

Jupyter example\_pi (autosaved)

File Edit View Insert Cell Kernel Widgets Help Hide Code Python 3

# Un document computationnel

Mon ordinateur m'indique que  $\pi$  vaut *approximativement*\*

In [1]:

```
from math import *
print(pi)
```

3.141592653589793

Mais calculé avec la méthode des aiguilles de Buffon ([https://fr.wikipedia.org/wiki/Aiguille\\_de\\_Buffon](https://fr.wikipedia.org/wiki/Aiguille_de_Buffon)), on obtiendrait comme approximation :

In [2]:

```
import numpy as np
N = 1000000
x = np.random.uniform(size=N, low=0, high=1)
theta = np.random.uniform(size=N, low=0, high=pi/2)
2/(sum((x*np.sin(theta))>1)/N)
```

Out[2]: 3.1437198694098765

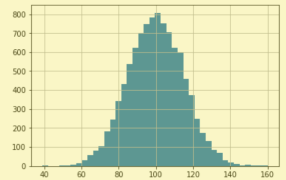
On peut inclure des formules mathématiques comme  $\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$  et des *dessins* qui n'ont rien à voir avec  $\pi$  (si ce n'est une constante de normalisation... ☺).

In [3]:

```
%matplotlib inline
import matplotlib.pyplot as plt

mu, sigma = 100, 15
x = mu + sigma*np.random.randn(10000)

plt.hist(x,40)
plt.grid(True)
plt.show()
```



## Résultats

## Un document computationnel

Mon ordinateur m'indique que  $\pi$  vaut *approximativement*

3.141592653589793

Mais calculé avec la méthode des aiguilles de Buffon, on obtiendrait comme approximation :

```
import numpy as np
N = 1000000
x = np.random.uniform(size=N, low=0, high=1)
theta = np.random.uniform(size=N, low=0, high=pi/2)
2/(sum((x*np.sin(theta))>1)/N)
```

3.1437198694098765

On peut inclure des formules mathématiques comme  $\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$  et des *dessins* qui n'ont rien à voir avec  $\pi$  (si ce n'est une constante de normalisation... ☺).

