Assignment 1

Yuting Liang

1.Data Processin

First, understand why there are $n+2$ observations.

If we have closing prices for $n+2$ days, we get $n+1$ comparisons between consecutive days, meaning there can be at most n consecutive up moves (for example, if the price increases for three consecutive days, we count it as 2 up moves).

After clarifying the definition of consecutive up moves, we use Python to read the closing prices, calculate the difference between consecutive days, and count how many differences are greater than 0. The results are:

- Total days (M): 1257
- n (M - 2): 1255
- X (number of up moves in the first n intervals): 731

2. Bayesian Estimation: Posterior Distribution and Mean

(a) Posterior Distribution with a Uniform Prior

Because $\pi(\theta) = 1$, the posterior distribution becomes

$$P(\theta \mid X) = (n+1)\,\theta^{x}(1-\theta)^{n-x},$$

which is exactly a Beta distribution:

$$\theta \mid X \sim \text{Beta}(x+1,\ n-x+1).$$

Hence, the posterior mean is

$$E[\theta \mid X] = \frac{x+1}{n+2}.$$

(b) Posterior Distribution with a Beta Prior

If the prior is $\text{Beta}(\alpha, \beta)$, the posterior is still

$$\theta \mid X \sim \text{Beta}(\alpha + x,\ \beta + n - x).$$

So, the posterior mean is

$$E[\theta \mid X] = \frac{\alpha + x}{\alpha + \beta + n}.$$

Substituting the given parameters (like 1.6 and 80.0) gives:

$$E[\theta \mid X] = \frac{1.6 + x}{1.6 + 80.0 + n}.$$

3. MLE Estimation

MLE (Maximum Likelihood Estimation) is found by maximizing the likelihood function:

$$\hat{\theta}_{ML} = \arg\max P(X \mid \theta).$$

Taking the derivative and setting it to zero gives:

$$\hat{\theta}_{ML} = \frac{x}{n}.$$

4. Results Comparison

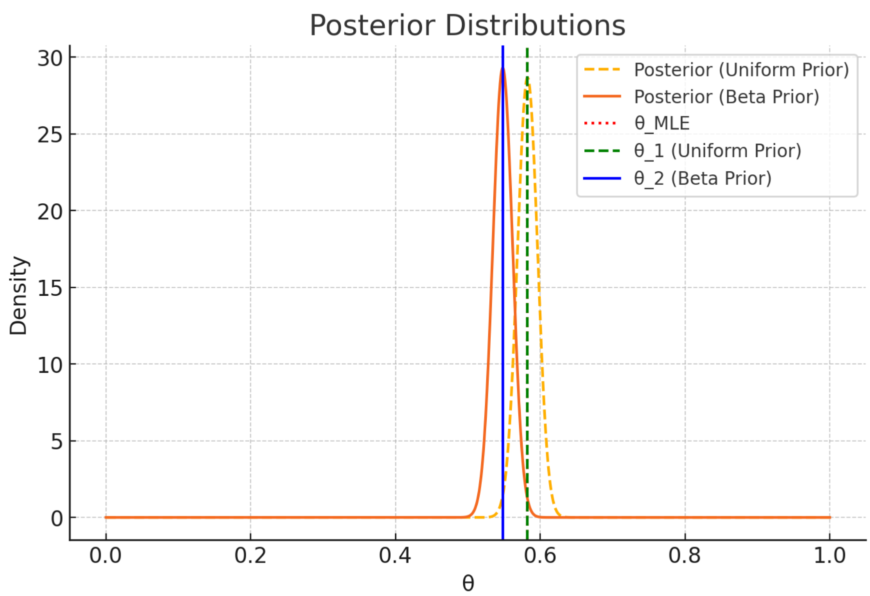
By plugging x and n into the formulas, we get:

$$\theta_1 \text{ (Uniform prior)} : 0.5823, \quad \theta_2 \text{ (Beta prior)} : 0.5481$$

The 95% credible interval from the Beta posterior is (0.5214, 0.5747).

$$\theta_{\text{MLE}} = 0.5825$$

Visualization:



In short:

1. From an estimation perspective, MLE only uses the data, while Bayesian estimates also include prior beliefs.

2. MLE gives just a single point estimate, whereas Bayesian methods provide a distribution that shows a likely range for the ratio.

3. In this assignment, there's a small difference in the estimates, but in the model1 example, they're almost the same. This is because our assignment data is limited, so the prior heavily influences the posterior. In the classroom example, there was a lot more data, so the data had a bigger effect on the posterior.

5. Python Code

```python
import pandas as pd

df = pd.read_csv(r"C:\Users\ASUS\Downloads\AMS522_Homework_Data.csv")

df = df.sort_values('Date').reset_index(drop=True)

prices = df['Close Price'].values

changes = prices[1:] - prices[:-1]

M = len(prices)   # Total days of data
n = M - 2
X = sum(changes[:n] > 0)

print("Total days (M):", M)
print("n (M - 2):", n)
print("X (number of up moves in first n intervals):", X)




import scipy.stats as stats

theta_MLE = X / n

theta_1 = (X + 1) / (n + 2)

alpha_prior = 1.6
beta_prior = 80.0
theta_2 = (alpha_prior + X) / (alpha_prior + beta_prior + n)

# Compute 95% credible interval for Beta posterior
posterior_beta_dist = stats.beta(alpha_prior + X, beta_prior + n - X)
theta_posterior_ci = posterior_beta_dist.interval(0.95)

# Print results
print(f" _MLE: {theta_MLE:.4f}")
```

```python
print(f" _1 (Uniform prior): {theta_1:.4f}")
print(f" _2 (Beta prior): {theta_2:.4f}")
print(f"95% CI (Beta posterior): {theta_posterior_ci}")

# Return results
theta_MLE, theta_1, theta_2, theta_posterior_ci




import matplotlib.pyplot as plt
import numpy as np

# Generate theta values for plotting
theta_values = np.linspace(0, 1, 1000)

# Compute posterior distributions
posterior_uniform = stats.beta(X + 1, n - X + 1).pdf(theta_values)
posterior_beta = stats.beta(alpha_prior + X, beta_prior + n - X).pdf(
    theta_values)

# Plot posterior distributions
plt.figure(figsize=(8, 5))
plt.plot(theta_values, posterior_uniform, label="Posterior (Uniform Prior)"
    , linestyle='dashed')
plt.plot(theta_values, posterior_beta, label="Posterior (Beta Prior)",
    linestyle='solid')
plt.axvline(theta_MLE, color='r', linestyle=':', label=" _MLE")
plt.axvline(theta_1, color='g', linestyle='--', label=" _1 (Uniform Prior)"
    )
plt.axvline(theta_2, color='b', linestyle='-', label=" _2 (Beta Prior)")

# Labels and legend
plt.xlabel(" ")
plt.ylabel("Density")
plt.title("Posterior Distributions")
plt.legend()
plt.grid(True)

# Show plot
plt.show()
```