Q6

October 1, 2024

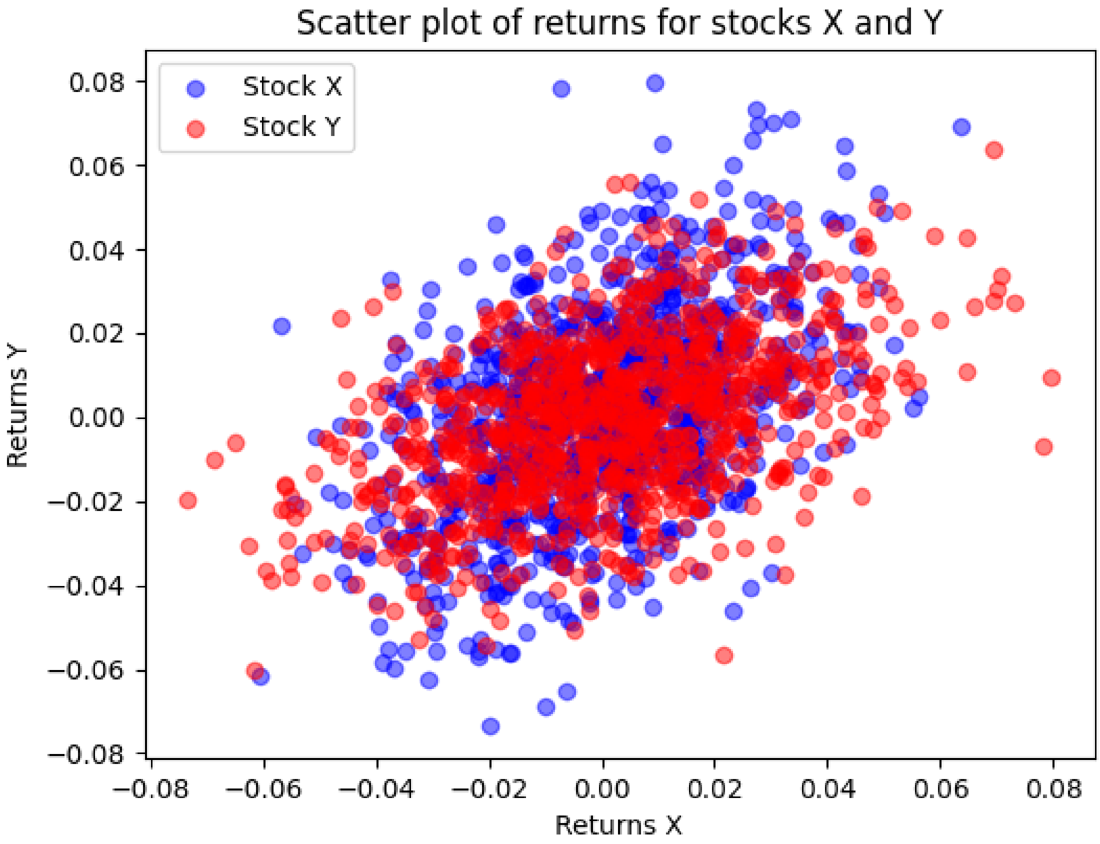
2. The daily returns of two correlated stocks, X and Y , follow a joint lognormal distribution with the following parameters: X = 0.001, Y = 0.002, X = 0.02, Y = 0.03, X,Y = 0.8

(c) (4 points) Simulate 1,000 days of returns for both stocks using the joint lognormal distribution. Plot the scatter plot and calculate the empirical correlation.

[3]:

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

mu_X = 0.001  # Expected return of stock X
sigma_X = 0.02  # Standard deviation of return of stock X
mu_Y = 0.0015  # Expected return of stock Y
sigma_Y = 0.025  # Standard deviation of return of stock Y
rho_XY = 0.5  # Correlation coefficient between returns of stock X and Y
num_days = 1000  # Number of days to simulate returns for both stocks

# Simulate 1,000 days of returns for both stocks using the joint lognormal␣
 ↪distribution
np.random.seed(42)  # For reproducibility
mean = [mu_X, mu_Y]
cov_matrix = [[sigma_X**2, rho_XY * sigma_X * sigma_Y], [rho_XY * sigma_X *␣
 ↪sigma_Y, sigma_Y**2]] # Covariance matrix of returns of stock X and Y
returns = np.random.multivariate_normal(mean, cov_matrix, num_days) # Simulate␣
 ↪returns for both stocks using the joint lognormal distribution

returns_X = returns[:, 0] # Returns of stock X
returns_Y = returns[:, 1] # Returns of stock Y


# Plot the scatter plot with different colors for each stock
plt.scatter(returns_X, returns_Y, alpha=0.5, c='blue', label='Stock X')
plt.scatter(returns_Y, returns_X, alpha=0.5, c='red', label='Stock Y')
plt.xlabel('Returns X')
plt.ylabel('Returns Y')
plt.title('Scatter plot of returns for stocks X and Y')
plt.legend()
plt.show()
```

```
print("The empirical correlation coefficient between the returns of stock X and␣
↪Y is:" , np.corrcoef(returns_X, returns_Y)[0, 1])
```



```
The empirical correlation coefficient between the returns of stock X and Y is:
0.462009489994836
```

(d) (2 points) Using the simulated data, calculate the percentage of days where both stocks have positive returns. Compare this to the theoretical correlation.

[4]:

```
# Calculate the percentage of days where both stocks have positive returns
positive_returns_both = np.sum((returns_X > 0) & (returns_Y > 0))
percentage_positive_returns_both = (positive_returns_both / num_days) * 100

print(f"The percentage of days where both stocks have positive returns is:␣
↪{percentage_positive_returns_both:.2f}%")
```

```
The percentage of days where both stocks have positive returns is: 33.70%
```