Financial Econometrics Assignment 2

Halis Sak

Due date: October 25, 2024

Question 1. Please see "data.csv" for monthly adjusted closing prices of Hang Seng Bank (0011.HK) and AIA (1299.HK).

a) Compute monthly expected return and standard deviation of both assets. Hint. First we need to compute monthly returns as follows

$$R_{t+1} = \frac{S_{t+1} - S_t}{S_t},$$

where S_t is the adjusted closing stock price for month t.

Answer.

Bank	Hang Seng Bank	AIA
Expected Return	0.007559	0.019073
Standard Deviation	0.062924	0.067147

Process:

Compute the monthly return from 2016.2 to 2020.12.

The fomula will be

$$R_{2016.3} = \frac{R_{2016.4.1} - R_{2016.3.1}}{R_{2016.3.1}}$$

$$R_{2016.4} = \frac{R_{2016.5.1} - R_{2016.4.1}}{R_{2016.4.1}}$$

$$R_{2016.5} = \frac{R_{2016.6.1} - R_{2016.5.1}}{R_{2016.5.1}}$$

……

$$R_{2020.12} = \frac{R_{2021.1.1} - R_{2020.12.1}}{R_{2020.12.1}}$$

So we use this following calculation to create a new colomn in the grapgh and caculate the monthly returns for HSB and AIA by using the `shift(1)` function to shift the stock price series by one month. This shift is necessary to calculate the price difference between consecutive months. The formula `(data['HSB'] - data['HSB'].shift(1)) / data['HSB'].shift(1)` is then applied to divide the difference in stock prices by the previous month's price, giving the monthly return. And the results are stored in new columns, 'Return_HSB' and 'Return_AIA'.

```
data['Return_HSB'] = (data['HSB'] - data['HSB'].shift(1)) / data['HSB'].shift(1)
data['Return_AIA'] = (data['AIA'] - data['AIA'].shift(1)) / data['AIA'].shift(1)
```

To calculate the expected returns for HSB and AIA, the .mean() function is used to compute the average of each stock's return column, representing the average return over the entire period.

```
expected_return_hsb = data['Return_HSB'].mean()
expected_return_aia = data['Return_AIA'].mean()
```

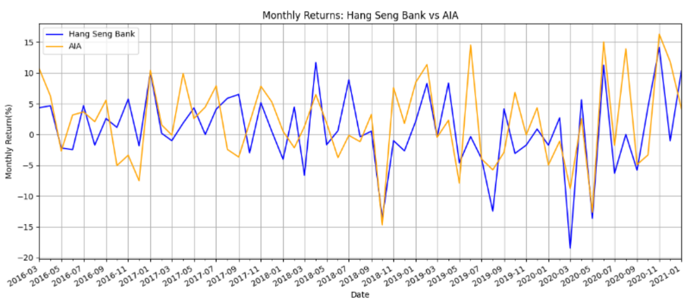
To calculate the standard deviation of the returns for HSB and AIA, the .std() function is applied to each stock's return column.

```
std_dev_hsb = data['Return_HSB'].std()
std_dev_aia = data['Return_AIA'].std()
```

b) Please plot the monthly returns of Hang Seng Bank versus monthly returns of AIA. And compute the correlation between Hang Seng Bank and AIA using the data provided. Does the correlation that you computed reflect (correlation-wise) the graph that you draw?

Answer.

Correlation between Hang Seng Bank and AIA: 0.5883(keep the decimal place to four)



The correlation reflect the graphs.

Process:

We use the .corr() method computes the correlation between Return_HSB and Return_AIA. This value measures how closely the returns of the two stocks move together. A positive value indicates they tend to move in the same direction, while a negative value indicates they move in opposite directions.

```
correlation = data['Return_HSB'].corr(data['Return_AIA'])
print("Correlation between Hang Seng Bank and AIA:", correlation)
```

We use plt.figure() to create a new graph object (figure). Use figsize, label, color, plt.title(),plt.xlabel(), plt.ylabel(), plt.grid() and so on to set the parameter of the gragh. Use plt.plot() to plot the monthly returns of Hang Seng Bank versus monthly returns of AIA .

Finally, We compute correlation between HSB and AIA return. Use the .corr() method computes the correlation between Return_HSB and Return_AIA.

```
correlation = data['Return_HSB'].corr(data['Return_AIA'])
print("Correlation between Hang Seng Bank and AIA:", correlation)
```

c) We want to create a portfolio from these two stocks with 65% and 35% weights for Hang Seng Bank and AIA. Please compute variance of the portfolio return.

Answer.

Variance of the portfolio return: 0.0034

Standard Deviation of the portfolio return: 0.0579

(keep the decimal place to four)

Process:

To create a portfolio from these two stocks with 65% and 35% weights for Hang Seng Bank and AIA, we can use the weights = np.array() to define them. The weights are stored as a numpy array.

```
weights = np.array([0.65, 0.35])
```

The variance of portfolio is calculated by this formula:

$$\sigma^2_{\{portfolio\}} = w_1^2 \cdot \sigma_1^2 + w_2^2 \cdot \sigma_2^2 + 2 \cdot w_1 \cdot w_2 \cdot \sigma_1 \cdot \sigma_2 \cdot \rho_{12}$$

w_1 and w_2 are the weights of Hang Seng Bank and AIA in the portfolio.

σ_1 and σ_2 are the standard deviation of yields for Hang Seng Bank and AIA.

ρ_{12} is the correlation coefficient between the yield of Hang Seng Bank and AIA.

We select the two columns % return HSB and % return AIA from the data DataFrame as returns to calculate standard deviation σ_1 and σ_2.

For σ_1 and σ_2, we can use std_devs = returns.std() to calculate.

For the ρ_{12}, we can ues the correlation = returns.corr().iloc[] to calculate.

So we can design this code to calculate the variance:

```
returns = data[['% return HSB', '% return AIA']]
std_devs = returns.std()
correlation = returns.corr().iloc[0, 1]
portfolio_variance = (weights[0]**2 * std_devs.iloc[0]**2 +
                      weights[1]**2 * std_devs.iloc[1]**2 +
                      2 * correlation * weights[0] * weights[1] * std_devs.iloc[0] *
std_devs.iloc[1])
```

Appendix

a) The entire code for Question a

```
import pandas as pd
data = pd.read_csv("data.csv")
# Print column names to confirm correctness
#print(data.columns)
# Calculate monthly returns
data['Return_HSB'] = (data['HSB'] - data['HSB'].shift(1)) / data['HSB'].shift(1)
data['Return_AIA'] = (data['AIA'] - data['AIA'].shift(1)) / data['AIA'].shift(1)
# Drop NaN values
data = data.dropna()
# Calculate expected returns
expected_return_hsb = data['Return_HSB'].mean()
expected_return_aia = data['Return_AIA'].mean()
# Calculate standard deviation
std_dev_hsb = data['Return_HSB'].std()
std_dev_aia = data['Return_AIA'].std()
# Create results DataFrame
results = pd.DataFrame({
    'Bank': ['Hang Seng Bank', 'AIA'],
    'Expected Return': [expected_return_hsb, expected_return_aia],
    'Standard Deviation': [std_dev_hsb, std_dev_aia]
})
results = results.set_index('Bank').T
# Print the transposed results table
print(results)
# Save the results to a file
#results.to_csv("expected_returns_and_std_dev_transposed.csv")
```

b) The entire code for Question b

```
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.dates as mdates
# Read data file
data = pd.read_csv("data.csv")
data['Date'] = pd.to_datetime(data['Date'], format='%d/%m/%Y', dayfirst=True)
# Calculate monthly returns
data['Return_HSB'] = (data['HSB'] - data['HSB'].shift(1)) / data['HSB'].shift(1) *100
data['Return_AIA'] = (data['AIA'] - data['AIA'].shift(1)) / data['AIA'].shift(1)  *100
data = data.dropna()
# Compute correlation
correlation = data['Return_HSB'].corr(data['Return_AIA'])
print("Correlation between Hang Seng Bank and AIA:", correlation)
# Plot monthly returns
plt.figure(figsize=(26, 6))
plt.plot(data['Date'], data['Return_HSB'], label='Hang Seng Bank', color='blue')
plt.plot(data['Date'], data['Return_AIA'], label='AIA', color='orange')
# Set title and labels
plt.title('Monthly Returns: Hang Seng Bank vs AIA')
```

```
plt.xlabel('Date')
plt.ylabel('Monthly Return(%)')
plt.legend()
plt.grid()
# Set x-axis date format and interval
plt.gca().xaxis.set_major_locator(mdates.MonthLocator(interval=1))  # Every month
plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%Y-%m'))  # Format the date
plt.gca().xaxis.set_major_locator(mdates.MonthLocator(interval=2))  # Every two months
plt.gca().xaxis.set_minor_locator(mdates.MonthLocator())  # Minor ticks for each month
plt.xlim(pd.Timestamp('2016-03-01'), pd.Timestamp('2021-1-1'))
# Rotate date labels for better readability
plt.gcf().autofmt_xdate()
plt.show()
# Interpret the correlation with respect to the graph
if correlation > 0:
    print("The correlation is positive, indicating that the monthly returns of Hang Seng Bank and AIA tend to move in the same direction.")
elif correlation < 0:
    print("The correlation is negative, indicating that the monthly returns of Hang Seng Bank and AIA tend to move in opposite directions.")
else:
    print("The correlation is zero, indicating no relationship between the monthly returns of Hang Seng Bank and AIA.")
```

c) The entire code for Question c

```
import pandas as pd
import numpy as np
data = pd.read_csv("data.csv")
data['Date'] = pd.to_datetime(data['Date'])
data.set_index('Date', inplace=True)
data['% return HSB'] = (data['HSB'] - data['HSB'].shift(1)) / data['HSB'].shift(1)
data['% return AIA'] = (data['AIA'] - data['AIA'].shift(1)) / data['AIA'].shift(1)
data = data.dropna()
# Portfolio of 65% HSB and 35% AIA
weights = np.array([0.65, 0.35])
# Calculate variance of the portfolio return
returns = data[['% return HSB', '% return AIA']]
std_devs = returns.std()
correlation = returns.corr().iloc[0, 1]
portfolio_variance = (weights[0]**2 * std_devs.iloc[0]**2 +
                      weights[1]**2 * std_devs.iloc[1]**2 +
                      2 * correlation * weights[0] * weights[1] * std_devs.iloc[0] * std_devs.iloc[1])
# #Calculate Standard Deviation of the portfolio return
portfolio_std_dev = np.sqrt(portfolio_variance)
print(f"Variance of the portfolio return: {portfolio_variance:.4f}")
print(f"Standard Deviation of the portfolio return: {portfolio_std_dev:.4f}")
```