

Spécifications de quelques fonctions du module numpy (alias np)

```
for i in range(N):  
    r=np.random.normal(m,s)
```

Ces lignes permettent d'obtenir N valeurs réparties selon la loi normale de moyenne m et d'écart-type s. L'utilisation de la fonction **random.normal** est adaptée lorsqu'on connaît l'**incertitude-type** $u(x)$ sur la grandeur x .

```
for i in range(N):  
    r=np.random.uniform(a,b)
```

Ces lignes permettent d'obtenir N valeurs réparties de manière uniforme entre a et b. L'utilisation de la fonction **random.uniform** est adaptée lorsqu'on connaît la **précision / tolérance** Δx sur la grandeur x .

$M=np.average(M_lst)$ permet de calculer la **moyenne** pour une série de mesures rassemblées dans la liste M_lst.

$S=np.std(M_lst,ddof=1)$ permet de calculer l'**écart type** pour la série de mesures rassemblées dans la liste M_lst.

$D=np.polyfit(x,y,1)$ permet de déterminer la **pente** et l'**ordonnée à l'origine** de la **droite** (polynôme de degré 1) passant au plus près des points de coordonnées (x,y). D[0] donne la pente et D[1] donne l'ordonnée à l'origine.

Spécifications de la fonction hist() du module matplotlib.pyplot (alias plt)

$plt.hist(M_lst,bins='rice')$ permet de représenter l'**histogramme** d'une série de mesures rassemblées dans la liste M_lst avec un nombre de classes k judicieux (*plus précisément qui respecte le critère de Rice : $k = 2n^{1/3}$ avec n le nombre de mesures*).