第三次作品：Python 函數繪製的觀念與技巧

學號：411073088

姓名：陳敬翰

作品目標

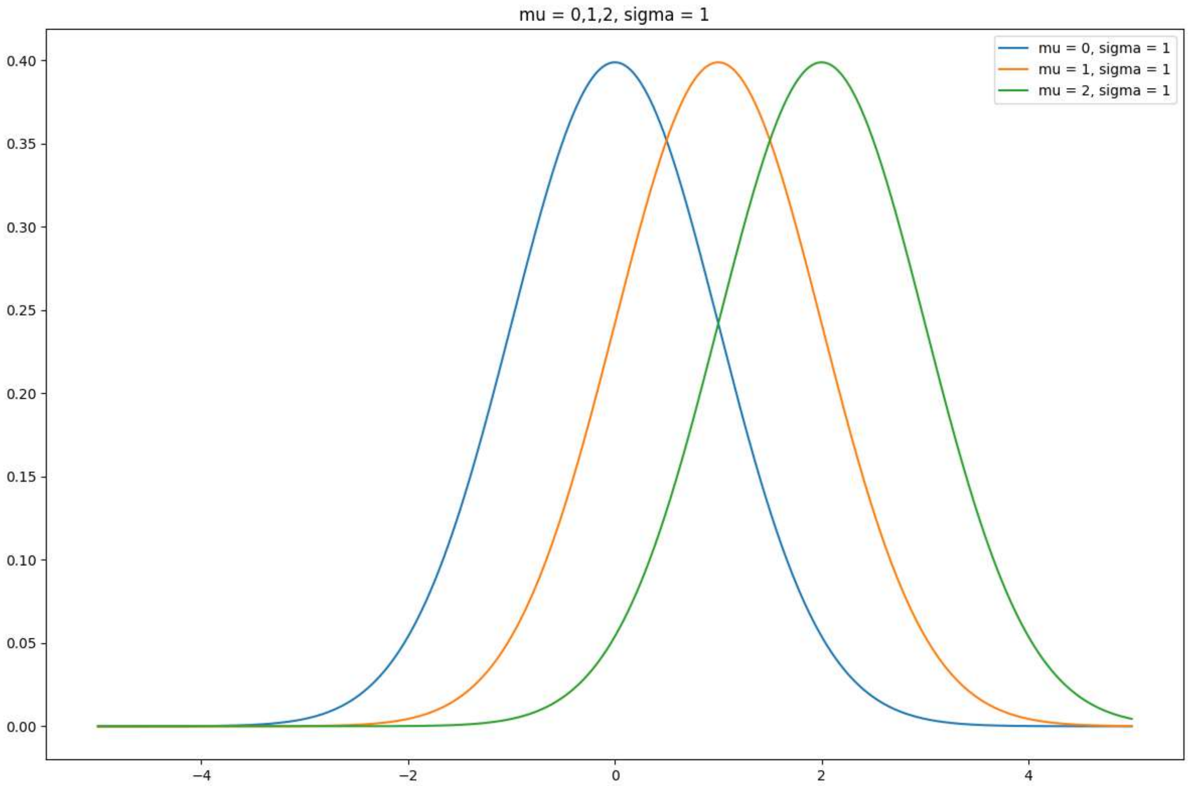
繪製曾學過的分配函數，含連續與離散型。

繪製函數：連續型分配至少包括常態、卡方、T、Beta、F 等五種，或再加上其他自選的分配。利用改變分配函數的參數，觀察其分配函數的「樣貌」；畫出所有可能的「形狀」並說明（或標示）與參數間的關係

常態分配

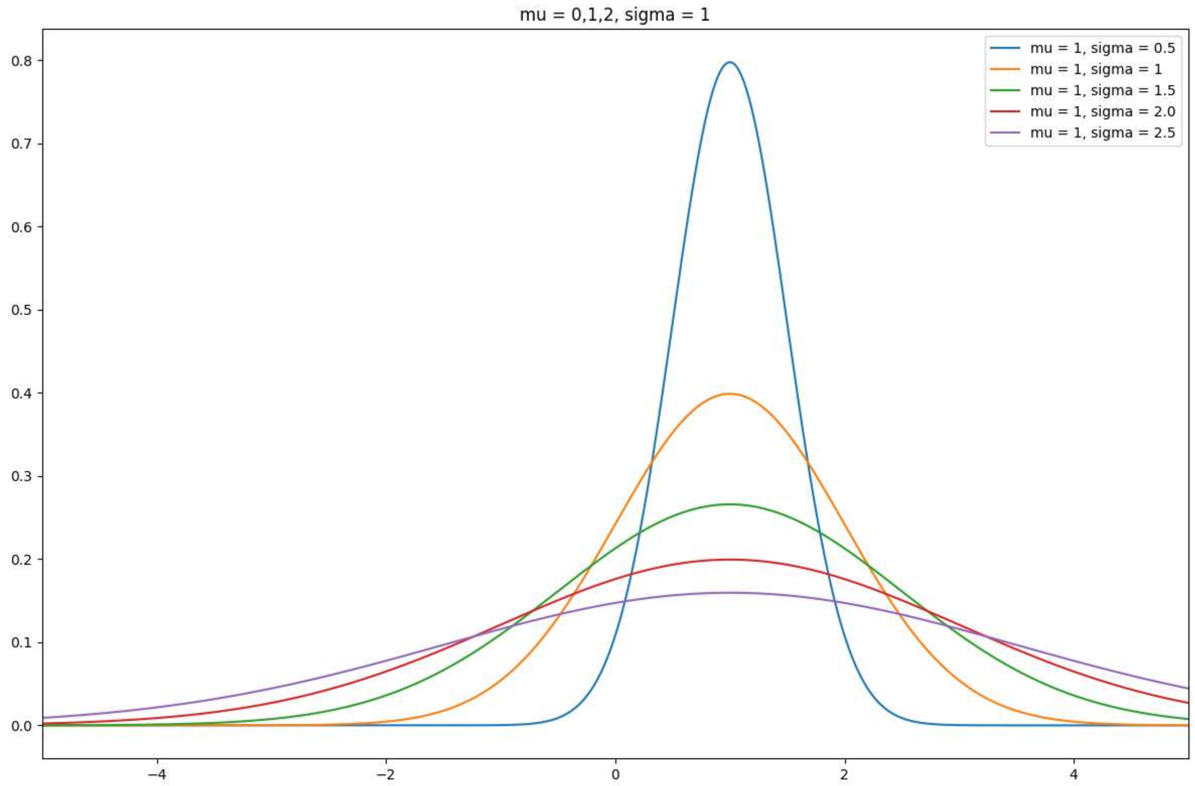
In [ ]:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm

mu_values = [0, 1, 2]

fig, ax = plt.subplots(figsize=(12, 8))

x = np.linspace(-5, 5, 1000)

for i, mu in enumerate(mu_values):
    y = norm.pdf(x, loc=mu, scale=1)
    ax.plot(x, y, label=f'mu = {mu}, sigma = 1')
ax.set_title(f'mu = 0,1,2, sigma = 1')
plt.tight_layout()
plt.legend()
plt.show()
```



In [ ]:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm

sigma_values = [0.5, 1, 1.5, 2.0, 2.5]

fig, ax = plt.subplots(figsize=(12, 8))

x = np.linspace(-5, 5, 1000)

for i, sigma in enumerate(sigma_values):
    y = norm.pdf(x, loc=1, scale=sigma)
    ax.plot(x, y, label=f'mu = 1, sigma = {sigma}')
ax.set_title(f'mu = 0,1,2, sigma = 1')
plt.xlim(-5, 5)
plt.tight_layout()
plt.legend()
plt.show()
```



作品結論：

常態分配的機率密度函數的PDF是對稱的，呈鐘形曲線，中心峰值位於平均數處，並具有單一峰。

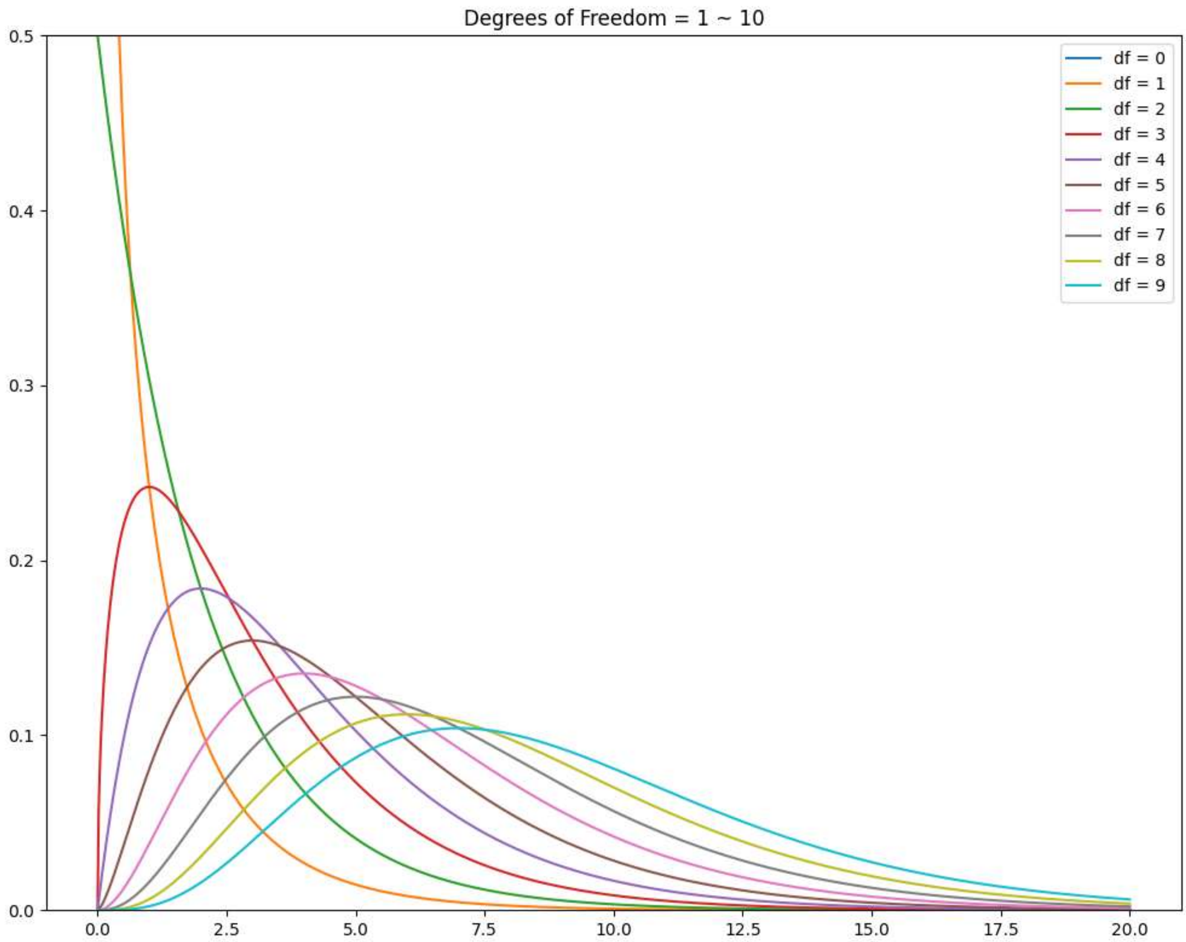
標準差控制了曲線的寬度。較大的標準差導致曲線變寬，較小的標準差導致曲線變窄。

常態分配的曲線永不觸及 x 軸，但它可以延伸到負無窮遠和正無窮遠。

大約 68% 的數據位於平均數的一個標準差範圍內，約 95% 位於兩個標準差範圍內，約 99.7% 位於三個標準差範圍內。這是常被稱為「68-95-99.7規則」的性質。

X^2分配

In [ ]:
```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import chi2

degrees_of_freedom = np.arange(10)

fig, ax = plt.subplots(figsize=(10, 8))

x = np.linspace(0, 20, 1000)

for i, df in enumerate(degrees_of_freedom):
    y = chi2.pdf(x, df)

    ax.plot(x, y, label=f'df = {df}')
ax.set_title(f'Degrees of Freedom = 1 ~ 10')

plt.ylim(0, 0.5)
plt.tight_layout()
plt.legend()
plt.show()
```



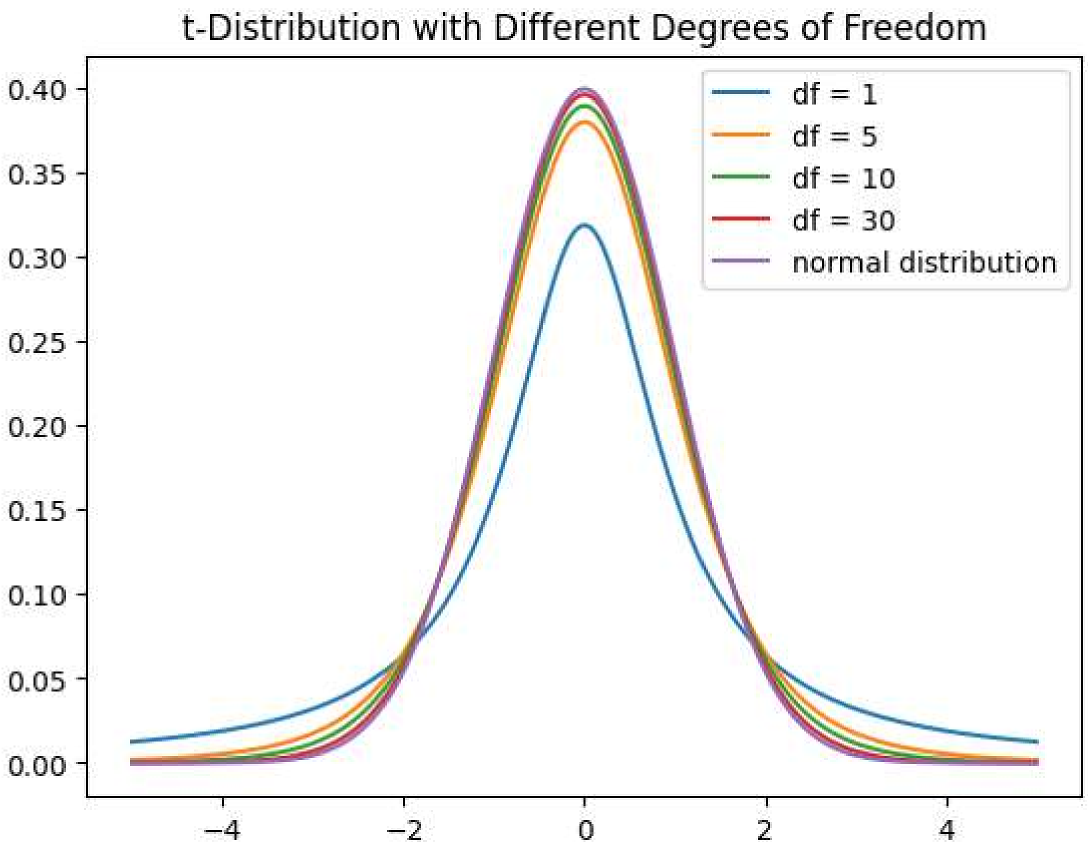
作品結論：

卡方分配的機率密度函數PDF呈正偏斜（右偏）形狀，隨著自由度的增加逐漸趨近於常態分配。

T分配

In [ ]:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import t
from scipy.stats import norm

degrees_of_freedom = [1,5,10,30]

x = np.linspace(-5, 5, 1000)

for df in degrees_of_freedom:
    pdf = t.pdf(x, df)
    plt.plot(x, pdf, label=f'df = {df}')

y = norm.pdf(x, loc=0, scale=1)
plt.plot(x, y, label=f'normal distribution')

plt.title('t-Distribution with Different Degrees of Freedom')
plt.legend()

plt.show()
```



作品結論：

t分配（Student's t-Distribution）是一種連續概率分配，具有以下特性：

形狀：t分配的機率密度函數PDF呈對稱的鐘形形狀，類似標準常態分配，但是尾部較厚。分佈的形狀受自由度的影響。

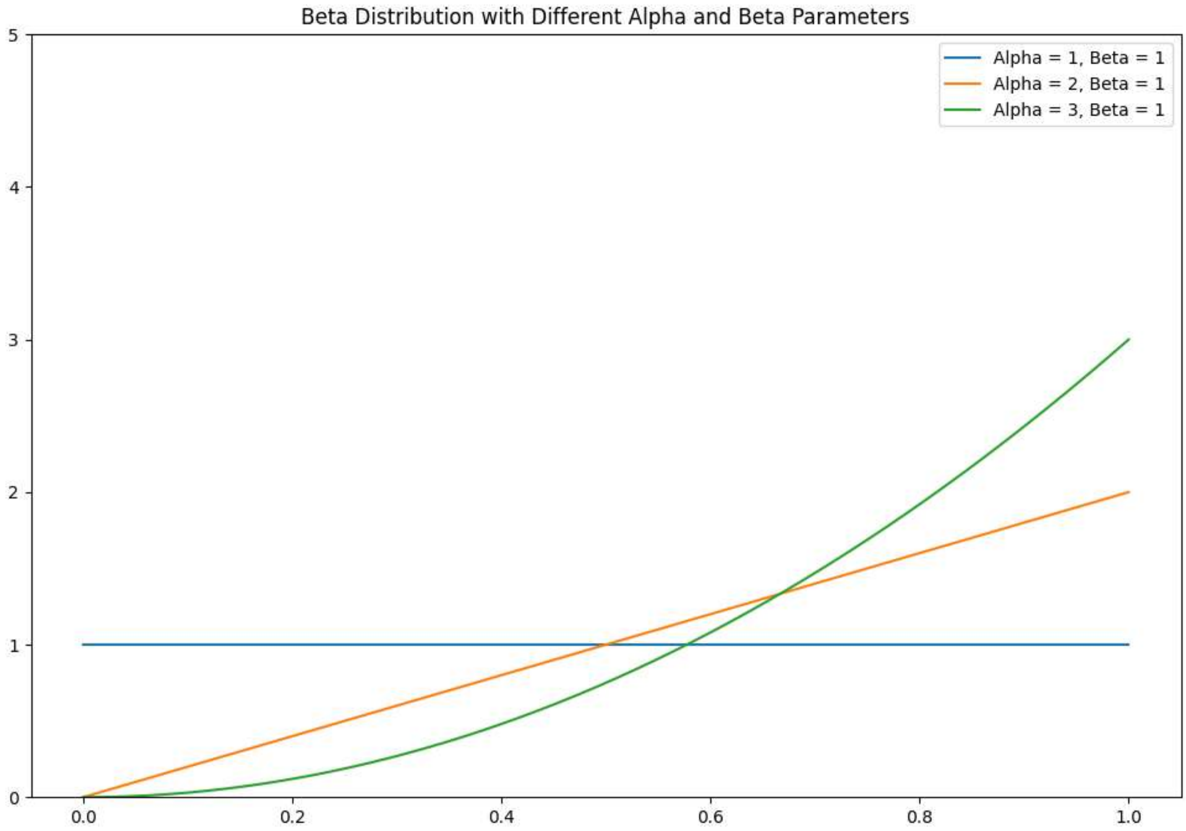
中心極限定理：當自由度足夠大（通常 df > 30）時，t分配近似於標準正態分配，因此，可以應用正態分配的相關性質進行分析。

beta分配

In [ ]:

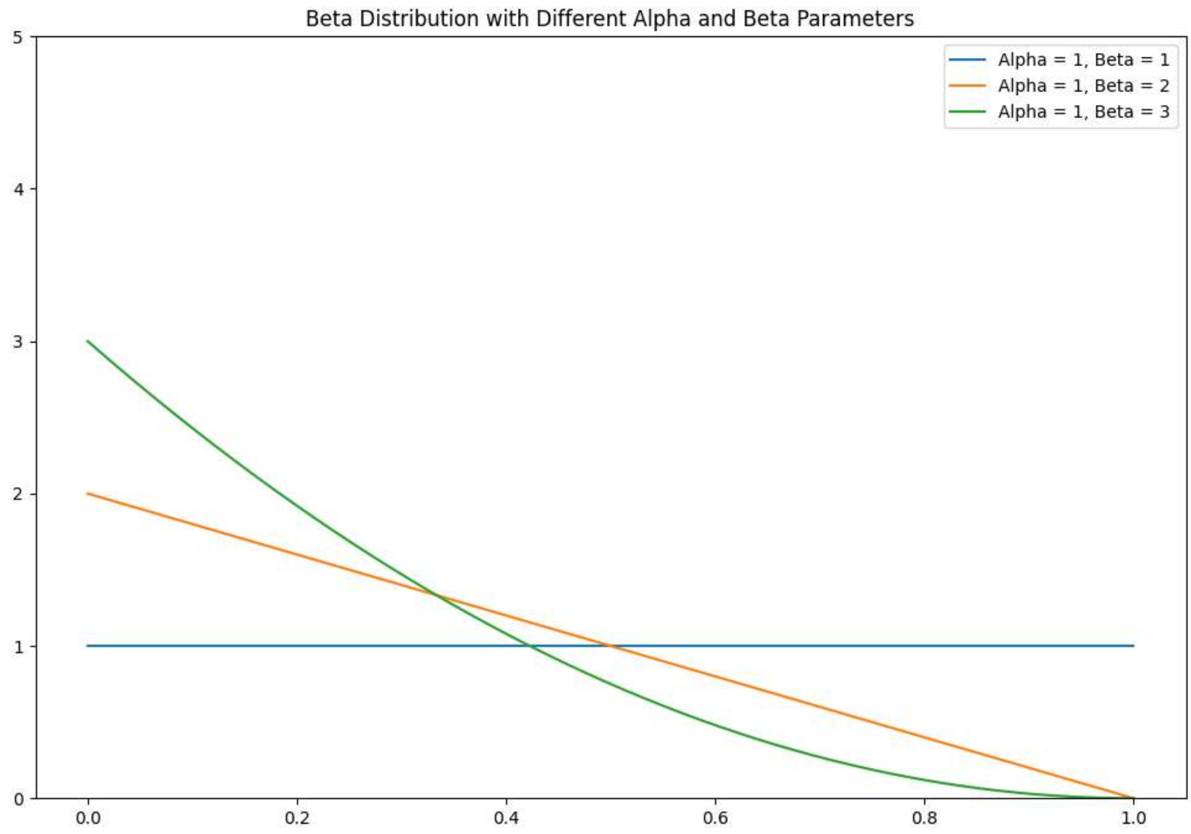
```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import beta

# 設定 Beta 分佈的參數
a_values = [1, 2, 3]  # 可根據需要調整
b_values = [1, 2, 3]  # 可根據需要調整

# 產生一組 x 值的範圍
x = np.linspace(0, 1, 1000)

# 繪製不同參數下的 Beta 分佈
plt.figure(figsize=(12, 8))
for a in a_values:
    pdf = beta.pdf(x, a, 1)
    label = f'Alpha = {a}, Beta = 1'
    plt.plot(x, pdf, label=label)

# 添加標題和圖例
plt.title('Beta Distribution with Different Alpha and Beta Parameters')
```

```
plt.legend(loc='upper right')
plt.ylim(0, 5)  # 可根據需要調整
# 顯示圖形
plt.show()
```



In [ ]:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import beta

# 設定 Beta 分佈的參數
a_values = [1, 2, 3]  # 可根據需要調整
b_values = [1, 2, 3]  # 可根據需要調整

# 產生一組 x 值的範圍
x = np.linspace(0, 1, 1000)

# 繪製不同參數下的 Beta 分佈
plt.figure(figsize=(12, 8))
for b in b_values:
    pdf = beta.pdf(x, 1, b)
    label = f'Alpha = 1, Beta = {b}'
    plt.plot(x, pdf, label=label)

# 添加標題和圖例
plt.title('Beta Distribution with Different Alpha and Beta Parameters')
plt.legend(loc='upper right')
plt.ylim(0, 5)  # 可根據需要調整
# 顯示圖形
plt.show()
```



作品結論：

Beta分配是一種連續概率分配，其特性如下：

參數：Beta分配有兩個參數，通常用α（alpha）和β（beta）表示。這些參數控制了分佈的形狀。α和β必須是正數。

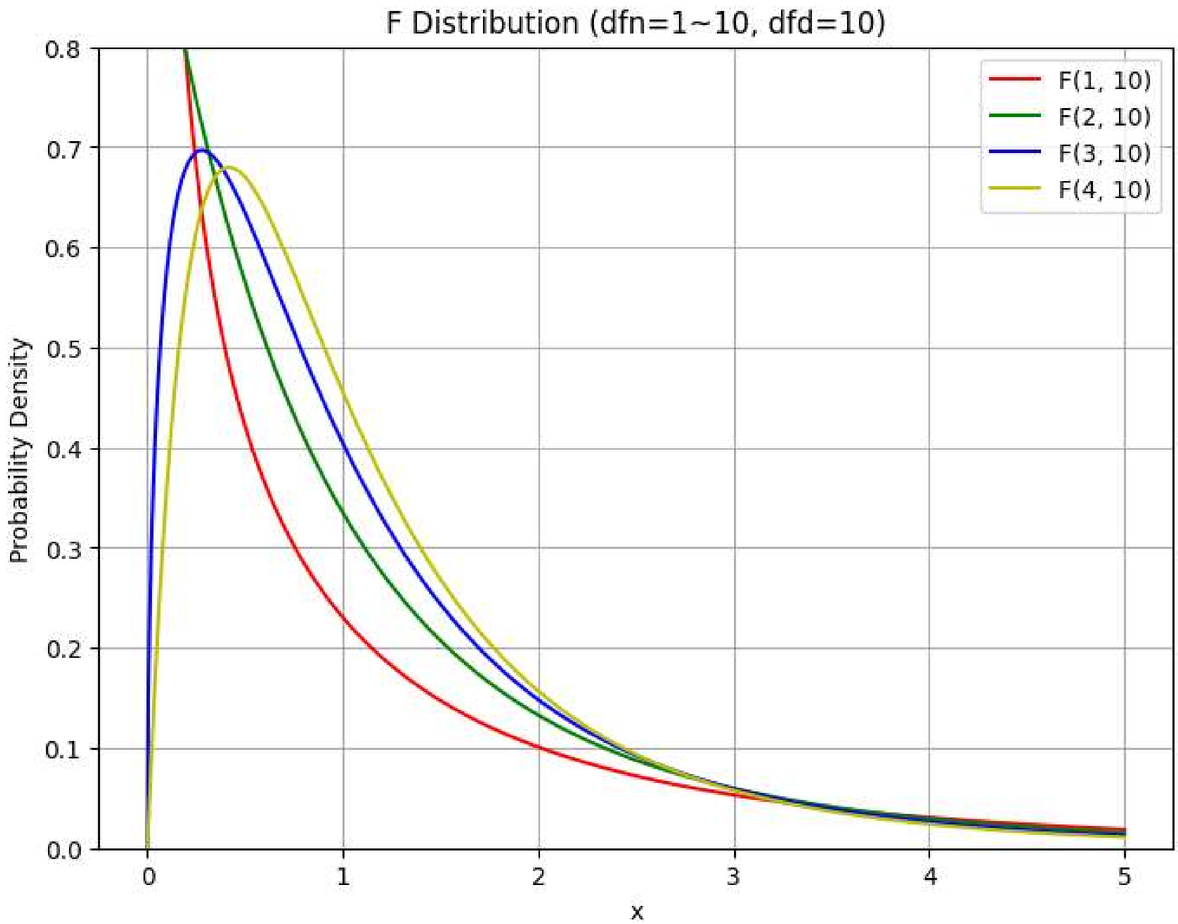
定義域：Beta分配的定義域在區間[0, 1]之間。這使得它特別適合表示隨機變數的比例或機率。

形狀：Beta分配的形狀取決於α和β的值。當α和β都等於1時，Beta分配變為均勻分配，即在[0, 1]區間內的所有值等可能。當α和β的值不相等時，分佈呈現非均勻的形狀，當α>β時，分佈向左偏斜，當α<β時，分佈向右偏斜。

F分配

In [ ]:
```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import f

# 自由度參數
dfn = np.arange(1,5)  # 分子自由度
dfd = 10  # 分母自由度

# 生成 x 值的範圍
x = np.linspace(0, 5, 1000)
plt.figure(figsize=(8, 6))

color = ['r', 'g', 'b', 'y', 'k', 'c', 'm', 'orange', 'purple', 'brown']

# 計算 F 分佈的概率密度函數
for i in range(len(dfn)):
```

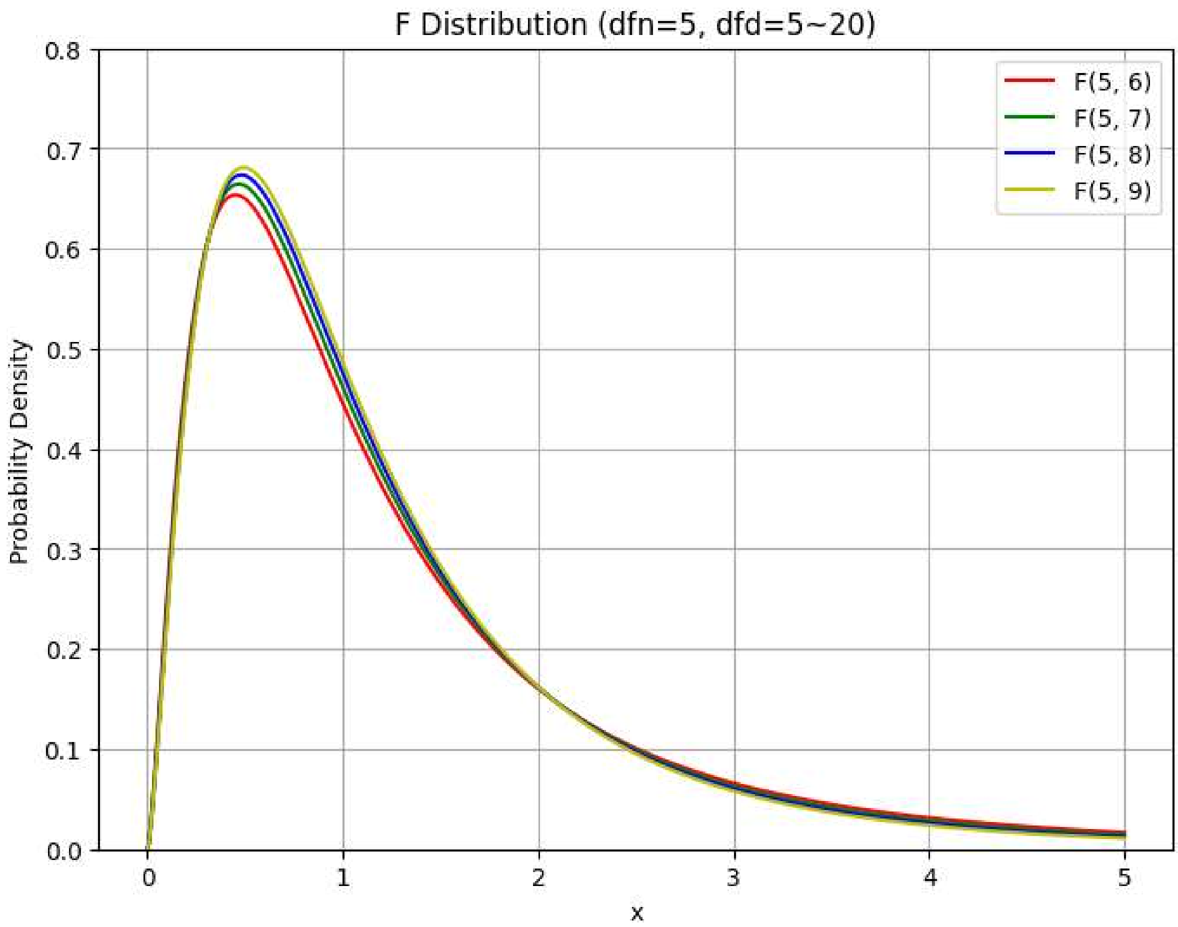
```
    pdf = f.pdf(x, dfn[i], dfd)
    plt.plot(x, pdf, label=f'F({dfn[i]}, {dfd})', color=color[i])

# 繪製概率密度函數
plt.title(f'F Distribution (dfn=1~10, dfd={dfd})')
plt.xlabel('x')
plt.ylabel('Probability Density')
plt.legend()
plt.grid()
plt.ylim(0, 0.8)
plt.show()
```



In [ ]:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import f

# 自由度參數
dfn = 5  # 分子自由度
dfd = np.arange(6,10)  # 分母自由度

# 生成 x 值的範圍
x = np.linspace(0, 5, 1000)
plt.figure(figsize=(8, 6))

color = ['r', 'g', 'b', 'y', 'k', 'c', 'm', 'orange', 'purple', 'brown', 'pink', 'g

# 計算 F 分佈的概率密度函數
for i in range(len(dfd)):
    pdf = f.pdf(x, dfn, dfd[i])
    plt.plot(x, pdf, label=f'F({dfn}, {dfd[i]})', color=color[i])

# 繪製概率密度函數
plt.title(f'F Distribution (dfn=5, dfd=5~20)')
plt.xlabel('x')
```

```
plt.ylabel('Probability Density')
plt.legend()
plt.grid()
plt.ylim(0, 0.8)
plt.show()
```



作品結論：

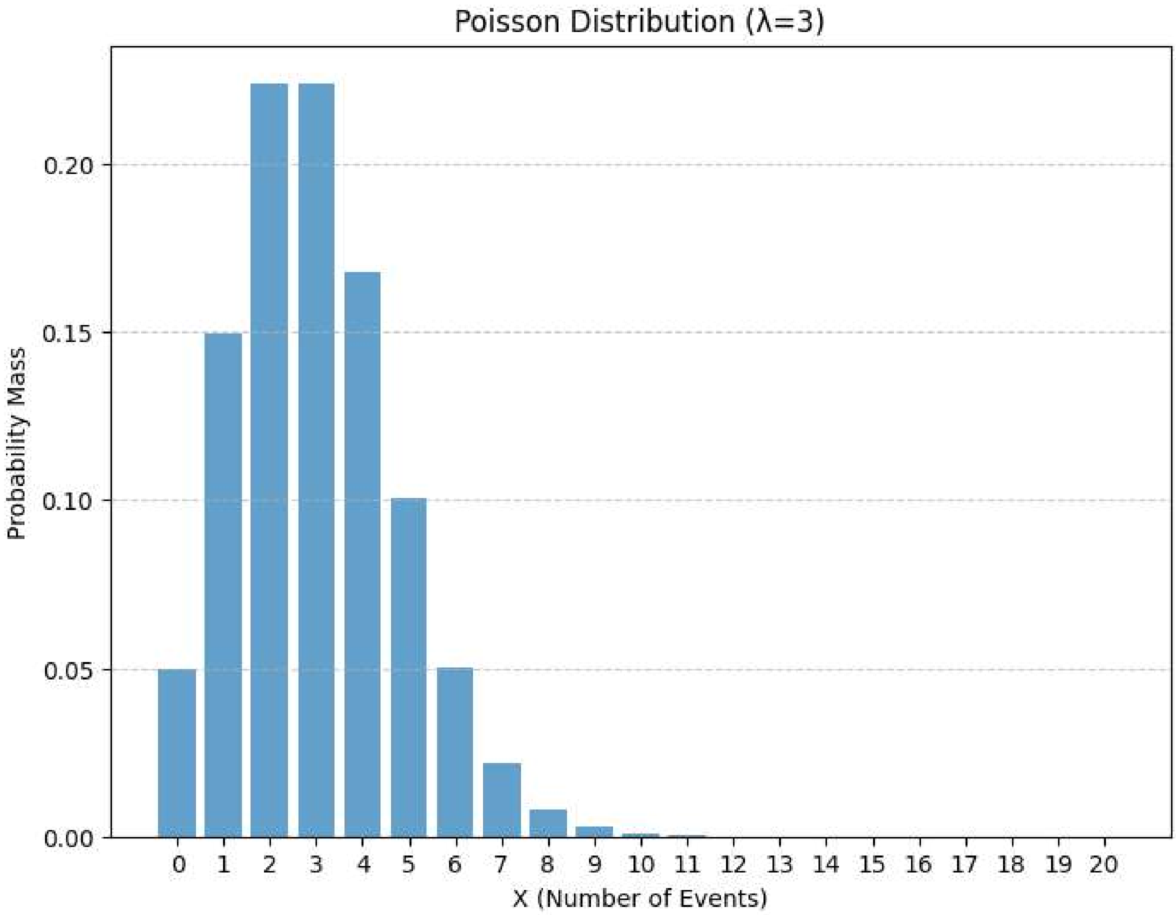
F分配是右偏的分佈，其形狀受兩個正數參數的影響，分別稱為自由度（degrees of freedom），通常表示為 df1 和 df2。F分配的形狀取決於這兩個自由度的值。當 df1 和 df2 的值較大時，分佈趨向於正態分佈，當 df1 和 df2 的值較小時，分佈呈現右偏的形狀，右偏程度取決於 df1 和 df2 的相對大小，較大的分子自由度（df1）和較小的分母自由度（df2）會導致更明顯的右偏。

poisson 分配

In [ ]:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import poisson

# 平均事件率（Poisson分配的參數 λ）
lambda_ = 3

# 生成 x 值的範圍（假設在 0 到 20 之間）
x = np.arange(0, 21)

# 計算 Poisson 分佈的 PMF
pmf = poisson.pmf(x, lambda_)

# 繪製 PMF 圖
```

```
plt.figure(figsize=(8, 6))
plt.bar(x, pmf, align='center', alpha=0.7)
plt.xticks(x)
plt.title(f'Poisson Distribution (λ={lambda_})')
plt.xlabel('X (Number of Events)')
plt.ylabel('Probability Mass')
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.show()
```



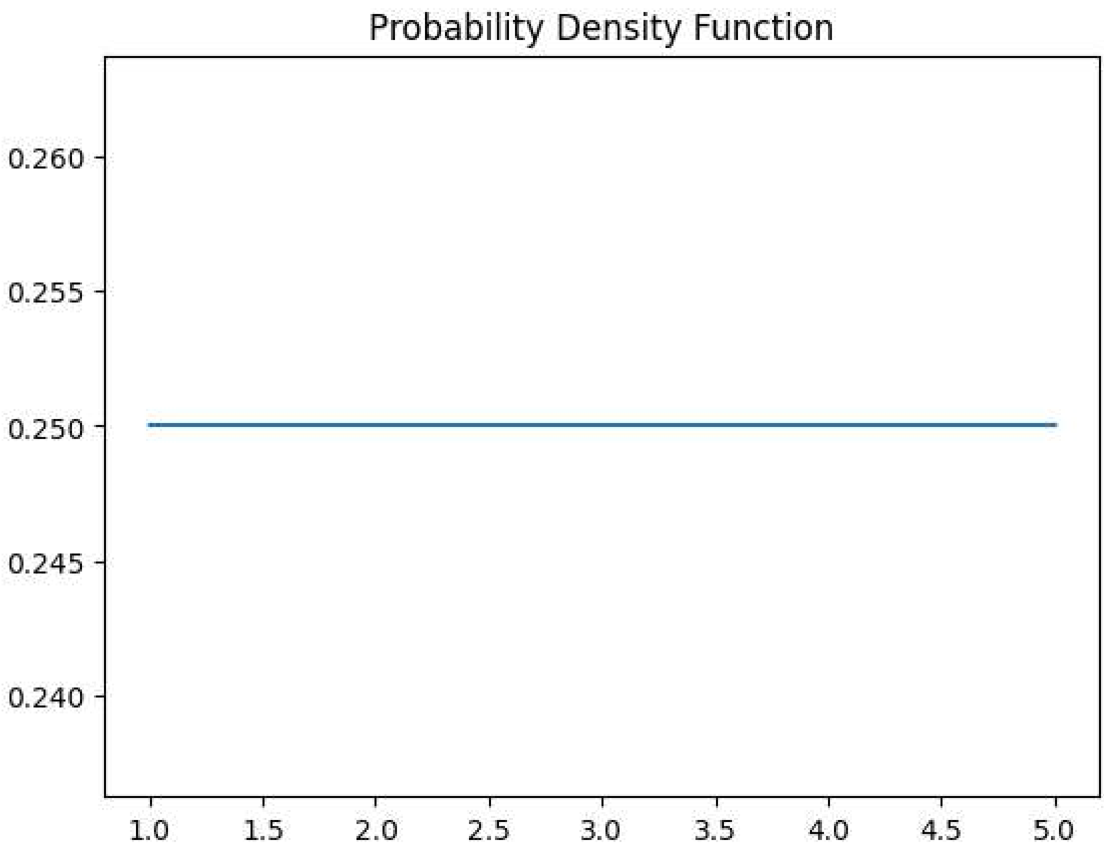
作品結論：

右偏分布：possion分布通常是右偏的，即發生次數較少的情況較常見。

uniform 分配

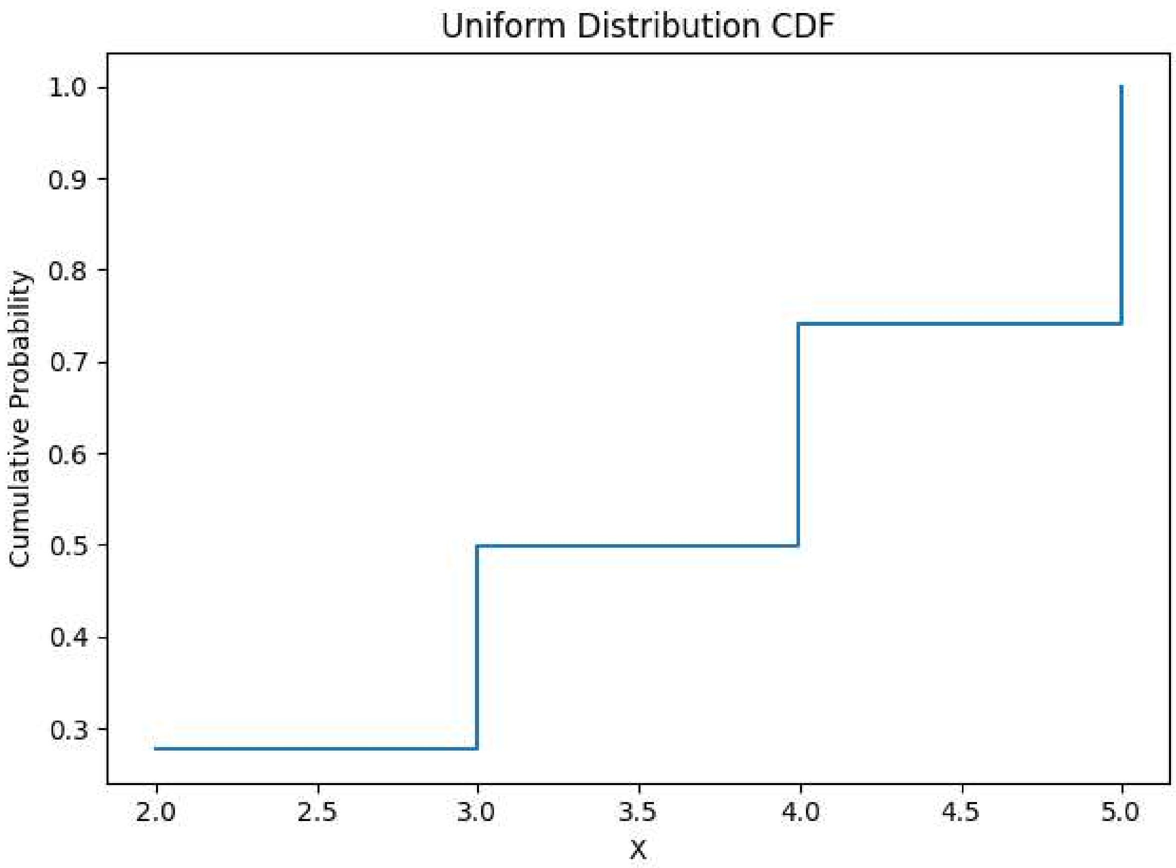
In [ ]:

```
# 繪製 PMF (stem 圖)
import numpy as np
import matplotlib.pyplot as plt

a = 1  # assuming a value for a
b = 5  # assuming a value for b

pro = 1 / (b - a)
x = np.linspace(a, b, 100)  # create an array of 100 points between a and b
y = np.full_like(x, pro)  # create an array of the same shape as x, filled with the
plt.plot(x, y)
plt.title('Probability Density Function')
plt.show()
```



In [ ]:

```
import numpy as np
import matplotlib.pyplot as plt

# 定義均勻分配的參數
a = 1  # 分配的下界
b = 5  # 分配的上界

plt.title("Uniform Distribution PMF")
plt.xlabel("X")
plt.ylabel("Probability")

# 繪製 CDF (stairs 圖)
uniform_data = np.random.uniform(a, b, 1000)

counts, bin_edges = np.histogram(uniform_data, bins=4, density=True)
cdf = np.cumsum(counts) * np.diff(bin_edges)
plt.plot(bin_edges[1:], cdf, drawstyle='steps-post')
plt.title("Uniform Distribution CDF")
plt.xlabel("X")
plt.ylabel("Cumulative Probability")

plt.tight_layout()
plt.show()
```



作品結論：

可能性相同：在均勻分布中，所有的數值在區間內出現的機會是相等的，每個數值都有相同的概率密度。

概率密度函數：均勻分布的概率密度函數（PDF）是一條水平線，其高度為 1 / (b - a)，在區間 [a, b] 內的值為常數。

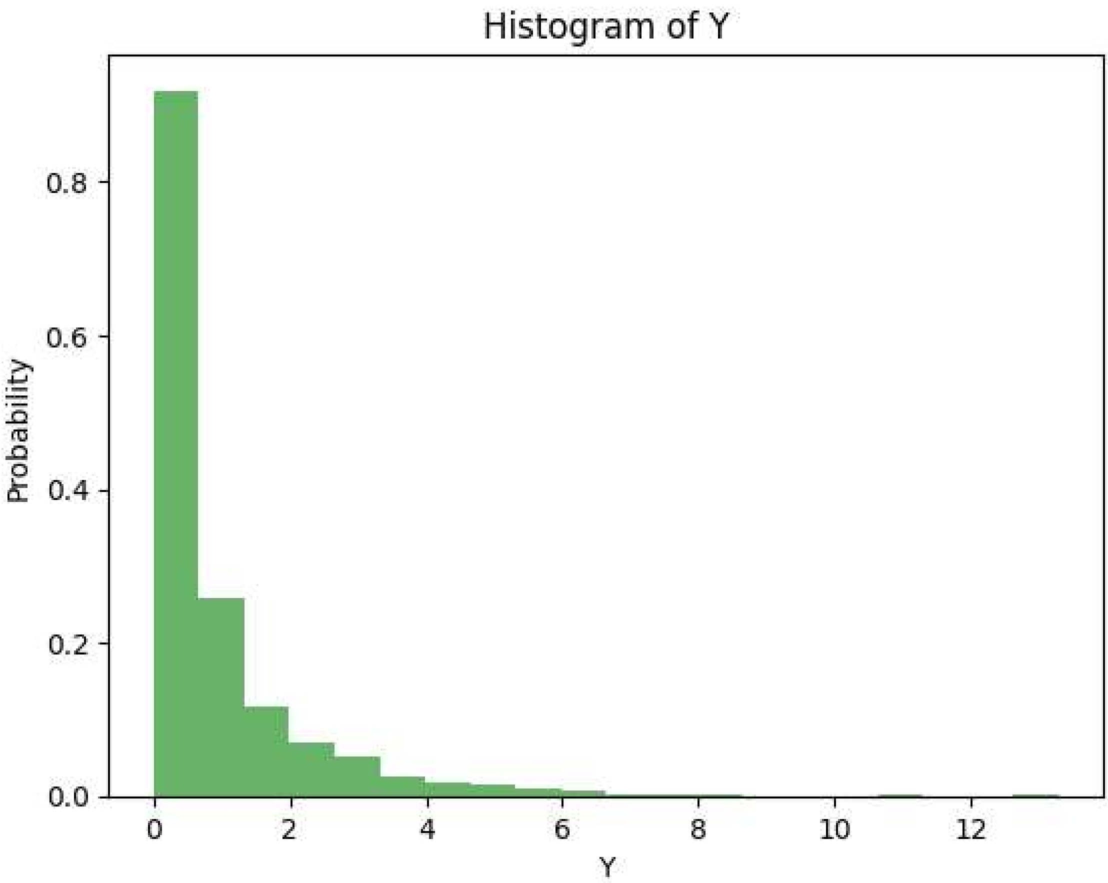
累積分佈函數：均勻分布的累積分佈函數（CDF）是一條斜線，從 a 開始，斜率為 1 / (b - a)，在區間 [a, b] 內均勻增加。

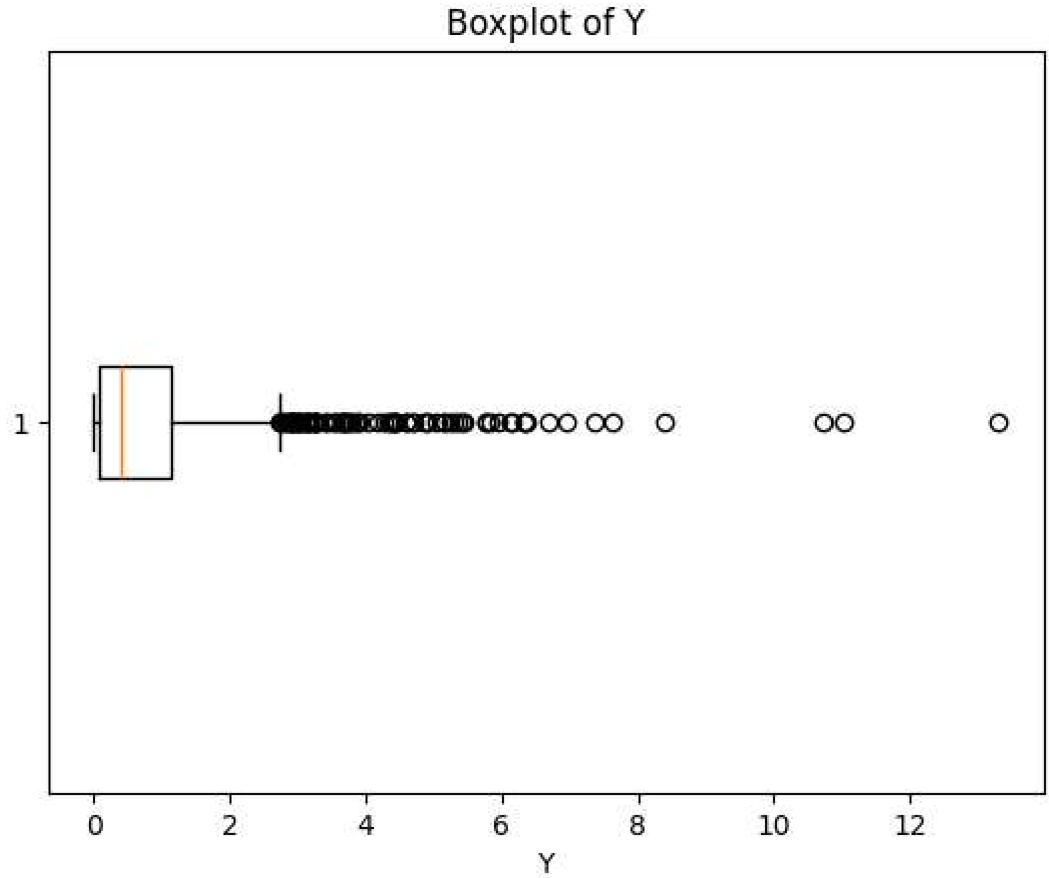
$Y = X^2$ 抽樣分配

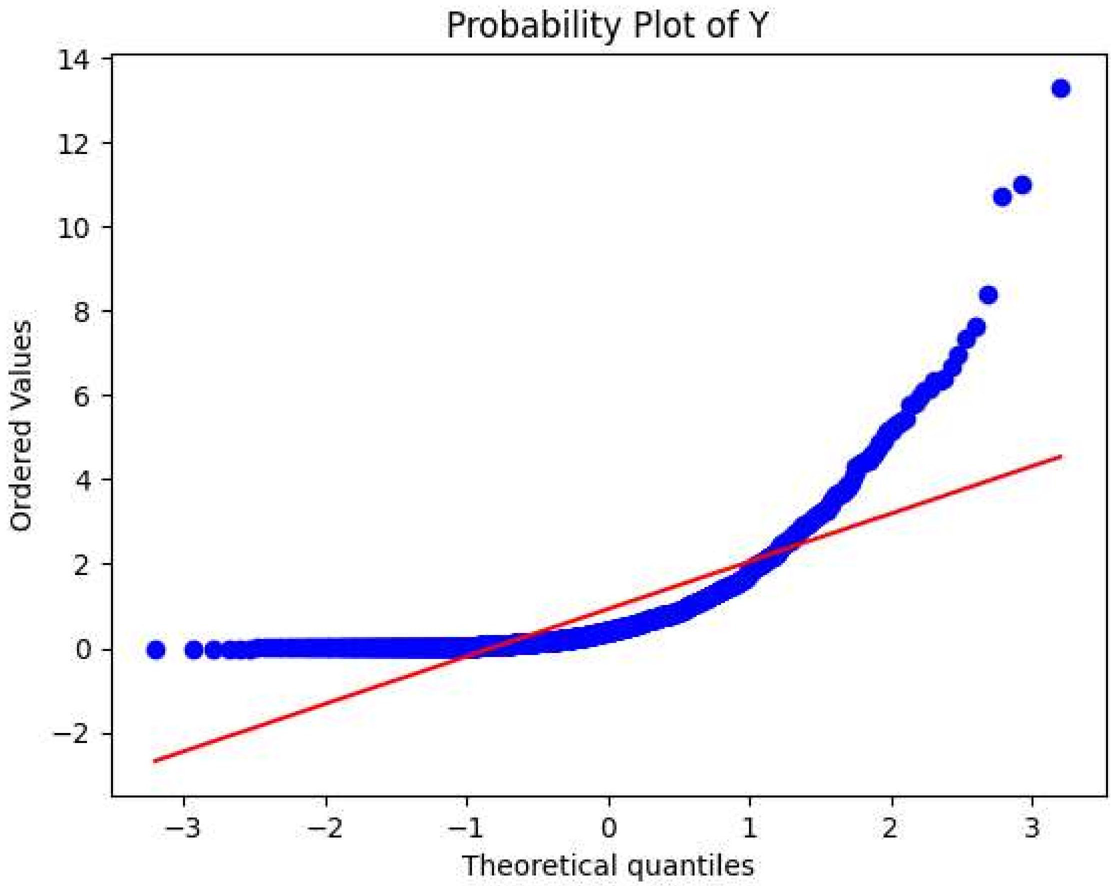
In [ ]:

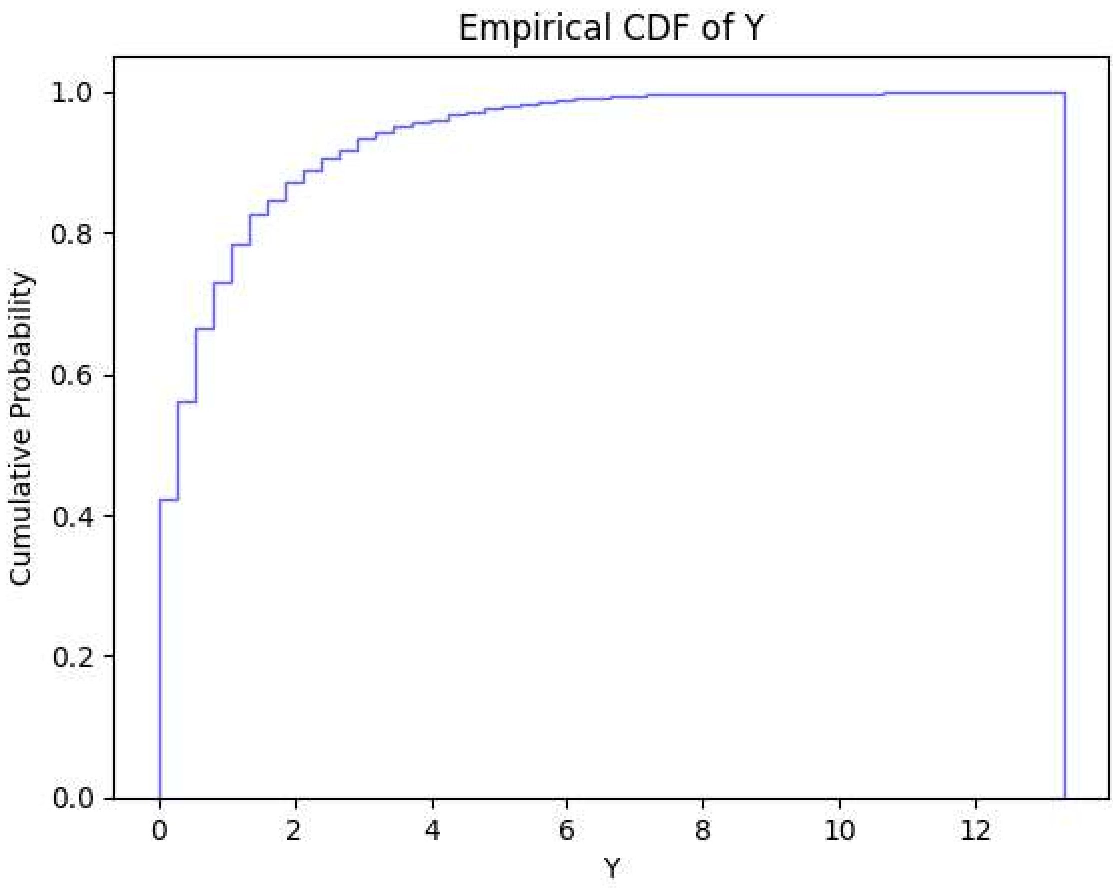
```
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as stats

sample_size = 1000
X = np.random.normal(0, 1, sample_size)

Y = X**2

plt.hist(Y, bins=20, density=True, alpha=0.6, color='g')
plt.title("Histogram of Y")
plt.xlabel("Y")
plt.ylabel("Probability")
plt.show()

# 繪製 Y 的盒鬚圖
plt.boxplot(Y, vert=False)
plt.title("Boxplot of Y")
```

```
plt.xlabel("Y")
plt.show()

# 繪製概率圖（probability plot）
plt.figure()
stats.probplot(Y, dist="norm", plot=plt)
plt.title("Probability Plot of Y")
plt.show()

# 繪製 Y 的經驗累積分佈函數（Empirical CDF）
plt.hist(Y, bins=50, density=True, cumulative=True, histtype='step', color='b', alp
plt.title("Empirical CDF of Y")
plt.xlabel("Y")
plt.ylabel("Cumulative Probability")
plt.show()
```




Boxplot of Y
1
0
2
4
6
8
10
12
Y


Probability Plot of Y
14
12
10
8
6
4
2
0
−2
Ordered Values
−3
−2
−1
0
1
2
3
Theoretical quantiles



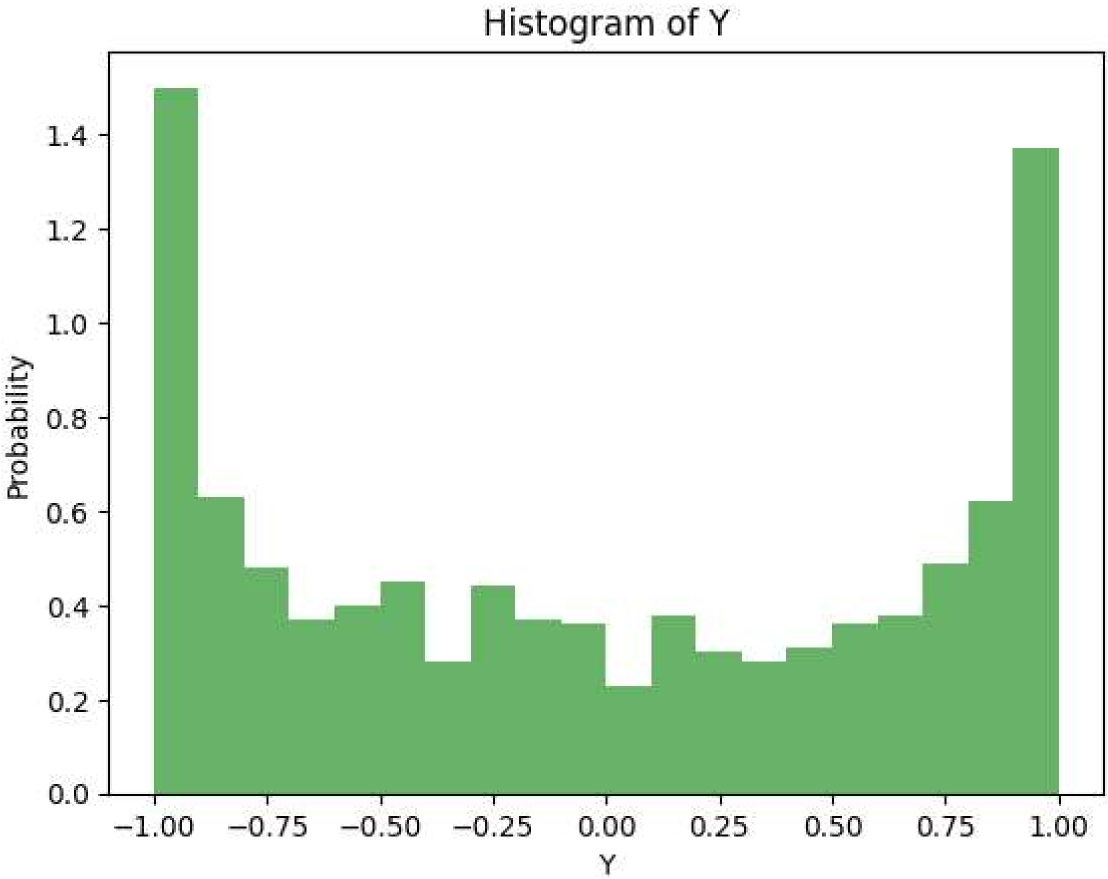
作品結論：

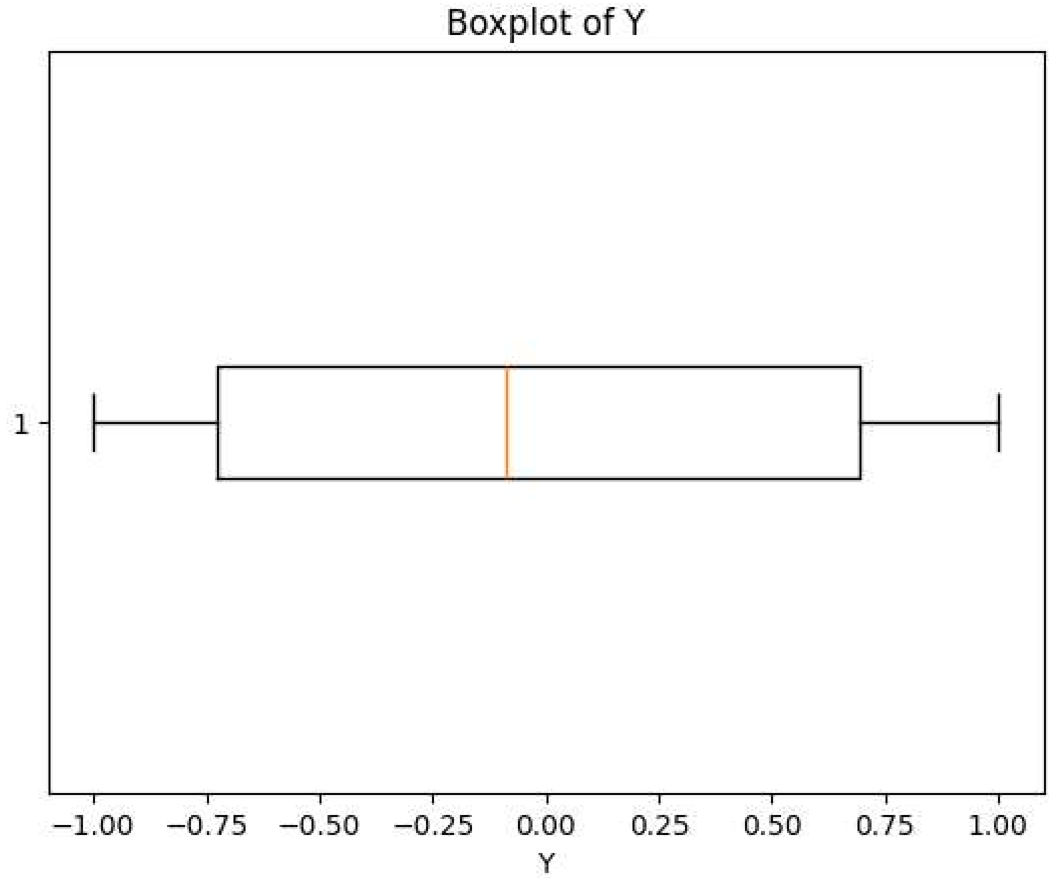
直方圖：直方圖顯示了 Y 的分布形狀。在這個例子中，Y 是 X 的平方，因此 Y 的分佈是偏斜的並呈現正偏斜（右偏）分佈。

概率圖（Probability Plot）：概率圖用於檢查數據是否符合某種特定的分佈。在這個例子中，我們使用常態分佈來檢查 Y 是否呈現線性趨勢。如果數據點基本上在一條直線上，則表明 Y 大致符合常態分佈，以本例子來說，並不符合常態分布。

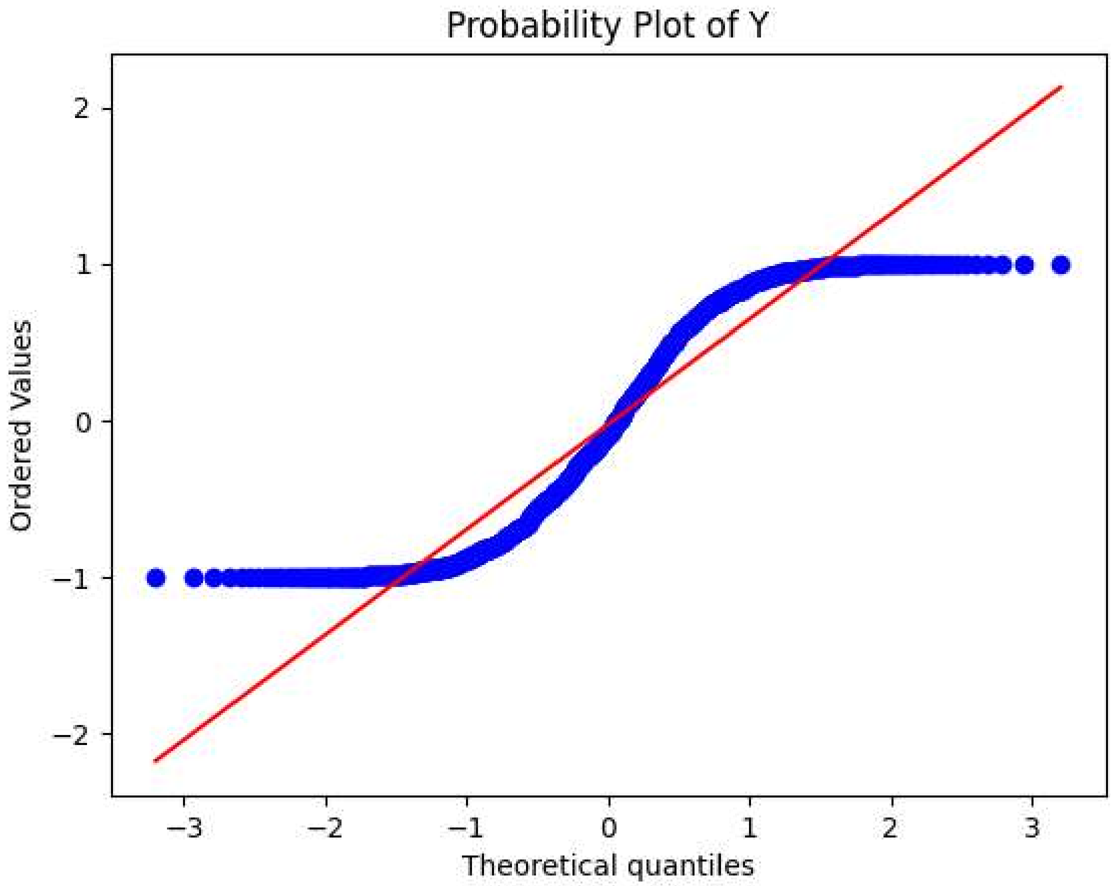
In [ ]:
```
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as stats

# 生成隨機樣本 X
sample_size = 1000
X = np.random.uniform(0, 2 * np.pi, sample_size)  # 這裡以均勻分佈為例，可以更改分佈

# 計算 Y = sin(X)
Y = np.sin(X)

# 繪製 Y 的直方圖
plt.hist(Y, bins=20, density=True, alpha=0.6, color='g')
plt.title("Histogram of Y")
plt.xlabel("Y")
plt.ylabel("Probability")
plt.show()

# 繪製 Y 的盒鬚圖
plt.boxplot(Y, vert=False)
plt.title("Boxplot of Y")
plt.xlabel("Y")
plt.show()
```

```python
# 繪製概率圖（probability plot）
plt.figure()
stats.probplot(Y, dist="norm", plot=plt)
plt.title("Probability Plot of Y")
plt.show()

# 繪製 Y 的經驗累積分佈函數（Empirical CDF）
plt.hist(Y, bins=50, density=True, cumulative=True, histtype='step', color='b', alp
plt.title("Empirical CDF of Y")
plt.xlabel("Y")
plt.ylabel("Cumulative Probability")
plt.show()
```

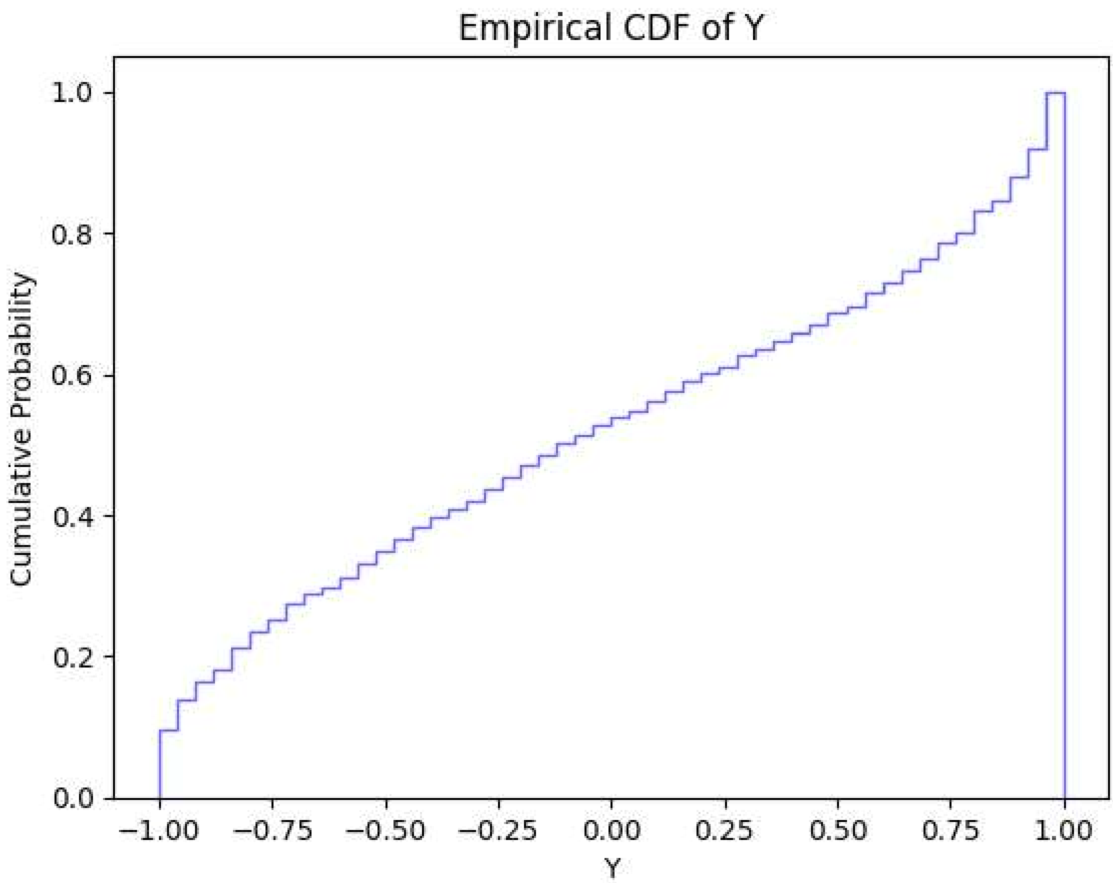




Boxplot of Y
1
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
Y



Probability Plot of Y
Ordered Values
2
1
0
−1
−2
−3
−2
−1
0
1
2
3
Theoretical quantiles



作品結論：

由於 y = sin(x) 的值介於 -1 到 1 之間，直方圖的高度應該在這個範圍內。

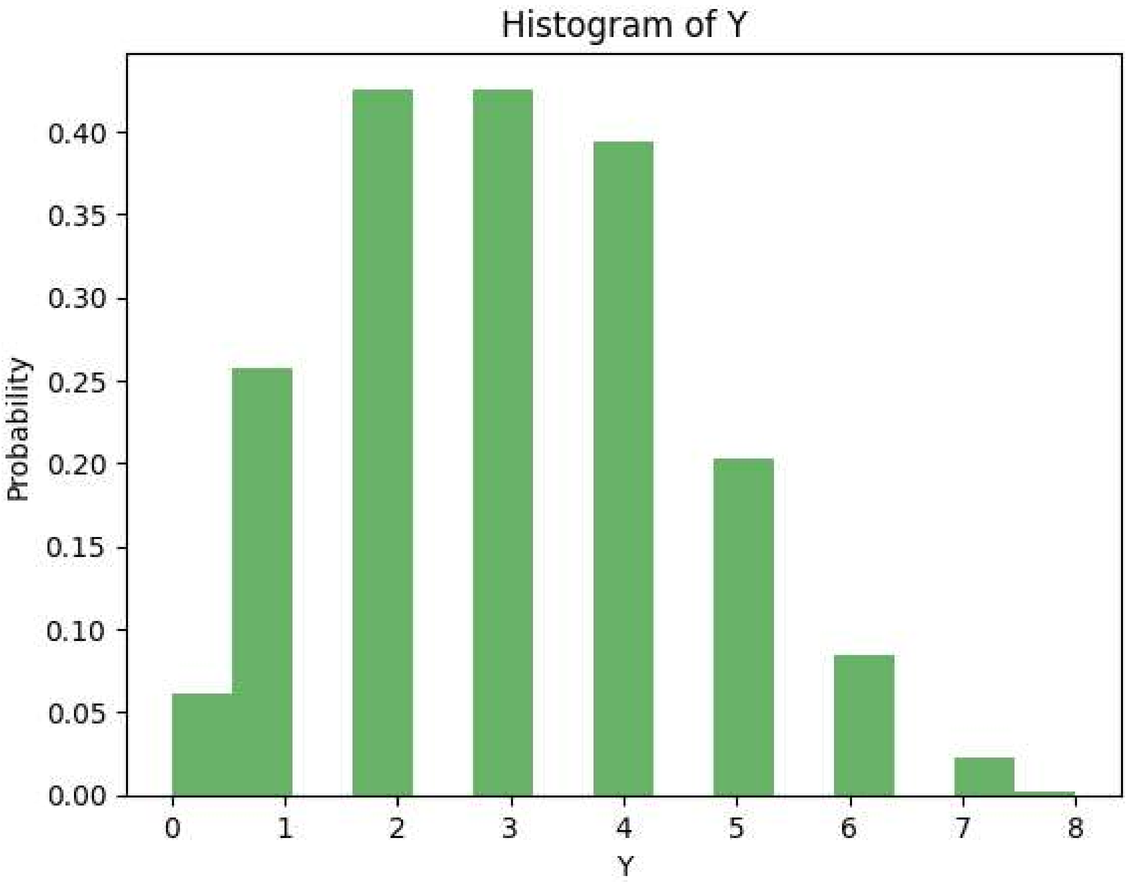
由於 y = sin(x) 的振幅為 1，盒鬚圖的箱體高度應在 -1 到 1 之間。

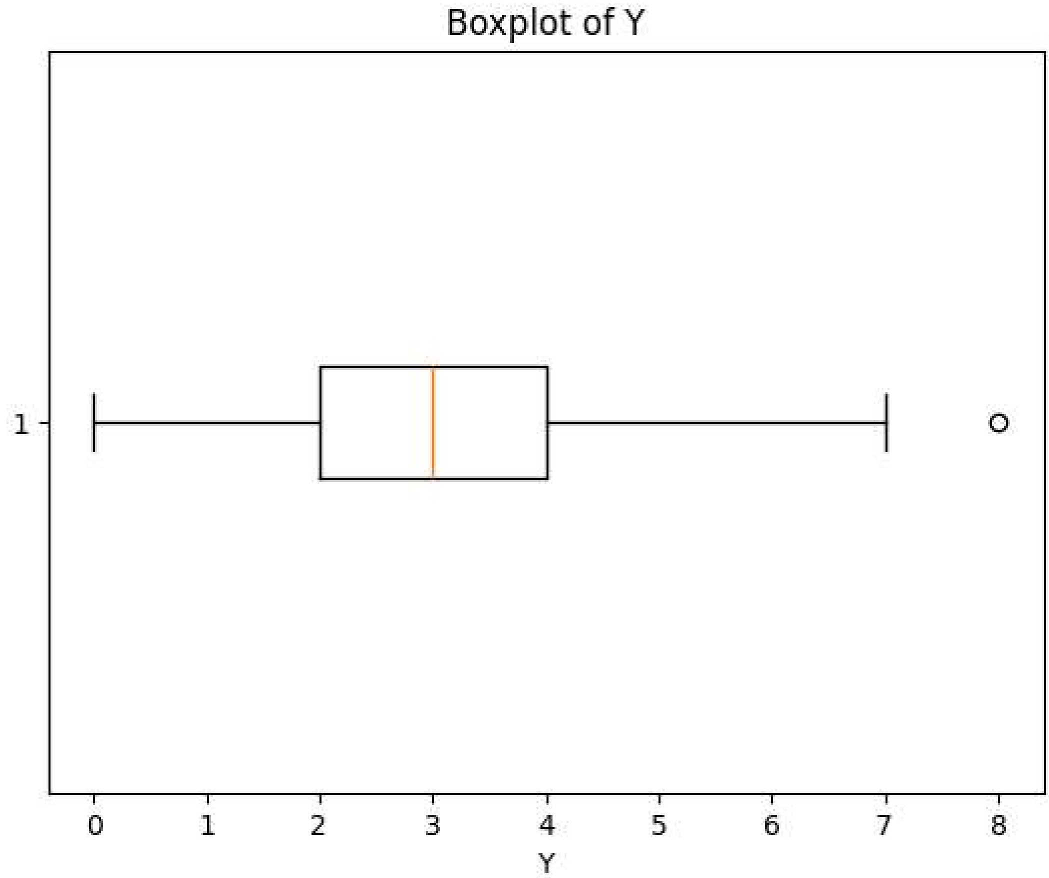
經驗累積分布函數將呈現 y = sin(x) 的值在特定範圍內的分佈情況，並且應該呈現週期性。

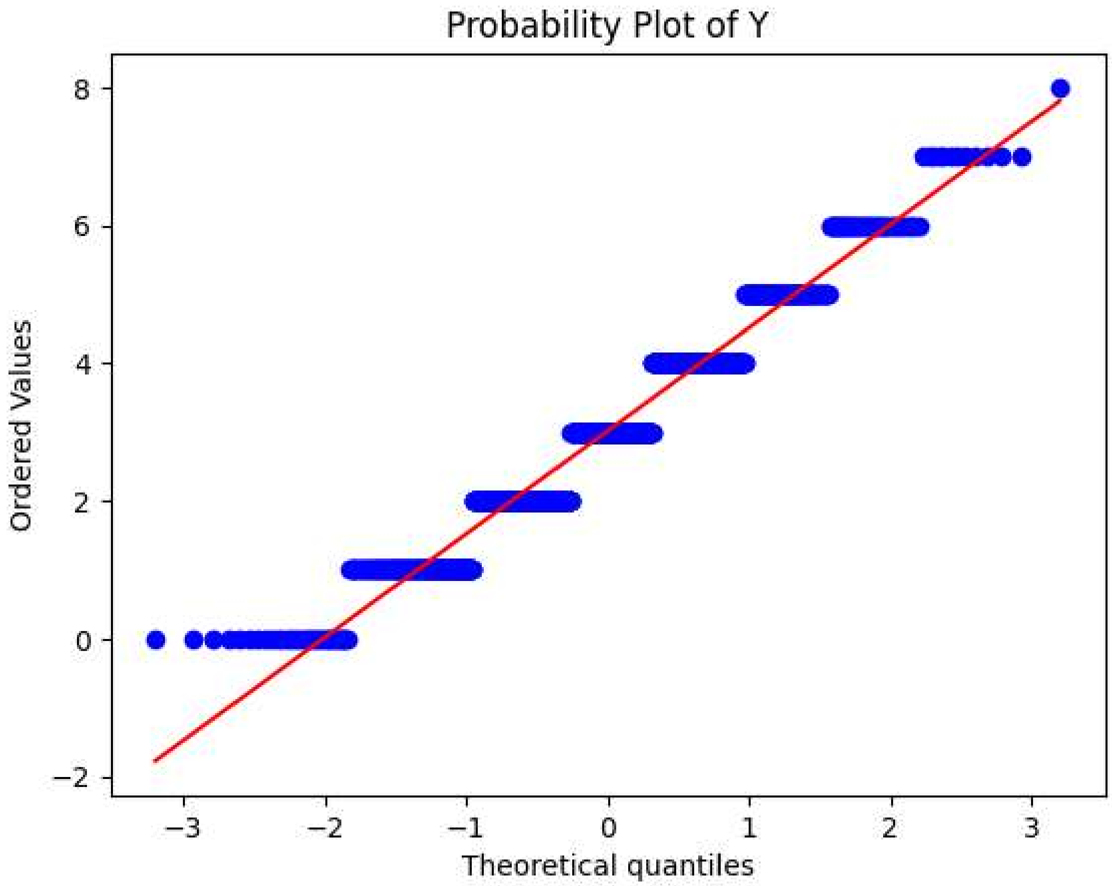
In [ ]:

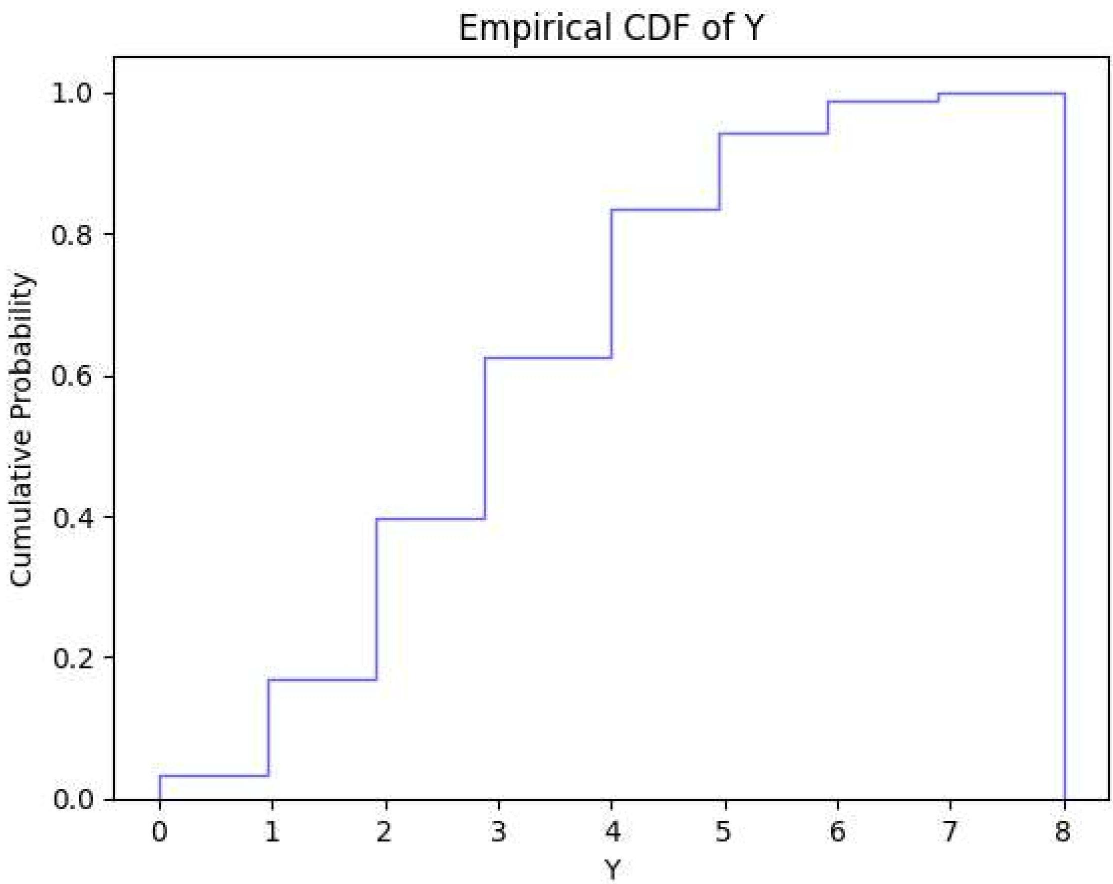
```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import binom

# 生成二項分配的隨機樣本 X
n = 10  # 試驗次數
p = 0.3  # 成功概率
sample_size = 1000
Y = np.random.binomial(n, p, sample_size)

# 繪製 Y 的直方圖
plt.hist(Y, bins=15, density=True, alpha=0.6, color='g')
plt.title("Histogram of Y")
plt.xlabel("Y")
plt.ylabel("Probability")
plt.show()

# 繪製 Y 的盒鬚圖
plt.boxplot(Y, vert=False)
plt.title("Boxplot of Y")
plt.xlabel("Y")
plt.show()

# 繪製概率圖 (probability plot)
```

```python
plt.figure()
stats.probplot(Y, dist="norm", plot=plt)
plt.title("Probability Plot of Y")
plt.show()

# 繪製 Y 的經驗累積分佈函數 (Empirical CDF )
plt.hist(Y, bins=50, density=True, cumulative=True, histtype='step', color='b', alp
plt.title("Empirical CDF of Y")
plt.xlabel("Y")
plt.ylabel("Cumulative Probability")
plt.show()
```




Boxplot of Y
1
0
1
2
3
4
5
6
7
8
Y


Probability Plot of Y
8
6
4
2
0
−2
Ordered Values
−3
−2
−1
0
1
2
3
Theoretical quantiles



作品結論：

二項分配的直方圖通常呈現離散的分佈，因為它描述了二項試驗中成功的次數。

盒鬚圖通常呈現右偏，因為在大多數情況下，成功次數不會太高。

概率圖通常不適用於二項分配，因為它描述的是離散分佈，而概率圖通常用於連續分佈。

在二項分配的情境下，經驗累積分布函數應該呈現隨著成功次數的增加而累積的分布情況。