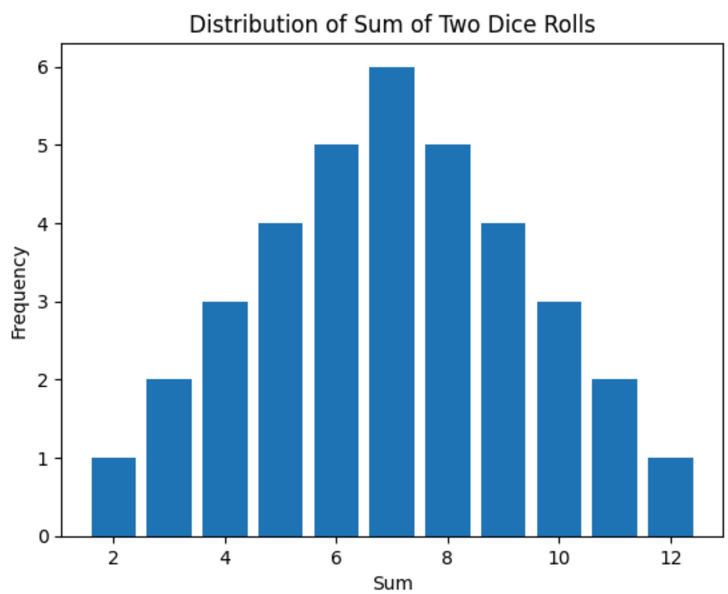
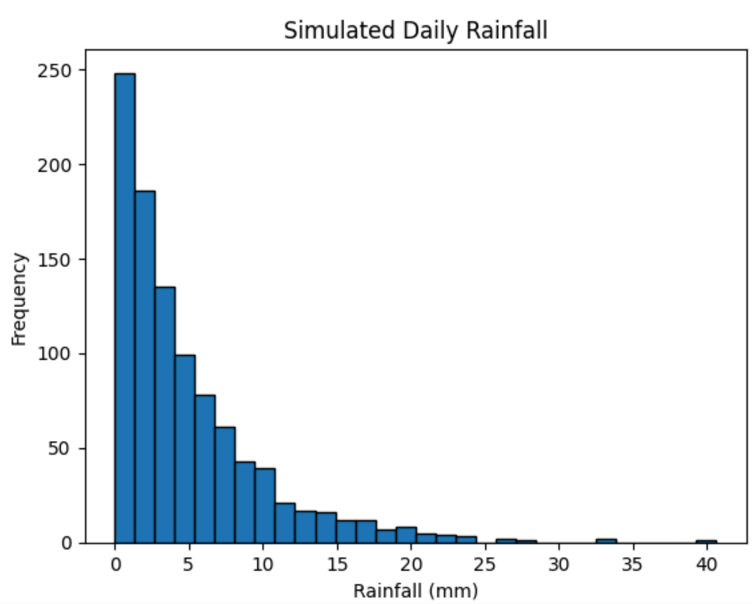
```
import numpy as np
import matplotlib.pyplot as plt
from collections import Counter

sums = [i + j for i in range(1, 7) for j in range(1, 7)]
counts = Counter(sums)

plt.bar(counts.keys(), counts.values())
plt.title('Distribution of Sum of Two Dice Rolls')
plt.xlabel('Sum')
plt.ylabel('Frequency')
plt.show()
```

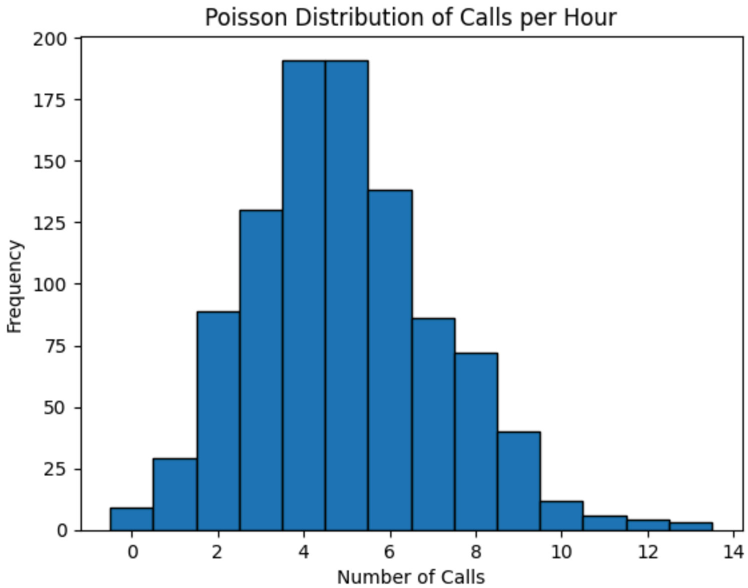



```
rainfall = np.random.exponential(scale=5.0, size=1000)

plt.hist(rainfall, bins=30, edgecolor='black')
plt.title('Simulated Daily Rainfall')
plt.xlabel('Rainfall (mm)')
plt.ylabel('Frequency')
plt.show()
```

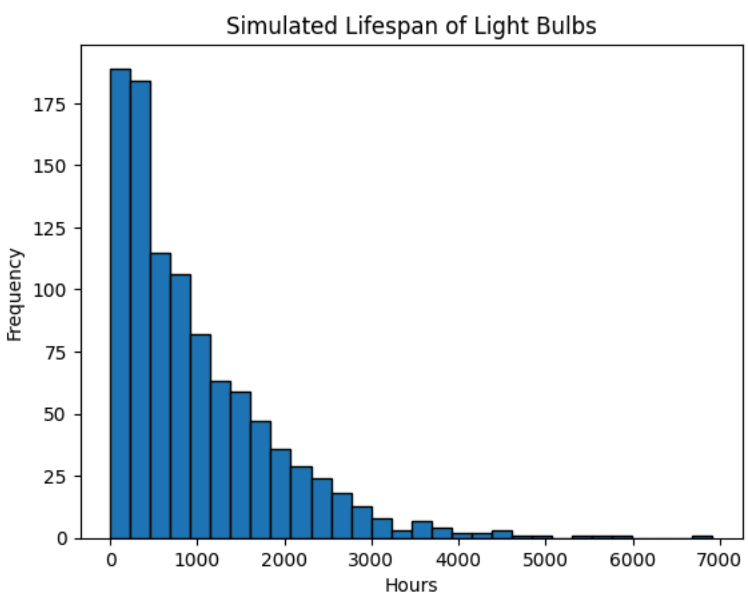



```
from scipy.stats import poisson
```

```
calls = poisson.rvs(mu=5, size=1000)
plt.hist(calls, bins=range(0, 15), align='left', edgecolor='black')
plt.title('Poisson Distribution of Calls per Hour')
plt.xlabel('Number of Calls')
plt.ylabel('Frequency')
plt.show()
```



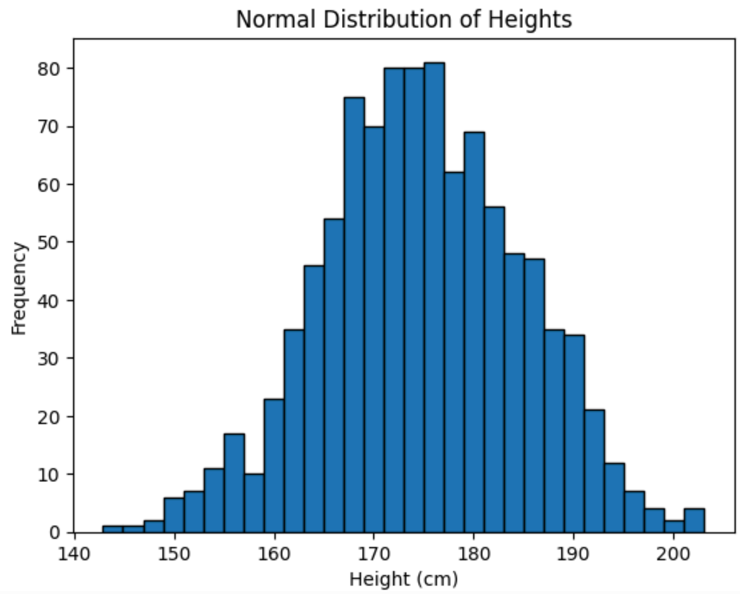

```
lifespans = np.random.exponential(scale=1000, size=1000)

plt.hist(lifespans, bins=30, edgecolor='black')
plt.title('Simulated Lifespan of Light Bulbs')
plt.xlabel('Hours')
plt.ylabel('Frequency')
plt.show()
```



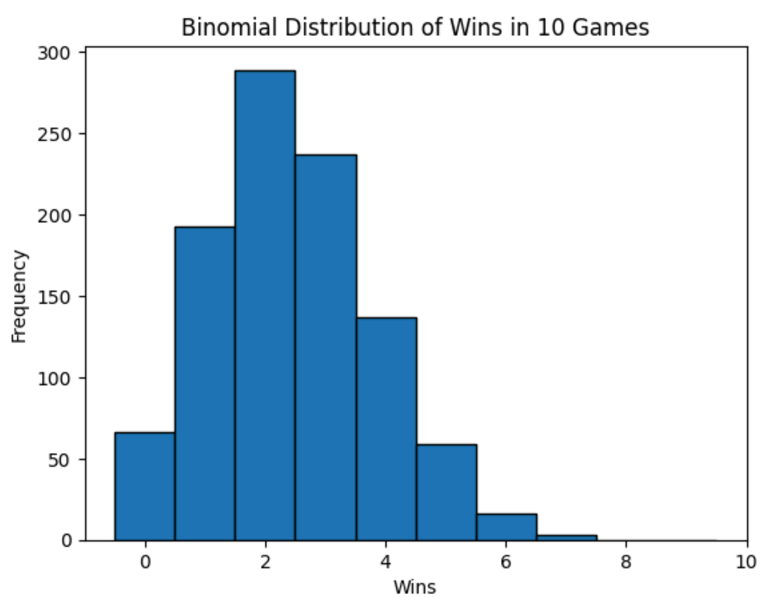

```
prior = 0.7
posterior = 0.9

print(f"Prior belief: {prior}")
print(f"Updated belief after studying (Bayesian): {posterior}")
```

```
Prior belief: 0.7
Updated belief after studying (Bayesian): 0.9
```

```
heights = np.random.normal(loc=175, scale=10, size=1000)

plt.hist(heights, bins=30, edgecolor='black')
plt.title('Normal Distribution of Heights')
plt.xlabel('Height (cm)')
plt.ylabel('Frequency')
plt.show()
```

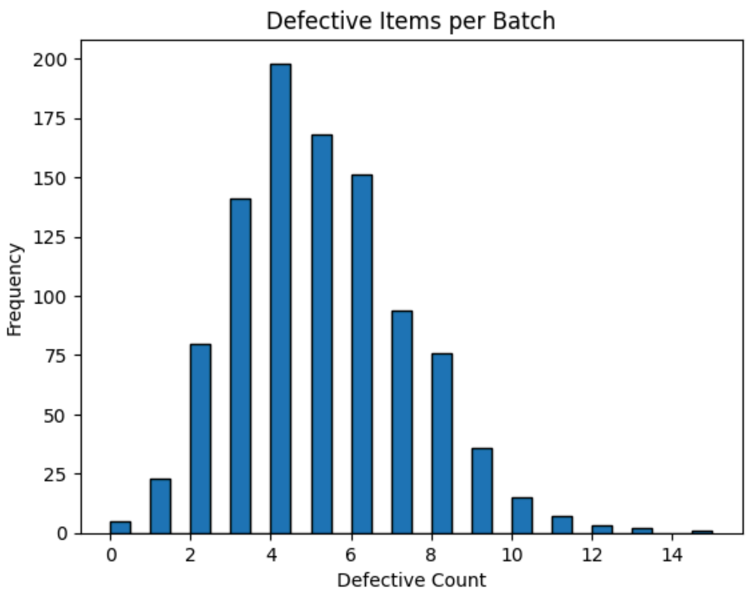



```
wins = np.random.binomial(n=10, p=0.25, size=1000)

plt.hist(wins, bins=range(0, 11), align='left', edgecolor='black')
plt.title('Binomial Distribution of Wins in 10 Games')
plt.xlabel('Wins')
plt.ylabel('Frequency')
plt.show()
```

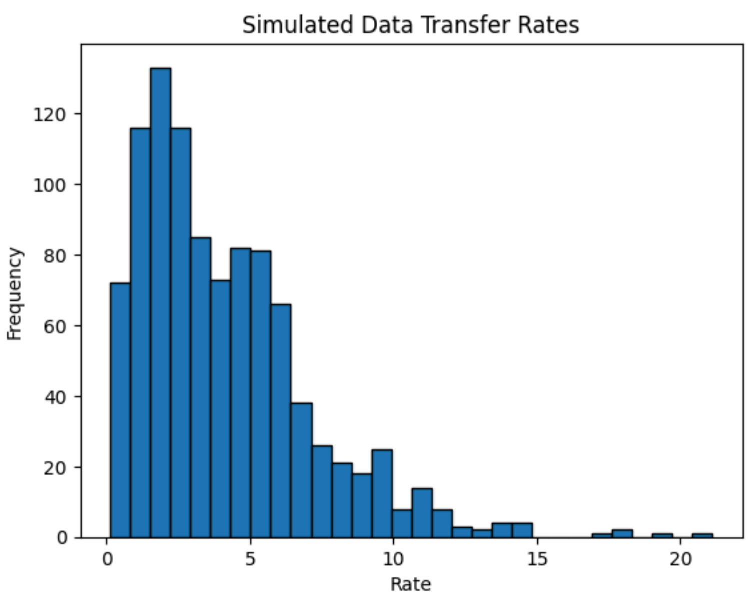



```
defectives = np.random.binomial(n=100, p=0.05, size=1000)

plt.hist(defectives, bins=30, edgecolor='black')
plt.title('Defective Items per Batch')
plt.xlabel('Defective Count')
plt.ylabel('Frequency')
plt.show()
```

```
plt.show()
```




```
data_rates = np.random.gamma(shape=2.0, scale=2.0, size=1000)

plt.hist(data_rates, bins=30, edgecolor='black')
plt.title('Simulated Data Transfer Rates')
plt.xlabel('Rate')
plt.ylabel('Frequency')
plt.show()
```

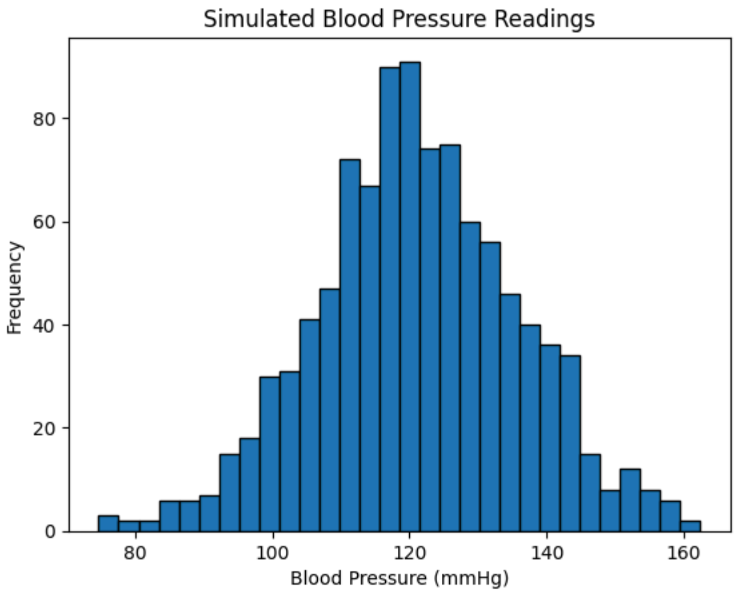



```
prior = 0.3
posterior = 0.6

print(f"Prior probability of rain: {prior}")
print(f"Posterior probability after forecast: {posterior}")
```

```
Prior probability of rain: 0.3
Posterior probability after forecast: 0.6
```

```
bp_readings = np.random.normal(loc=120, scale=15, size=1000)

plt.hist(bp_readings, bins=30, edgecolor='black')
plt.title('Simulated Blood Pressure Readings')
```

```
plt.xlabel('Blood Pressure (mmHg)')
plt.ylabel('Frequency')
plt.show()
```




```
clicks = poisson.rvs(mu=3, size=1000)

plt.hist(clicks, bins=range(0, 15), align='left', edgecolor='black')
plt.title('Poisson Distribution of Ad Clicks')
plt.xlabel('Clicks')
plt.ylabel('Frequency')
plt.show()
```