}\sin(2\pi(Nf_0)t)$$

   Les conditions que s doit respecter sont les conditions de Dirichlet. La première condition est que s doit être continue par morceaux. La seconde est que s doit être monotone par morceaux. Et enfin, la troisième condition est que s doit être partout intégrable.

4. A faire.

5. Les equations générales permettant de calculer les coeffecients de Fourier : a_k et b_k:

   $a_k = \frac{2}{T} \int_{\frac{-T}{2}}^{\frac{T}{2}} s(t) \cos(nt\frac{2\pi}{T})$

   $b_k = \frac{2}{T} \int_{\frac{-T}{2}}^{\frac{T}{2}} s(t) \sin(nt\frac{2\pi}{T})$.

6. Pour le signal s,

   $a_0 = 0$,
   $a_k = 0$,
   $b_k = \frac{4}{k\pi}$ pour k impair,
   $b_k = 0$ pour k impair.

7. Calcul d'un terme de la série de Fourir en Python :

```
terme = (2*(1-(-1)**k)*np.sin(k*x))/(k*(np.pi));
```

8. Programme Python serieFourier (d'après le cours) :

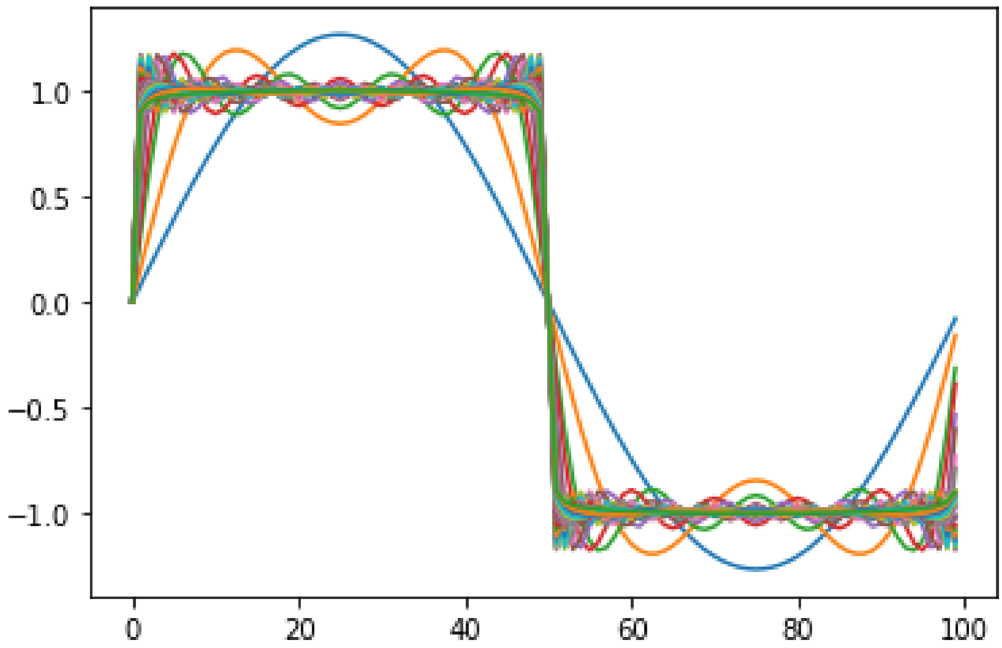
```
def serieFourier (ordre, N) :

R = np.zeros(N)
t = np.linspace(0,((N-1)/N)*2*(np.pi),N)

for i in range(N):
x = t[i]
y = 0

for k in range(1,ordre+1):
terme = (2*(1-(-1)**k)*np.sin(k*x))/(k*(np.pi))
y = y + terme

R[i]=y
return R
```

9. Plus N se rapproche de l'ordre plus on retrouve le signal carré. Mais la convergence vers ce signal est lente et de l'ordre de $\frac{1}{k}$.

10. Affichage de manière superposée les décompositions de s :

```
for i in range(1, 100, 3):
plt.plot(serieFourier(i, 100))
```

    Et voici le résultat graphique :

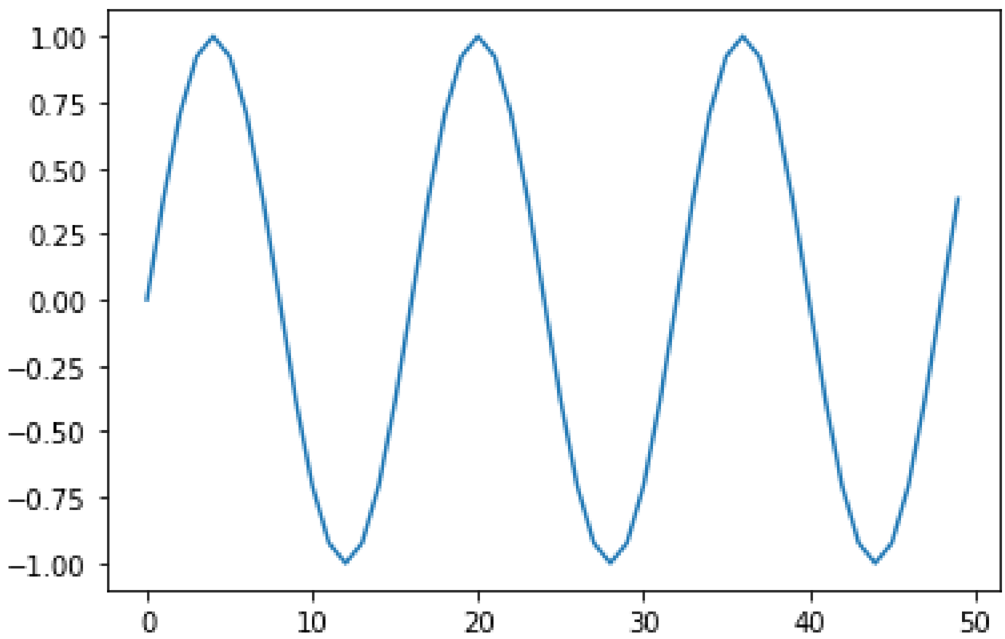


11. On observe le phénomène de Gibbs, il semble que c'est un effet de dépassement du max de s(t).

2 Transformée de Fourier

Réponses aux question :

1. Sinuisoïde de fréquence 16 Hz :

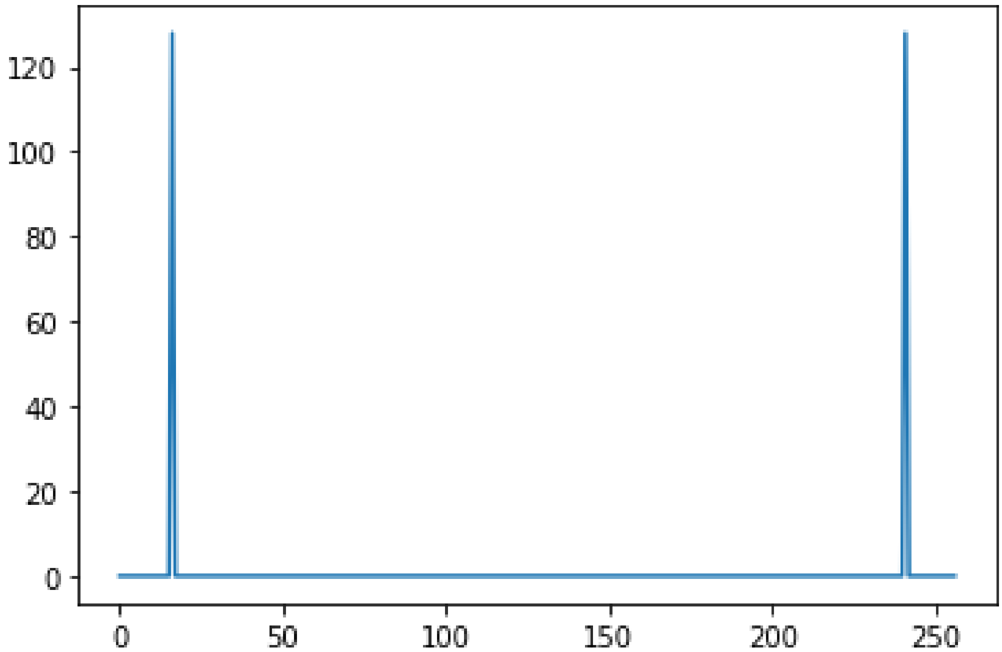


2. On souhaite changer d'espace de représentation en utilisant la transformée de Fourier du signal s.

3. Calcul du spectre :

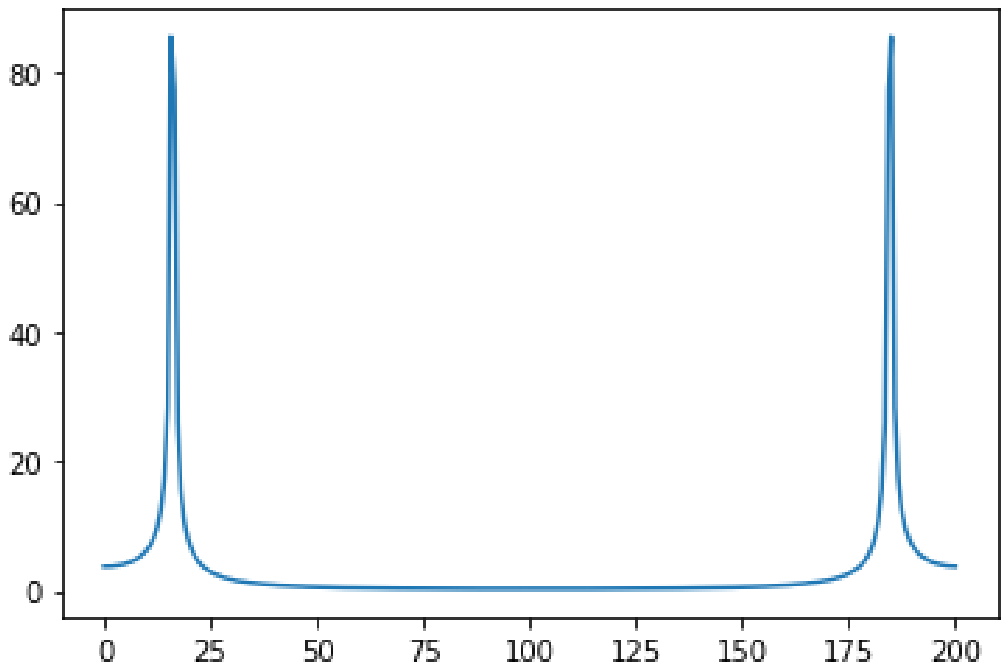
```
TFD = fft(echantillons) # spectre
A = np.absolute(TFD/N) # amplitude
An = A/A.max() # ampmlitude normalisée
P = np.angle(TFD/N) # phase normalisée
```

4. Affichage du spectre :

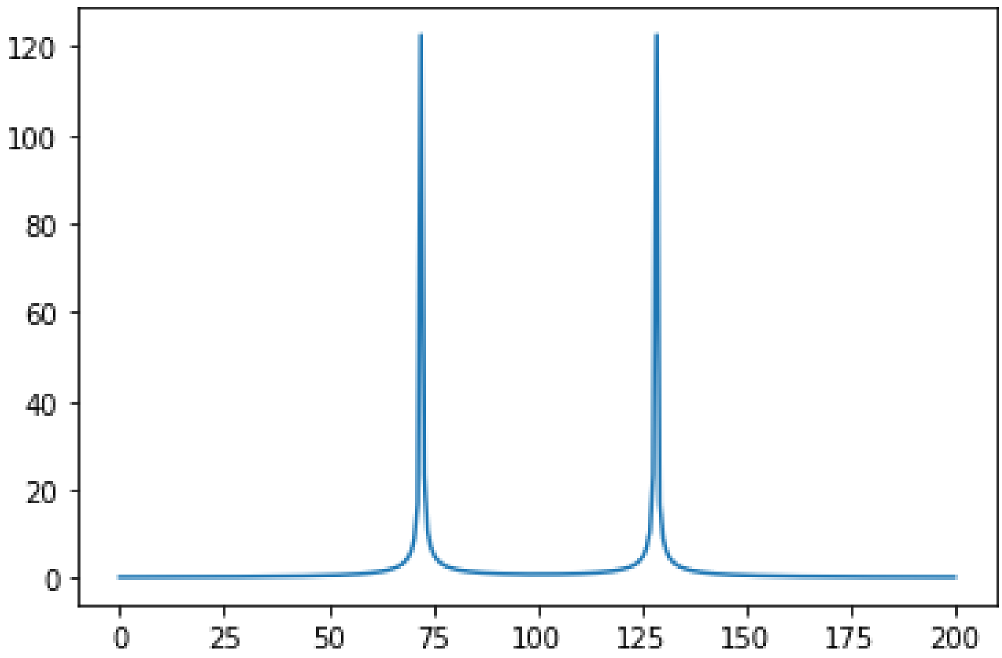


On observe dans le spectre une seconde raie. Cela est du au fait que ce signal est périodique.

5. On observe un élargissement de la base des raies du spectre. Cela est du à la durée fini de l'échantillonnage. En effet, si la durée d'échantillonnage était infinie, les raies seraient de largeur 0.



6.



On peut observer une raie à k=128 mais également une autre raie à une fréquence inférieure. Cela est du à un effet de repliement. En effet, si l'on ne respecte pas le théorême de Shannon-Nyquist (fs ¿¿ fmax), le spectre se replie sur lui même.

7. **`Créer le signal numérique suivant :`**
`x = 3*math.cos(50*pi*t) + 10*math.sin(300*pi*t) - math.sin(100*pi*t)`