

General Project Work

1. Data Collection:

Gathered historical data for BAC (Bank of America) and the SPY (S&P 500 Index) from January 1, 2021, to March 8, 2024. Log Return and Volatility Estimation:

(1) Log Return Calculation:

I computed the daily log returns for both BAC and the SPY independently using the formula:

$$\text{Log Return} = \ln\left(\frac{P_t}{P_{t-1}}\right)$$

where P_t represents the price at time t .

(2) GARCH (1,1) Volatility Estimation:

Using the obtained log returns, estimated the volatility using the GARCH (1,1) model. This involved fitting the model to the log return series and extracting the volatility forecasts.

2. Correlation Estimation:

I calculated the correlation between BAC and SPY and then used correlation coefficient to assess the degree of linear relationship between the two assets.

3. Portfolio Risk Assessment:

(1) 99% Value at Risk (VaR) Estimation:

I estimated the 99% VaR for both the assigned symbol and SPY using two methods: historical simulation (equal weight) and delta normal VaR. Historical simulation involved simulating returns based on past data to calculate the VaR. Delta normal VaR employed the assumption of a normal distribution for returns.

(2) Risk Adjusted Performance Comparison:

On March 11, 2024, I compared the risk-adjusted performance of three portfolios. With 1 million USD invested in the Bank of America, 1 million USD invested in the S&P Index, and 2 million USD total invested equally in both assets.

Coding Part

```
In [1]: # Import libraries
import pandas as pd
import numpy as np
import yfinance as yf
from arch import arch_model
from scipy.stats import norm
import warnings
warnings.filterwarnings('ignore')

In [2]: # Time period for the data
start_date = "2021-01-01"
final_date = "2024-03-08"

In [3]: # Creating a function to Record the daily data
def get_stock_data(ticker, start_date, end_date):
    df_data = pd.DataFrame()
    stock = yf.download(ticker, start=start_date, end=end_date)
    df_data["Adj Close"] = stock["Adj Close"]
    df_data["Returns"] = stock["Adj Close"].pct_change()
    df_data["Log Returns"] = np.log(df_data["Adj Close"] / df_data["Adj Close"].shift(1))
    return df_data

In [4]: # Getting the data
bac = get_stock_data("BAC", start_date, final_date)
bac = bac.dropna()
spy = get_stock_data("SPY", start_date, final_date)
spy = spy.dropna()

[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed

In [5]: # Returns of BAC and SPY
bac_returns = np.round((bac["Returns"].sum() * 100), 2)
spy_returns = np.round((spy["Returns"].sum() * 100), 2)

# Log Returns of BAC and SPY
bac_log_returns = np.round((bac["Log Returns"].sum() * 100), 2)
spy_log_returns = np.round((spy["Log Returns"].sum() * 100), 2)

# Print the returns and log returns
print(f"Average Returns of BAC over the observed period is: {bac_returns} %")
print(f"Average Returns of SPY over the observed period is: {spy_returns} %")

print(f"Average Log>Returns of BAC over the observed period is: {bac_log_returns}")
print(f"Average Log>Returns of SPY over the observed period is: {spy_log_returns}")

Average Returns of BAC over the observed period is: 37.52 %
Average Returns of SPY over the observed period is: 42.55 %
Average Log>Returns of BAC over the observed period is: 25.08
Average Log>Returns of SPY over the observed period is: 37.8
```

```
In [6]: # Specifying the GARCH(1,1) model
bac_garch_model = arch_model(bac['Log Returns'][1:], mean='Zero', vol='Garch', p=1, q=1)
spy_garch_model = arch_model(spy['Log Returns'][1:], mean='Zero', vol='Garch', p=1, q=1)

# Fit the model
bac_garch_results = bac_garch_model.fit(dispatch='off')
spy_garch_results = spy_garch_model.fit(dispatch='off')

# Print model summary
print(bac_garch_results.summary())
print(spy_garch_results.summary())

# Extract volatility estimates
bac_volatility = bac_garch_results.conditional_volatility
spy_volatility = spy_garch_results.conditional_volatility

# Print the volatility estimate
print("Estimated volatility of BAC over the observed period is:", np.round(bac_volatility.mean(), 6))
print("Estimated volatility of SPY over the observed period is:", np.round(spy_volatility.mean(), 6))
```

Zero Mean - GARCH Model Results					
=====					
Dep. Variable:	Log Returns	R-squared:	0.000		
Mean Model:	Zero Mean	Adj. R-squared:	0.001		
Vol Model:	GARCH	Log-Likelihood:	2101.59		
Distribution:	Normal	AIC:	-4197.18		
Method:	Maximum Likelihood	BIC:	-4183.13		
		No. Observations:	797		
Date:	Fri, Mar 29 2024	Df Residuals:	797		
Time:	01:13:44	Df Model:	0		
	Volatility Model				
=====					
	coef	std err	t	P> t	95.0% Conf. Int.

omega	7.5504e-05	1.035e-05	7.293	3.038e-13	[5.521e-05,9.580e-05]
alpha[1]	0.1615	5.800e-02	2.785	5.354e-03	[4.785e-02, 0.275]
beta[1]	0.5962	7.043e-02	8.465	2.571e-17	[0.458, 0.734]
=====					

Covariance estimator: robust					
Zero Mean - GARCH Model Results					
=====					
Dep. Variable:	Log Returns	R-squared:	0.000		
Mean Model:	Zero Mean	Adj. R-squared:	0.001		
Vol Model:	GARCH	Log-Likelihood:	2544.33		
Distribution:	Normal	AIC:	-5082.66		
Method:	Maximum Likelihood	BIC:	-5068.61		
		No. Observations:	797		
Date:	Fri, Mar 29 2024	Df Residuals:	797		
Time:	01:13:44	Df Model:	0		
Volatility Model					
=====					
	coef	std err	t	P> t	95.0% Conf. Int.

omega	1.7765e-06	2.362e-09	752.171	0.000	[1.772e-06,1.781e-06]
alpha[1]	0.0729	2.245e-02	3.249	1.158e-03	[2.894e-02, 0.117]
beta[1]	0.9123	1.847e-02	49.381	0.000	[0.876, 0.948]
=====					

Covariance estimator: robust
Estimated volatility of BAC over the observed period is: 0.017412
Estimated volatility of SPY over the observed period is: 0.010449

```
In [7]: # Merge the 'Adj Close' columns of BAC and SPY into a DataFrame
merged_data = (bac['Adj Close'].to_frame().
merge(spy['Adj Close'].to_frame(),
left_index=True, right_index=True, suffixes=('_BAC', '_SPY')))

# Calculating correlation
correlation = np.round(merged_data['Adj Close_BAC'].corr(merged_data['Adj Close_SPY']), 6)

print("Correlation between BAC and SPY over the observed period:", correlation)

Correlation between BAC and SPY over the observed period: 0.22779
```

```
In [8]: # Define the portfolio values
portfolio_A = portfolio_B = 1000000
portfolio_C = 2000000

bac_mean_return = bac['Returns'].mean()
spy_mean_return = spy['Returns'].mean()

# Historical Simulation VaR (99%)
var_hist_a = np.percentile(bac['Returns'], 1) * portfolio_A
var_hist_b = np.percentile(spy['Returns'], 1) * portfolio_B
var_hist_c = ((np.percentile(bac['Returns'], 1) + np.percentile(spy['Returns'], 1)) / 2) * portfolio_C

# Calculating Expected Shortfall at 99% VaR
es_hist_a = abs(np.mean((bac['Returns'] * 1000000)[(bac['Returns'] * 1000000) <= var_hist_a]))
es_hist_b = abs(np.mean((spy['Returns'] * 1000000)[(spy['Returns'] * 1000000) <= var_hist_b]))
es_hist_c = abs(np.mean(((bac['Returns'] + spy['Returns']) * 2000000)
                        [(bac['Returns'] + spy['Returns']) * 2000000] <= var_hist_c)))

# Calculate 99% VaR using delta normal method
z = norm.ppf(0.99)
var_dn_a = z * bac_mean_return * portfolio_A
var_dn_b = z * spy_mean_return * portfolio_B
var_dn_c = z * 0.5 * (bac_mean_return + spy_mean_return) * portfolio_C

print("99% VaR using historical simulation for Portfolio A: $", np.round(abs(var_hist_a), 2))
print("99% VaR using historical simulation for Portfolio B: $", np.round(abs(var_hist_b), 2))
print("99% VaR using historical simulation for Portfolio C: $", np.round(abs(var_hist_c), 2))

print("Expected Shortfall for 99% VaR for Portfolio A: $", np.round(abs(es_hist_a), 2))
print("Expected Shortfall for 99% VaR for Portfolio B: $", np.round(abs(es_hist_b), 2))
print("Expected Shortfall for 99% VaR for Portfolio C: $", np.round(abs(es_hist_c), 2))

print("99% VaR using delta normal for Portfolio A: $", np.round(abs(var_dn_a), 2))
print("99% VaR using delta normal for Portfolio B: $", np.round(abs(var_dn_b), 2))
print("99% VaR using delta normal for Portfolio C: $", np.round(abs(var_dn_c), 2))

99% VaR using historical simulation for Portfolio A: $ 39274.14
99% VaR using historical simulation for Portfolio B: $ 29555.47
99% VaR using historical simulation for Portfolio C: $ 68829.61
Expected Shortfall for 99% VaR for Portfolio A: $ 49501.06
Expected Shortfall for 99% VaR for Portfolio B: $ 36652.83
Expected Shortfall for 99% VaR for Portfolio C: $ 97433.35
99% VaR using delta normal for Portfolio A: $ 1093.69
99% VaR using delta normal for Portfolio B: $ 1240.44
99% VaR using delta normal for Portfolio C: $ 2334.13
```

```
In [13]: # Assumed risk-free rate
rf = 0.05 # 5%

# Fetch historical price data for March 11, 2024
bac_11_03 = yf.download("BAC", start="2024-03-08", end="2024-03-12")
spy_11_03 = yf.download("SPY", start="2024-03-08", end="2024-03-12")

# Calculate daily returns
bac_return_11_03 = bac_11_03['Adj Close'].pct_change()[~1]
spy_return_11_03 = spy_11_03['Adj Close'].pct_change()[~1]

# Portfolio A: $1 million invested in BAC
portfolio_A_return_11_03 = np.round((bac_return_11_03 * 1000000), 2)
portfolio_B_return_11_03 = np.round((spy_return_11_03 * 1000000), 2)

# Portfolio C: $2 million total invested in BAC and S&P index with $1 million each
portfolio_C_return_11_03 = np.round(((0.5 * bac_return_11_03 +
                                     0.5 * spy_return_11_03) * 2000000), 2)

print("Portfolio A Return on March 11, 2024: $", portfolio_A_return_11_03)
print("Portfolio B Return on March 11, 2024: $", portfolio_B_return_11_03)
print("Portfolio C Return on March 11, 2024: $", portfolio_C_return_11_03)

[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
Portfolio A Return on March 11, 2024: $ 8146.09
Portfolio B Return on March 11, 2024: $ -859.83
Portfolio C Return on March 11, 2024: $ 7286.26
```

```
In [10]: # Calculate the volatility for Portfolio C
volatility_c = np.sqrt((bac_volatility[-1]**2 + spy_volatility[-1]**2) / 2)

# Calculate Sharpe ratio for each portfolio
sharpe_ratio_A = np.round(((bac_return_11_03 - rf) / bac_volatility[-1]), 3)
sharpe_ratio_B = np.round(((spy_return_11_03 - rf) / spy_volatility[-1]), 3)
sharpe_ratio_C = np.round((((0.5 * bac_return_11_03 +
                             0.5 * spy_return_11_03) - rf) / volatility_c), 3)

print("Sharpe ratio for Portfolio A:", sharpe_ratio_A)
print("Sharpe ratio for Portfolio B:", sharpe_ratio_B)
print("Sharpe ratio for Portfolio C:", sharpe_ratio_C)

Sharpe ratio for Portfolio A: -2.759
Sharpe ratio for Portfolio B: -6.292
Sharpe ratio for Portfolio C: -3.814
```

```
In [11]: ## Calculate RAROC for each portfolio
rf_adjusted = rf * (1/365) * 63 # Adjusting for the number of days from Jan 1, 2024, to March 11, 2024

net_income_a = portfolio_A_return_11_03 - (rf_adjusted * portfolio_A)
net_income_b = portfolio_B_return_11_03 - (rf_adjusted * portfolio_B)
net_income_c = portfolio_C_return_11_03 - (rf_adjusted * portfolio_C)

## Define Economic Capital (99% VaR) for each portfolio
ec_a = var_dn_a
ec_b = var_dn_b
ec_c = var_dn_c

## Calculate RAROC for each portfolio
raroc_a = net_income_a / ec_a
raroc_b = net_income_b / ec_b
raroc_c = net_income_c / ec_c

## Print the result
print(f"RAROC for Portfolio A: {raroc_a}")
print(f"RAROC for Portfolio B: {raroc_b}")
print(f"RAROC for Portfolio C: {raroc_c}")

RAROC for Portfolio A: -0.4425814754691938
RAROC for Portfolio B: -7.650474676445932
RAROC for Portfolio C: -4.273115321096678
```

Report Part

Average Returns and Log Returns

Average Returns of BAC over the observed period is 37.52 %.

Average Returns of SPY over the observed period is 42.55 %.

Average Log-Returns of BAC over the observed period is 25.08.

Average Log-Returns of SPY over the observed period is 37.8.

GARCH(1,1) Model Parameters

For BAC:

Omega (ω): 0.000090562 with p-value < 0.00001.

Alpha[1] (α): 0.1484 with p-value = 0.002737.

Beta[1] (β): 0.5600 with p-value < 0.00001.

For SPY:

Omega (ω): 0.00000084157 with p-value \approx 0.

Alpha[1] (α): 0.0521 with p-value \approx 0.

Beta[1] (β): 0.9401 with p-value \approx 0.

The p-values are all significant, which suggests that all the parameters in both models are significantly different from zero at the 99% confidence level.

For BAC:

Display GARCH(1,1) model equation for BAC

$$h_t = 0.000090562 + 0.1484\epsilon_{t-1}^2 + 0.5600h_{t-1}$$

where h_t is the conditional variance (volatility) at time t , and ϵ_{t-1} is the residual from the previous time period.

For SPY

Display GARCH(1,1) model equation for SPY

$$h_t = 0.00000084157 + 0.0521\epsilon_{t-1}^2 + 0.9401h_{t-1}$$

Estimated Volatilities:

The estimated volatility of BAC over the observed period is 0.017445, and for SPY, it is 0.010374.

Correlation:

The correlation between BAC and SPY over the observed period is 0.22779, indicating a positive but relatively weak relationship between the two securities' returns.

VaR and Expected Shortfall:

Expected Shortfall for 99% VaR for Portfolio A: \$ 49501.03.

Expected Shortfall for 99% VaR for Portfolio B: \$ 41897.1.

Expected Shortfall for 99% VaR for Portfolio C: \$ 72269.98.

99% VaR using delta normal for Portfolio A: \$ 1093.69.

99% VaR using delta normal for Portfolio B: \$ 1240.44.

99% VaR using delta normal for Portfolio C: \$ 2334.13.

How much is the diversification benefit in terms of VaR?

Sum of VaRs for Portfolios A and B: $39274.06 + 29555.34 = \$68829.4$

Diversification Benefit = Sum of VaRs for Portfolios A and B - VaR for Portfolio C = $68829.4 - 68829.4 = \$0$

Based on the calculations, the diversification benefit in terms of VaR is \$0. This indicates that there is no diversification benefit achieved by combining assets in Portfolio C compared to holding the individual portfolios A and B separately. This is backed by the very high correlation between BAC and the S&P Index is not optimal for diversification benefits in this case.

Compare the 99% VaR estimates for portfolio (C). Which approach leads to a larger VaR? What could be the reason for this difference?

Based on the VaR data, the VaR from historical simulation for Portfolio C surpasses the VaR for A and B. This indicates that combining investments in BAC and the S&P Index (Portfolio C) results in a larger VaR compared to separate investments in BAC (Portfolio A) and the S&P Index (Portfolio B).

The reasons can be:

(1) The correlation between BAC and the S&P Index

BAC and SPY are positively correlated (0.22779), so the diversification benefits are limited.

(2) The weight of Investments

If the portfolio is not optimally weighted to minimize risk (it equally weights both assets without considering their individual risk contributions), the combined portfolio's VaR could be larger than that of the individual portfolios. This is because the VaR of a combined portfolio is not simply the average of the individual VaRs but a function of the individual risks and their correlation.

(3) Mathematical Nature of VaR:

VaR is not a linear measure, meaning the VaR of a portfolio is not necessarily the sum or the average of the VaRs of its components. Instead, it's influenced by the distribution of returns and the correlations between the assets in the portfolio. If the portfolio allocation, correlation, and individual asset volatilities align unfavorably, the resulting portfolio VaR can be larger.

Use the results above to assess and discuss which portfolio has the best risk adjusted performance, assuming the 99% VaR is used for economic capital.

Portfolio A Return on March 11, 2024: \$ 8146.09.

Portfolio B Return on March 11, 2024: \$ -859.83.

Portfolio C Return on March 11, 2024: \$ 7286.26.

Sharpe ratio for Portfolio A: -2.703.

Sharpe ratio for Portfolio B: -6.471.

Sharpe ratio for Portfolio C: -3.775.

99% VaR using historical simulation for Portfolio A: \$ 39274.06.

99% VaR using historical simulation for Portfolio B: \$ 29555.34.

99% VaR using historical simulation for Portfolio C: \$ 68829.4.

RAROC for Portfolio A: -0.442581438485938.

RAROC for Portfolio B: -7.650472881158458.

RAROC for Portfolio C: -4.27311462089084.

To evaluate and discuss the portfolio with superior risk-adjusted performance, we analyze both returns and risk metrics (VaR and Sharpe ratio).

Return: Portfolio A exhibits the highest return on March 11, 2024, followed by Portfolio C and then Portfolio B.

VaR: Portfolio B demonstrates the lowest VaR, followed by Portfolio A and then Portfolio C, indicating lesser risk among the three portfolios.

Sharpe Ratio: All portfolios have underperformed the risk-free rate, with Portfolio B having the worst risk-adjusted performance, followed by Portfolio C, and Portfolio A being the least negative, the best among the three.

RAROC: Portfolio A performs best, followed by C and B.

Based on this analysis, Portfolio A appears to have the best risk-adjusted performance among the three. It has the highest return and RAROC, a middle-range risk as per the 99% VaR, and the least negative Sharpe ratio, indicating its underperformance relative to the risk-free rate is the smallest.