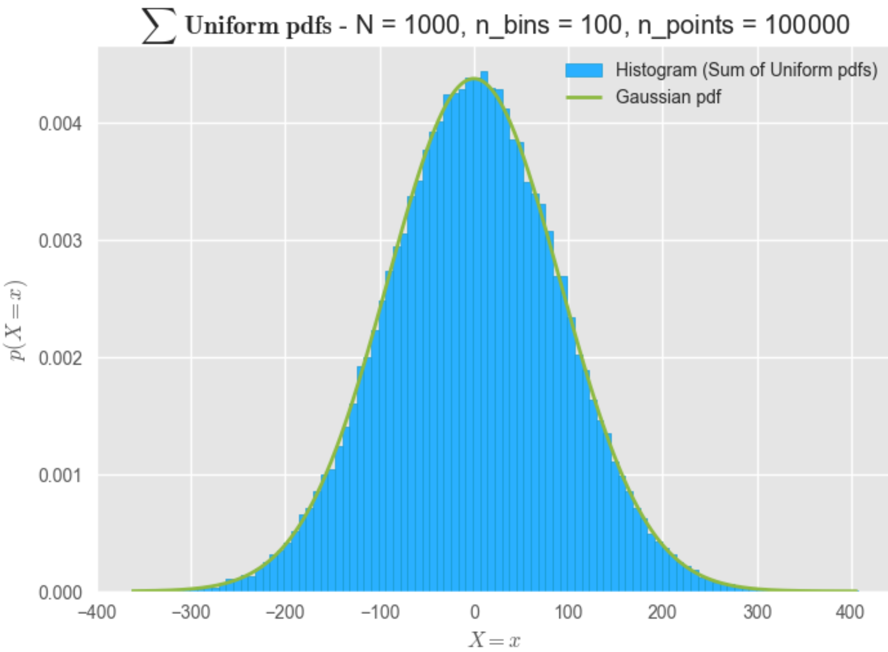
```python
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as stats
plt.style.use('ggplot')
```

Uniform pdf:

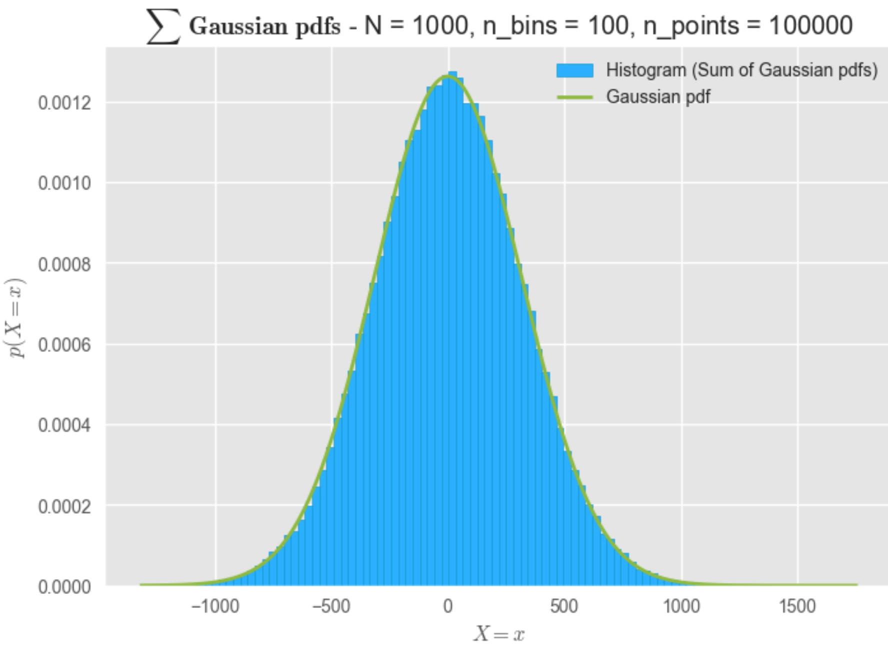
```python
# Definitions
N = 1_000; c = 5; n_points = 100_000; sum_X_vals = []

for n in range(n_points):
    X = np.random.uniform(-c, c, N)
    sum_X_vals.append(sum(X))

# Creating histogram
n_bins = 100
hist, bins, _ = plt.hist(sum_X_vals, density = True, bins = n_bins, label = 'Histogram (Sum
                         facecolor = '#2ab0ff', edgecolor='#169acf', linewidth=0.5) # dimgr

# Analytic parameters (computed by hand)
mu    = 0
sigma = np.sqrt(N*c**2/3)

x = np.linspace(bins.min(), bins.max(), 1_000)
plt.title(r'$\sum$ $\bf{Uniform}$ $\bf{pdfs}$ - ' + f'N = {N}, n_bins = {n_bins}, n_points =
plt.plot(x, stats.norm.pdf(x, mu, sigma), label = 'Gaussian pdf', color = 'C5')
plt.xlabel(r'$X = x$')
plt.ylabel(r'$p(X = x)$')
plt.legend()
plt.show()
```



Gaussian pdf:

```
# Definitions
N = 1_000; n_points = 100_000; sum_X_vals = []

# Initial Gaussian's parameters
mu    = 0
sigma = 10

for n in range(n_points):
    X = np.random.normal(mu, sigma, N)
    sum_X_vals.append(sum(X))

# Creating histogram
n_bins = 100
hist, bins, _ = plt.hist(sum_X_vals, density = True, bins = n_bins, label = 'Histogram (Sum
                         facecolor = '#2ab0ff', edgecolor='#169acf', linewidth=0.5) # dimgr

# Analytic parameters (computed by hand)
mu_X    = 0
sigma_X = np.sqrt(N) * 10

x = np.linspace(bins.min(), bins.max(), 1_000)
plt.title(r'$\sum$ $\bf{Gaussian}$ $\bf{pdfs}$ - ' + f'N = {N}, n_bins = {n_bins}, n_points
```

```python
plt.plot(x, stats.norm.pdf(x, mu_X, sigma_X), label = 'Gaussian pdf', color = 'C5')
plt.xlabel(r'$X = x$')
plt.ylabel(r'$p(X = x)$')
plt.legend()
plt.show()
```

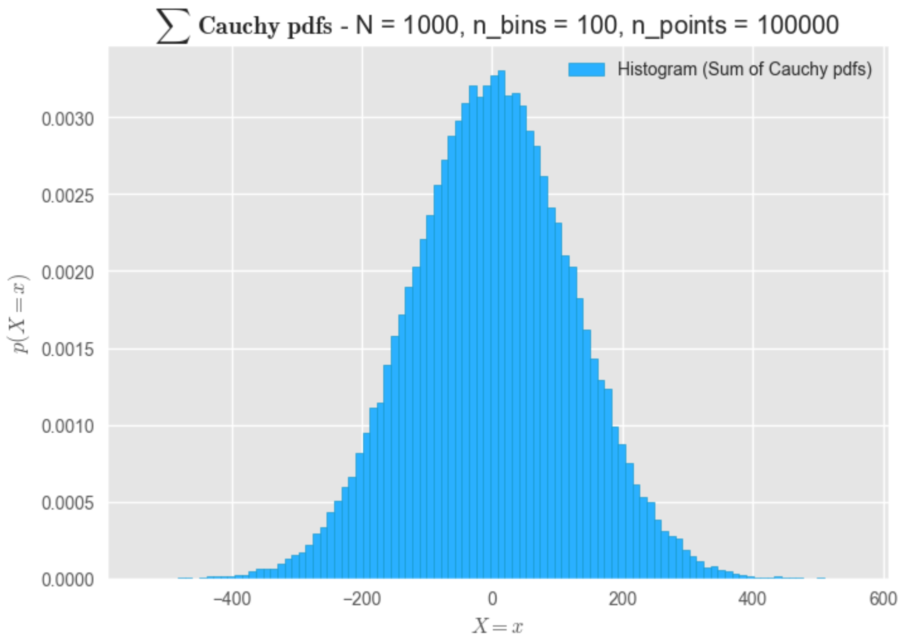


Lorentzian (Cauchy) pdf:

- Here, the CLT is NOT valid!

```python
# Definitions
N = 1_000; n_points = 100_000; sum_X_vals = []

# Standard Cauchy distribution's parameters. Not used in the code. Just here for reference/
x0    = 0
gamma = 1

for n in range(n_points):
    X = np.random.standard_cauchy(N)
    X = X[(X>-25) & (X<25)] # Truncate distribution so it plots well
    sum_X_vals.append(sum(X))

# Creating histogram
n_bins = 100
hist, bins, _ = plt.hist(sum_X_vals, density = True, bins = n_bins, label = 'Histogram (Sum
```

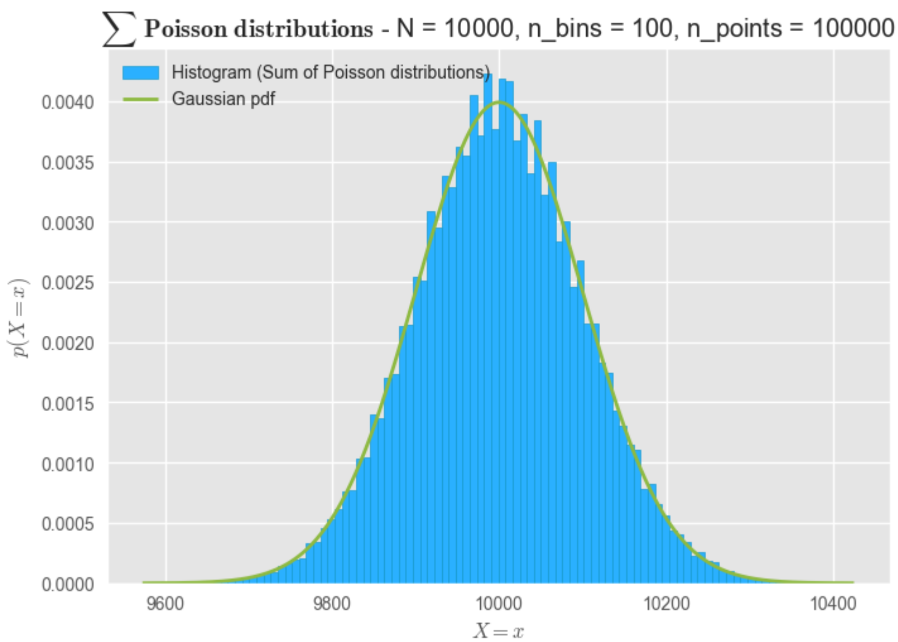
```
                         facecolor = '#2ab0ff', edgecolor='#169acf', linewidth=0.5) # dimgr

x = np.linspace(bins.min(), bins.max(), 1_000)
plt.title(r'$\sum$ $\bf{Cauchy}$ $\bf{pdfs}$ - ' + f'N = {N}, n_bins = {n_bins}, n_points =
plt.xlabel(r'$X = x$')
plt.ylabel(r'$p(X = x)$')
plt.legend()
plt.show()
```



Poisson distribution:

```
# Definitions
N = 10_000; n_points = 100_000; sum_X_vals = []

# Poisson's lambda parameter
lambda_par = 1

for n in range(n_points):
    X = np.random.poisson(lambda_par, N)
    sum_X_vals.append(sum(X))

# Creating histogram
n_bins = 100
hist, bins, _ = plt.hist(sum_X_vals, density = True, bins = n_bins, label = 'Histogram (Sum
                         facecolor = '#2ab0ff', edgecolor='#169acf', linewidth=0.5) # dimgr
```

```
# Analytic parameters (computed by hand)
mu    = N * lambda_par
sigma = np.sqrt(N * lambda_par)

x = np.linspace(bins.min(), bins.max(), 1_000)
plt.title(r'$\sum$ $\bf{Poisson}$ $\bf{distributions}$ - ' + f'N = {N}, n_bins = {n_bins}, n
plt.plot(x, stats.norm.pdf(x, mu, sigma), label = 'Gaussian pdf', color = 'C5')
plt.xlabel(r'$X = x$')
plt.ylabel(r'$p(X = x)$')
plt.legend()
plt.show()
```



As can be seen from the plots, the Central Limit Theorem (CLT) is verified, except for the Cauchy distribution. In fact, this distribution's mean and variance are not defined. As such, we cannot expect the CLT to hold.