

# TP 13 : Propagation d'une onde

Code Capytale : febf-8676535

```
Entrée[1]: import numpy as np
import matplotlib.pyplot as plt
```

## Données

```
Entrée[2]: deltat=[0.00026,0.0004,0.00056,0.0007]
d=[0.05,0.1,0.15,0.2]
```

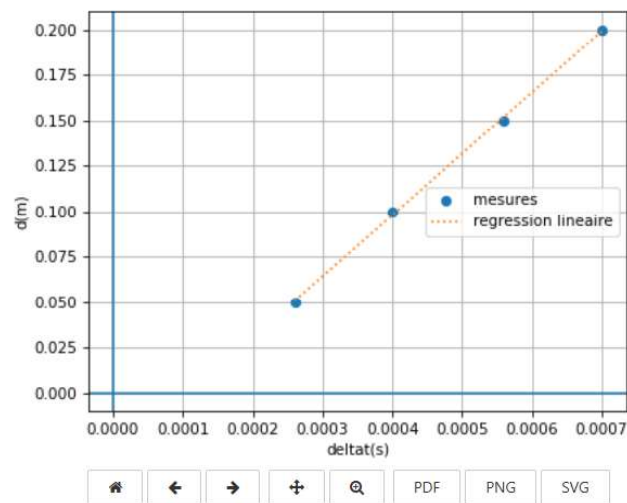
## Régression linéaire

```
Entrée[3]: [a,b]=np.polyfit(deltat,d,1)
print(a,b)
337.5912408759123 -0.03704379562043784
```

## Tracé

```
Entrée[4]: dmodel=[a*t+b for t in deltat]
plt.clf() #pour effacer les tracés précédents
plt.plot(deltat,d,'o',label='mesures')
plt.plot(deltat,dmodel,'-',label='regression lineaire')
plt.legend()
plt.grid()
plt.axhline()
plt.axvline()
plt.xlabel('deltat(s)')
plt.ylabel('d(m)')
plt.show()
```

Figure 1



## Incertitude sur la célérité

```
Entrée[5]: N = 10000
A = [] # Listes des valeurs de a (= célérité)
delta_deltat = 0.000010 #précision sur la mesure de deltat
delta_d = 0.5*10**-3 #précision = 1/2 graduation,0.5 mm
for k in range(N):
    deltatsim = deltat + np.random.uniform(-delta_deltat,delta_deltat,len(deltat))
    dsim = d + np.random.uniform(-delta_d,delta_d,len(d))
    a,b = np.polyfit(deltatsim,dsim,1) #régression linéaire
    A.append(a)
```

```
Entrée[6]: u_a=np.std(A,ddof=1)
print("u(c) = {:.3f} m/s ".format(u_a))
u(c) = 5.955 m/s
```