

Corrigé du TP Informatique 20

Exercice 1

1. On saisit :

```
file=open("data01.csv")
tx=[]
for ligne in file:
    tx.append(float(ligne))
plt.hist(tx,density=True)
plt.show()
```

2. On obtient :

```
>>> np.mean(tx)
9.8145624700746
>>> np.std(tx)
3.730025016936028
```

3. On saisit :

```
def f(x):
    return 1/np.sqrt(2*np.pi*s**2)*np.exp(-(x-m)**2/(2*s))

tt=np.linspace(0,20,100)
ty=[f(t) for t in tt]

plt.hist(tx,density=True)
plt.plot(tt,ty,color='r',lw=3)
plt.show()
```

On observe :

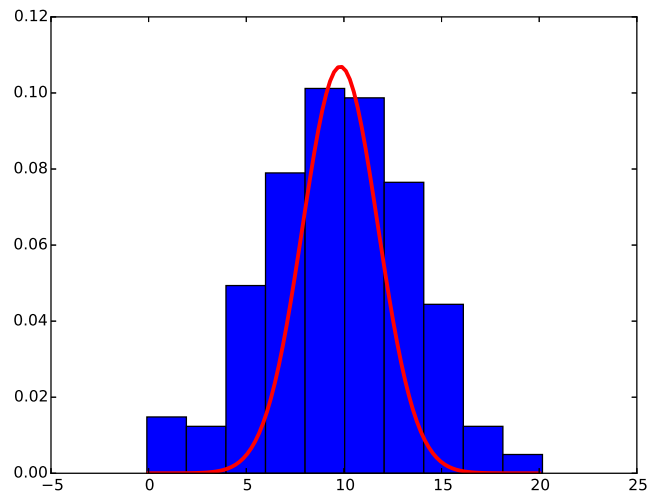


FIGURE 1 – Histogramme et loi normale

La loi normale « approche » (en un sens à préciser) l'histogramme de l'échantillon.

Exercice 2

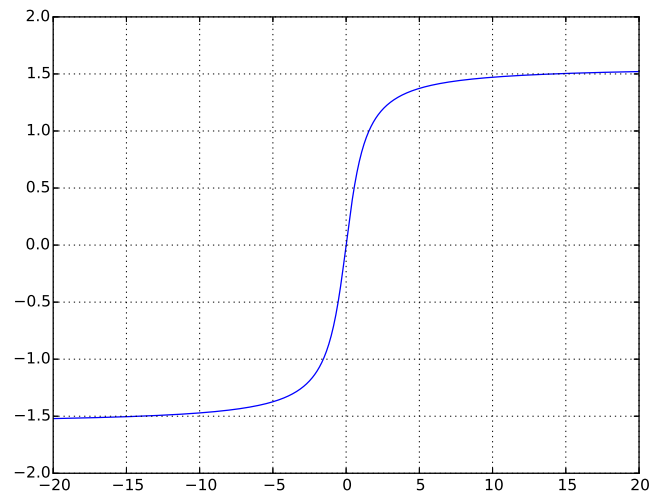
1. On saisit :

```
file=open("data02.csv")
tx,ty=[],[]
for ligne in file:
    aux=ligne.split(";")
    tx.append(float(aux[0]))
    ty.append(float(aux[1]))
```

2. On saisit :

```
plt.plot(tx,ty)
plt.grid();plt.show()
```

On observe :



On reconnaît évidemment le graphe de la fonction Arctan.

Exercice 3

1. On saisit :

```
m,s=10,4
file=open("data03.csv","w")
n=1000
for k in range(n):
    file.write(str(rd.normal(m,s))+ "\n")
file.close()
```

3. On saisit :

```
file=open("data03.csv")
tx=[]
for ligne in file:
    tx.append(float(ligne))

m,s=np.mean(tx),np.std(tx)
print("moyenne = ",m)
print("ecart-type = ",s)
```

On observe :

```
moyenne = 10.0381442069
ecart-type = 3.95386290614
```

La moyenne et l'écart-type de l'échantillon sont des *estimateurs* des grandeurs **m** et **s** de la loi normale simulée par tirage par l'instruction `rd.normal`.

Exercice 4

1. On trouve

$$RC \frac{ds}{dt} + s = e$$

2. On saisit :

```
fichier=open("data.csv")
tt,te=[],[]
for ligne in fichier:
    aux=ligne.split(",")
    tt.append(float(aux[0]))
    te.append(float(aux[1]))
fichier.close()
```

3. On saisit :

```
def debruit(tau,e,t):
    x=[te[0]]
    for k in range(1,len(t)):
        h=t[k]-t[k-1]
        x.append(x[k-1]+h*(e[k]-x[k-1])/tau)
    return x
```

4. On saisit :

```
plt.plot(tt,te)
ts=debruit(.2,te,tt)
plt.plot(tt,ts,'r',linewidth=2)
plt.grid();plt.show()

plt.plot(tt,te)
ts=debruit(.5,te,tt)
plt.plot(tt,ts,'r',linewidth=2)
plt.grid();plt.show()
```

On obtient :

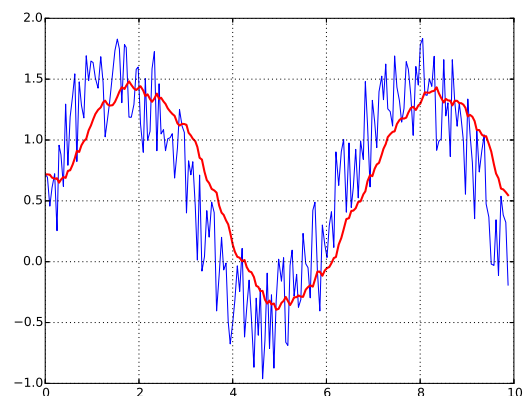
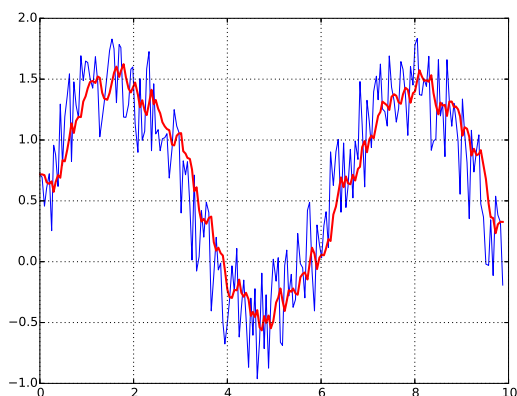


FIGURE 2 – Débruitage par filtre passe-bas RC pour différentes valeurs de τ