Contents

MA7008 – Financial Mathematics - Coursework 2022/23

Dinh Viet Tuan – 21041406

INTRODUCTION – PORTFOLIO REVIEW

IBM, or International Business Machines Corporation, is a multinational technology company that works in many areas, such as IT services, cloud computing, artificial intelligence, and business consulting.

Dell Technologies is an American multinational company specializing in developing, selling, and supporting technology products, including personal computers, servers, data storage devices, and networking equipment.

Google is a multinational American technology company specializing in Internet-related services and products. These include search engines, online advertising technologies, cloud computing, software, and hardware.

Microsoft is an American multinational technology company that develops, licenses, and sells computer software, consumer electronics, personal computers, and related services. It is best known for the Windows operating system and the Microsoft Office productivity software suite.

Oracle Corporation is an American multinational computer technology corporation. It specializes in developing and marketing computer hardware and enterprise software products, particularly its brands of database management systems. It also offers services like consulting and training in areas of technology like cloud services, data management, and artificial intelligence, among others.

Historical Data of five stocks were colected from 23/12/2017 to 23/12/2022 (5 years), and downloaded from Yahoo Finance.

In finance, a share price, or stock price, is the cost of purchasing one share of a company's stock. The price of a share changes in response to market circumstances. It is likely to rise if the firm is regarded as doing well and decline if it is not achieving expectations (Agmon, 1972).

The closing price is the price at which a stock was last traded on a given trading day. It could be employed to compute the daily return, which evaluates the stock price change over a single trading day (Muniesa, 2007).

The adjusted closing price takes into account things like stock splits, dividends, and other things that might have changed the price of the stock. This means that the adjusted closing price represents the stock's actual value, taking into account any changes in the number of outstanding shares (De Loecker, Goldberg, Khandelwal, & Pavcnik, 2016).

TASK 1: Calculate the expected return and the volatility for each company as well as the correlations between the asset returns.

Expected return and volatility are basic financial concepts used to figure out the risk-return trade-off of an investment. The "expected return" is the average amount of money an investor can expect back from an investment over time. It is usually given as a percentage and can be figured out by multiplying the chance of each possible return by the return itself. The expected return can be estimated using historical returns or by forecasting future returns (Black, 1993; Martin, 2017).

Volatility, on the other hand, is a measure of the degree of variation or fluctuations in the returns of an investment. It is usually measured by the standard deviation of the returns, which shows how much the returns are likely to differ from the expected return. High volatility indicates that returns are likely to be dispersed and can be positive or negative. Low volatility means the returns are likely more consistent and closer to the expected return (Andersen, Bollerslev, Christoffersen, & Diebold, 2006; Shiller, 1992).

Let's have a look about the data set.

DAILY SHARE PRICES FOR 5 ASSETS					
Date	IBM	DELL	GOOG	MSFT	ORCL
26/12/2017	114.9685	22.28999	52.837002	80.422455	43.498119
27/12/2017	115.19419	22.25712	52.468498	80.714355	43.452251
28/12/2017	115.87876	22.19412	52.407001	80.723793	43.580654
29/12/2017	115.41235	22.2626	52.32	80.554283	43.360554
02/01/2018	116.03672	22.57759	53.25	80.940384	42.764435
03/01/2018	119.22632	22.81588	54.124001	81.31707	43.754898

Table 1: Overview about Stock data

1.1 Calculate Daily Expected Return and Annual Expected Return

To calculate the expected annual return from daily returns, the average daily returns over a period of time were calculated and then annualize it. Here is the process for calculating the expected annual return from daily returns:

- Collect the daily returns for the period of time you are interested in.
- Calculate the daily return by formula :
    - $\boldsymbol{Daily\ return} = \frac{(\boldsymbol{Adj.Closed\ Price}(\ \boldsymbol{day\ n}) - \boldsymbol{Adj.Closed\ Price}\ (\boldsymbol{day}(\ \boldsymbol{n}-\mathbf{1})))}{\boldsymbol{Adj.Closed\ Price}\ (\boldsymbol{day\ n}-\mathbf{1})}$
- Calculate the average daily return by formula:
    - $\boldsymbol{Average\ daily\ return} = \frac{\sum_{i=0}^{i=1260} \boldsymbol{Daily\ return}(\boldsymbol{i})}{\mathbf{1260}}$
- Annualize the average daily return by formula:
    - $\boldsymbol{Annual\ average\ return} = \boldsymbol{Average\ daily\ return} * \mathbf{252}$

Here is the result:

Stock	Daily Expected Return	Daily Standard Dev	Annual Expected Return	Annual STDEV
IBM	0.050%	2.482%	12.704%	39.405%
DELL	0.037%	1.984%	9.245%	31.489%
GOOG	0.081%	1.969%	20.520%	31.263%
MSFT	0.042%	1.885%	10.705%	29.931%
ORCL	0.049%	1.869%	12.263%	29.664%

Table 2: Annual Expected Return

1.2 Calculate Correlations Between Five Stocks

Correlation is a statistical measure that shows how much the prices of two investments move together (Morck, Yeung, & Yu, 2000). A positive correlation means that when one investment goes up, the other usually increases. A negative correlation means that when one investment goes up, the other investment tends to go down. A correlation coefficient of 1 indicates a perfect positive correlation; a coefficient of -1 indicates a perfect negative correlation, and a coefficient of 0 indicates no correlation.

By using this formula, correlation can be calculated:

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

Where,

r = Pearson Correlation Coefficient

x_i = x variable samples $\qquad$ y_i = y variable sample

$\bar{x}$ = mean of values in x variable $\qquad$ $\bar{y}$ =mean of values in y variable

Here is the result:

Correlation between five stocks					
	IBM	*DELL*	*GOOG*	*MSFT*	*ORCL*
IBM	1				
DELL	0.47	1			
GOOG	0.46	0.487	1		
MSFT	0.5	0.52	0.81	1	
ORCL	0.51	0.43	0.52	0.61	1

Table 3: Correlations Between Five Stocks

The result indicates a moderate-to-strong positive correlation between the stocks. With a coefficient of 0.81, GOOG and MSFT have the highest correlation. This means that they are strongly linked. The lowest correlation is between DELL and ORCL, with a coefficient of 0.43, indicating a moderate positive correlation. The other correlations are between IBM and DELL (0.57), IBM and GOOG (0.46), IBM and MSFT (0.5), IBM and ORCL (0.51), and DELL and MSFT (0.52). The stocks tend to move in the same direction, but not necessarily at the same magnitude.

TASK 2: Using Solver Function and Draw Efficient Frontier

2.1 Use an appropriate Solver function to determine the portfolio risk and the percentage of investment in each asset in your portfolio for a target return of your choice.

The Solver function in Excel is a powerful tool for financial mathematics (Vasilev, Turygina, Kosarev, & Nazarova, 2016). It allows finding the optimal solution for problems with multiple variables and constraints. It can be used for optimization, forecasting, and simulation problems by defining the objective, variables, and constraints. The objective is for the cells that want to optimize, such as for maximum return or minimizes risk. The constraints are the conditions that the solution must meet. The variables are the cells that are used to figure out the goal. The Solver function uses mathematical algorithms to find the optimal solution and can be used for multiple objectives, constraints, and non-linear problems.

First of all, Variance and Covariance need to be calculated.

Mathematically, the variance of a random variable X is defined as:

$$\mathbf{VAR(X)} = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - E(X))^2$$

where n is the number of observations

x_i is the individual observation

And the covariance of 2 random variables X and Y is defined as:

$$\mathbf{COV(X, Y)} = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - E(X))(y_i - E(Y))$$

Here is the result:

Variance - Corvariance Matrix					
	IBM	*DELL*	*GOOG*	*MSFT*	*ORCL*
IBM	0.07665	0.05153	0.04026	0.04315	0.04216
DELL	0.05153	0.15425	0.05990	0.06335	0.05012
GOOG	0.04026	0.05990	0.09818	0.07836	0.04788
MSFT	0.04315	0.06335	0.07836	0.09628	0.05594
ORCL	0.04216	0.05012	0.04788	0.05594	0.08792

Table 4: Variance – Covariance Matrix

All the highlight values in table 4 are the variance of 5 stocks, and other cells are covariance between 2 stocks.

After that, A equally weighted portfolio was build.

Solver Function was applied to calculate Expected Return, Std dev and Shape R. In this report, the risk-free rate is 0.12%.

Equally weighted portfolio	
Stocks	Weights
IBM	0.2
DELL	0.2
GOOG	0.2
MSFT	0.2
ORCL	0.2
Sum	1
Expected Return	0.119355406
Standard Dev	0.251286107

Table 5: Equally weight portfolio

2.2 Solver Function:

Option 1: Maximize the Expected Return

Constrants were used:

- Weight of all assests are higher than 0
- Total Weight equal 1
- Max Objective “Expected Return” by changing Weight of five assests.

High Return Portfolio	
Stocks	Weights
IBM	0.0000
DELL	0.0000
GOOG	0.0000
MSFT	1.0000
ORCL	0.0000
Sum	1.0000
Expected Return	0.2177
Standard Dev	0.3103

Table 6: High Return Portfolio

The table 6 indicates that the portfolio have the highest expected return when investor invest 100% capital to MSFT stock.

Opiton 2: Minimize the standard deviation

Constrants was used:

- Weight of all assests are higher than 0
- Total Weight equal 1
- Min Objective "Standard Dev" by changing Weight of five assests.

Low Risk Portfolio	
Stocks	**Weights**
IBM	0.4317
DELL	0.0353
GOOG	0.2161
MSFT	0.0366
ORCL	0.2802
Sum	1.0000
Expected Return	0.0861
Standard Dev	0.2388

Table 7: Low Risk Portfolio

To get the lowest Standard Deviation, the investor should invest 43.17% on IBM, 3.53% on DELL, 21.61% on GOOG, 3.66% on MSFT and 28.02% on ORCL stocks.

2.3 Draw the efficient frontier curve.

An efficient frontier is a curve that shows the set of optimal portfolios that offer the highest expected return for a given level of risk. It is plotted with volatility on the x-axis and expected return on y-axis. It is characterized by a positive slope, showing the trade-off between risk and return. The shape and location of the curve depend on the underlying assets (Anderson, Ghysels, & Juergens, 2009).

Here are portfolios that offer the minimum risk at a given expected portfolio return:

Efficient Frontier																		
	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17	W18
IBM's weight	0.8510	0.7187	0.5896	0.460	0.421	0.393	0.366	0.338	0.311	0.277	0.227	0.176	0.126	0.076	0.026	0	0	0
DELL's Weight	0.0000	0.0000	0.0011	0.032	0.035	0.035	0.035	0.034	0.032	0.032	0.028	0.023	0.018	0.013	0.008	0	0	0
GOOG's Weight	0.1490	0.1911	0.2197	0.236	0.200	0.158	0.115	0.073	0.032	0.000	0.000	0.000	0.000	0.000	0.000	0	0	0
MSFT's Weight	0.0000	0.0000	0.0000	0.000	0.065	0.137	0.210	0.282	0.355	0.425	0.486	0.548	0.609	0.671	0.732	0.81	0.92	1
ORCL's Weight	0.0000	0.0902	0.1896	0.272	0.279	0.277	0.274	0.272	0.270	0.266	0.260	0.253	0.247	0.240	0.233	0.19	0.08	0
Volatility	0.261	0.249	0.2419	0.239	0.239	0.239	0.241	0.243	0.246	0.249	0.254	0.259	0.265	0.273	0.281	0.29	0.3	0.31
Exp Return	0.050	0.060	0.0700	0.080	0.090	0.100	0.110	0.120	0.130	0.140	0.150	0.160	0.170	0.180	0.190	0.2	0.21	0.218

Table 8: Different Weights by Solver Function

Here is the portfolio optimised weights chart:

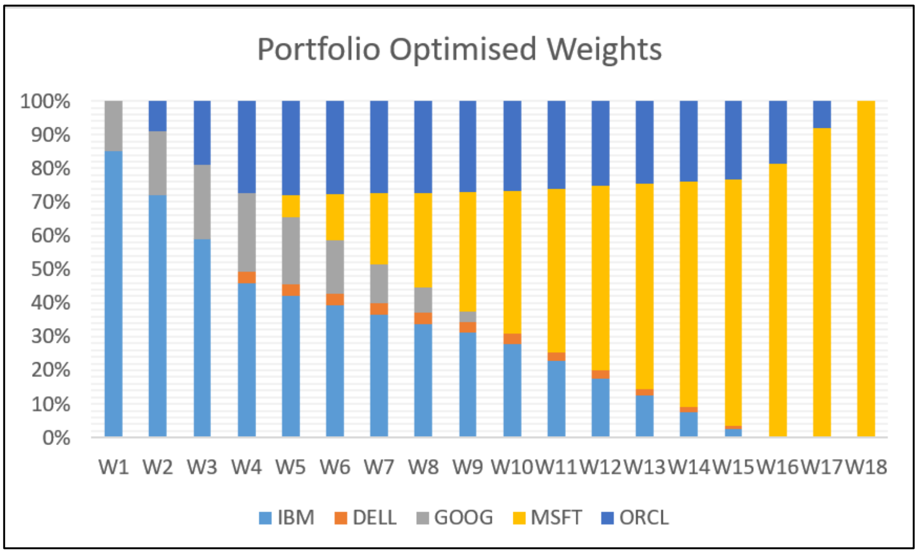


Figure 1: Portfoli Optimised Weights

From figure 1, it can be seen that to get an increasing expected return (from W1 to W18), investors should invest more in MSFT stock and reduce investment in the remaining four stocks, especially shares of IBM and GOOG. Most portfolios only have a small amount of DELL stock, which means that having or not having DELL stock doesn't change the expected return or volatility of the portfolio.

From this data of Volatility and Expected Return, an Efficient Frontier Curve was created.

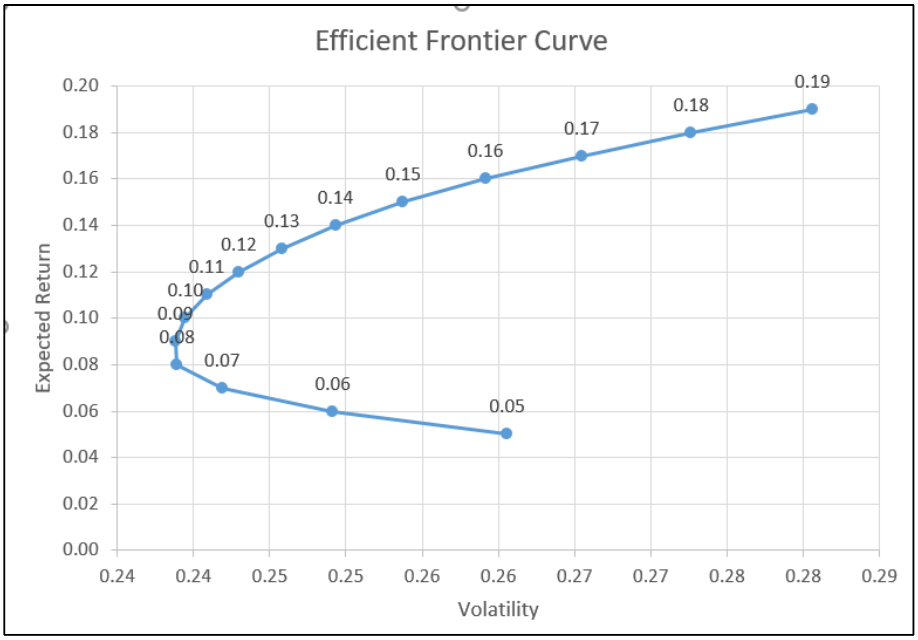


Figure 2: Efficient Frontier Curve

The slope of the efficient frontier is not steep, indicating a weak trade-off between risk and return. The location of the curve is in the lower-left corner, suggesting high-risk, low-return investments. Specific points like the tangency portfolio, the highest Sharpe ratio, is when the expected return equals 16%, and the minimum variance portfolio, which is the least risky, is when the expected return equals 8%.

Task 3: Calculate the Sharpe ratios and determine the equation of the Capital Market Line

3.1 Calculate the Sharpe Ratios

The Sharpe ratio is a measure that compares the return of a portfolio to that of a risk-free investment, adjusting for its volatility. It is used to evaluate the risk-adjusted performance of a portfolio in financial mathematics. The higher the Sharpe ratio, the better the portfolio's risk-adjusted return.

The formula to calculate Sharpe Ratio is defined as:

$$\text{Sharpe Ratio} = \frac{R_p - R_f}{\sigma_p}$$

- R_p Return of portfolio
- R_f Risk-free rate
- σ_p Standard deviation of portfolio's excess return

By using the risk free rate is 1.5%, here is the result:

Efficient Frontier																		
	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17	W18
IBM's weight	0.8510	0.7187	0.5896	0.460	0.421	0.393	0.366	0.338	0.311	0.277	0.227	0.176	0.126	0.076	0.026	0	0	0
DELL's Weight	0.0000	0.0000	0.0011	0.032	0.035	0.035	0.035	0.034	0.032	0.032	0.028	0.023	0.018	0.013	0.008	0	0	0
GOOG's Weight	0.1490	0.1911	0.2197	0.236	0.200	0.158	0.115	0.073	0.032	0.000	0.000	0.000	0.000	0.000	0.000	0	0	0
MSFT's Weight	0.0000	0.0000	0.0000	0.000	0.065	0.137	0.210	0.282	0.355	0.425	0.486	0.548	0.609	0.671	0.732	0.81	0.92	1
ORCL's Weight	0.0000	0.0902	0.1896	0.272	0.279	0.277	0.274	0.272	0.270	0.266	0.260	0.253	0.247	0.240	0.233	0.19	0.08	0
Volatility	0.261	0.249	0.2419	0.239	0.239	0.239	0.241	0.243	0.246	0.249	0.254	0.259	0.265	0.273	0.281	0.29	0.3	0.31
Exp Return	0.050	0.060	0.0700	0.080	0.090	0.100	0.110	0.120	0.130	0.140	0.150	0.160	0.170	0.180	0.190	0.2	0.21	0.218
Sharpe Ratio	0.13	0.24	0.29	0.33	0.38	0.42	0.46	0.49	0.53	0.56	0.59	0.62	0.64	0.66	0.68	0.69	0.70	0.70
Sharpe Ratio = (Expected Return - Risk_Free)/ Standard Dev																		

Table 9: Calculate the Sharpe Ratio

3.2 Determine the equation of the Capital Market Line. Discuss the economic significance of the Capital market Line.

From the Efficent Frontier Curve, now an Opimal Portfolio is found out. The aim here in this portfolio optimization is to have the best set of weight, the weight that could guarantee the minimum risk or the least std at a corresponding highest expected return. The goal is to maximize the sharp ratio, which should not be negative.

Capital Allocation Line		
Weight	**Return**	**Standard Dev**
0%	1.50%	0.00%
100%	16.00%	25.90%
150%	24.00%	38.85%
Optimal Portfolio		
Exp Return	0.160	
Std Dev	0.259	
Sharpe Ratio	0.620	
Risk-free	0.015	

Table 10: Opimal Portfolio and Capital Allocation Line

The Capital Market Line (CML) is a graphical representation of the relationship between the expected returns and risk of a portfolio. It shows the efficient frontier, which is the set of portfolios with the highest expected returns for a given level of risk, and the risk-free asset, which is the investment with the lowest possible risk.

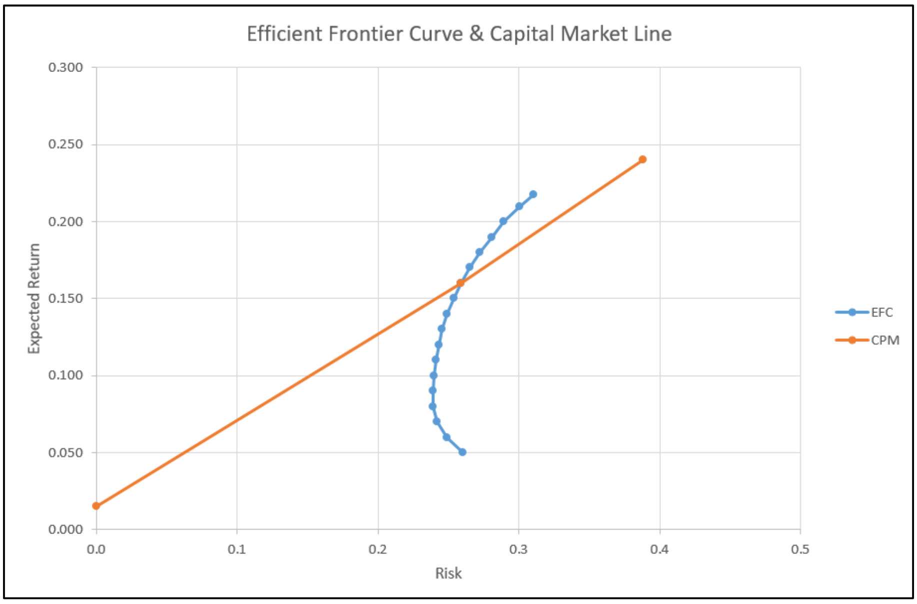


Figure 3: Efficient Frontier Curve and Capital Market Line

The Capital Market Line (CML) is a decision-making tool that is often used in finance and investments. It plots the expected returns and risks of different portfolios, so investors and financial managers can see which portfolios have the highest expected returns for a given level of risk and match their portfolios to their investment goals.

Also, the CML shows the trade-off between risk and return. This makes it easy to see how these two things are related and how to find the right balance between them to get the risk and return levels investors want.

Also, the CML is an excellent way to compare the risk and return of a portfolio to the market. It lets investors and financial managers compare the portfolio on the CML to where the market portfolio is, so they can determine if it is doing better or worse than the market portfolio.

In short, the CML is vital to the economy because it helps investors and financial managers make intelligent choices about their portfolios, understand the trade-off between risk and return, and compare the risk and return of a portfolio to the market.

TASK 4: Calculate the *beta* for each asset and provide discussion. Calculate how much each asset contributes to estimated VaR 5%.

4.1 Using linear regression analysis, calculate the *beta* for each asset in the portfolio and discuss the significance of this quantity.

First step, we will look at the overal view of stocks prices over time.

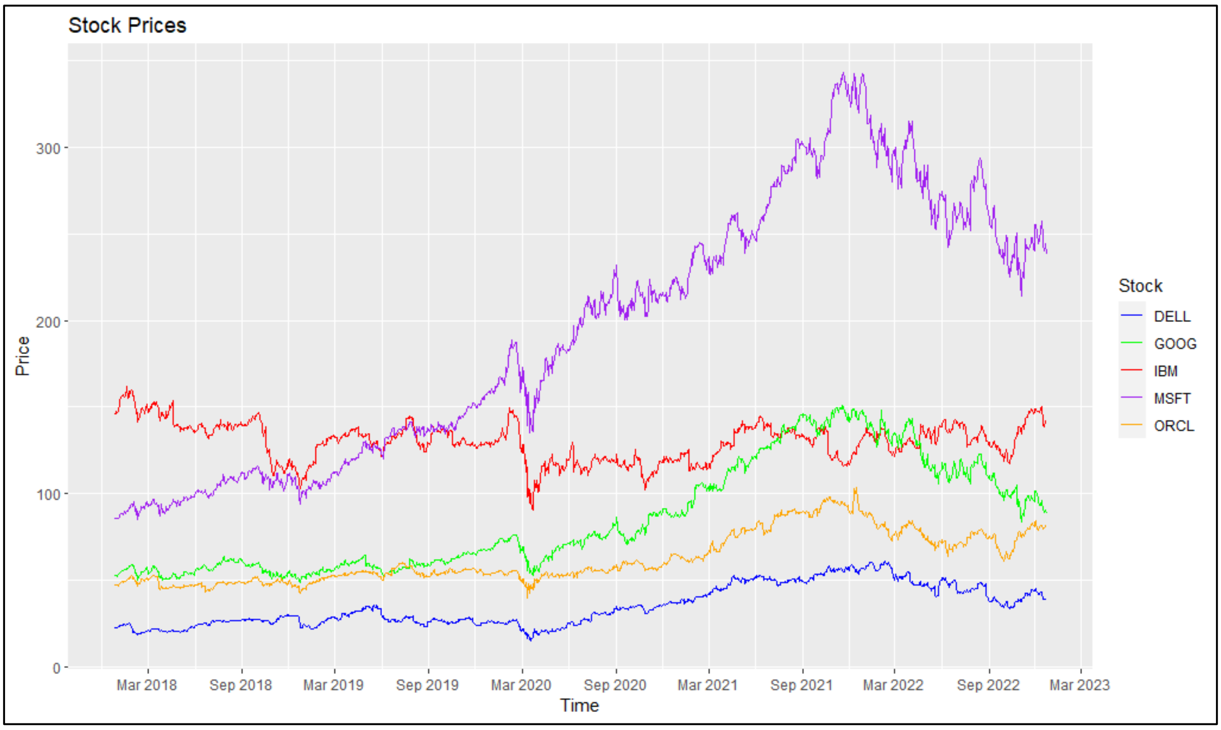


Figure 4: Stocks Prices over time

In December 2018, stock DELL was priced at $25, stock GOOG at $52, stock IBM at $145, stock MSFT at 80$, and stock ORCL at $48. Over the next five years, stock MSFT experienced the most fluctuation in price, reaching as high as $340 in December 2021 before dropping down to $250 in 2022. On the other hand, stock DELL had the least fluctuation and saw steady growth, ending at $45 in December 2022. Stock IBM had moderate fluctuation and ended at $140 in December 2022, while stock GOOG and ORCL had a similar trend and ended at $80 and $75 respectively.

Next we will have the look at the Portfolio Return Performance

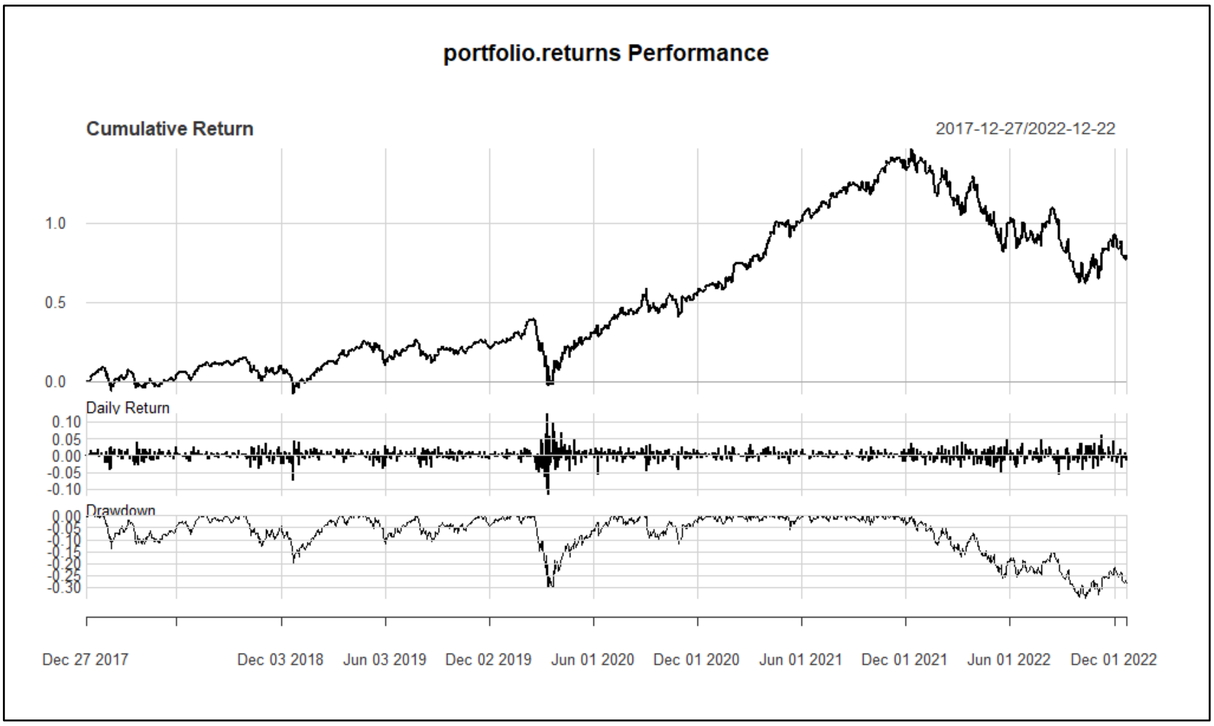


Figure 5: Portfolio Return Performance Chart

The chart has several components, including a line chart of the cumulative returns of the portfolio over time, a line chart of drawdown, and a series of bar charts that display the distribution of returns.

This top line chart provides a visual representation of the performance of a portfolio over a five-year period, from December 2017 to December 2022. The chart shows the cumulative return on the y-axis and the date on the x-axis. A value of 0 means that the portfolio has not generated any returns since the starting point, a value of 0.5 means that the portfolio has generated a 50% return since the starting point, and a value of 1 means that the portfolio has generated a 100% return since the starting point.

The portfolio's cumulative return demonstrated a moderate uptrend from December 2017 until March 2020. However, due to market volatility and the impact of unforeseen events, the portfolio's cumulative return experienced a sharp decline, reaching a low point of 0 in March 2020. Subsequently, the portfolio's cumulative return rebounded, reaching a peak of 1.5 in December 2021. Despite this recovery, the portfolio's cumulative return experienced a slight decrease in the final months of 2022, closing at 0.8 in December 2022.

The chart enables the identification of overall trends and patterns in the portfolio's returns, as well as the time period of highest and lowest returns. Furthermore, it provides a comprehensive overview of the portfolio's risk-return profile. This information is crucial for making informed decisions regarding the portfolio's management, allocation, and diversification strategy.

The bar chart of the portfolio's daily returns from December 2017 to December 2022 shows that the portfolio's returns were mostly stable, with the majority of returns falling within the range of -0.05 to 0.05. However, there was a period of increased volatility in March and April 2020, where returns fluctuated between -0.1 and 0.12.

And the last line chart at the bottom is showing the drawdown. The graph displays the fluctuations in the portfolio's drawdown, which is the percentage loss from the highest point reached. The graph indicates that for most of the time the drawdown fluctuated between 0 and -0.1. However, in March and April of 2020, there was a significant drop to -0.3. After that, the drawdown gradually increased again and remained around 0 for nearly 2 years. In December 2022, the drawdown slightly dropped to -0.2. Overall, the graph illustrates the portfolio's performance during the given period, then the portfolio had a significant drop in drawdown in 2020 and a slight drop in December 2022, with relatively stable performance in between.

Calculate BETA:

```
> #Calculate Metrics with the risk-free rate =1.5% per year
> # CAPM.beta function is used to calculate beta
> CAPM.beta(portfolioReturn, benchmarkReturns, .015/252)
[1] 1.09106
>
> # CAPM.beta.bull function is used to calculate beta during bullish market
> CAPM.beta.bull(portfolioReturn, benchmarkReturns, .015/252)
[1] 1.134165
>
> # CAPM.beta.bear function is used to calculate beta during bearish market
> CAPM.beta.bear(portfolioReturn, benchmarkReturns, .015/252)
[1] 1.055519
```

Figure 6: Result of Caculation of Beta (See Appendix 2)

The Capital Asset Pricing Model (CAPM) was used to calculate the beta of the portfolio in relation to the benchmark returns. Beta is a measure of a portfolio's volatility in relation to the market, with a beta of 1 indicating that the portfolio's volatility is the same as the market's volatility, a beta greater than 1 indicating that the portfolio is more volatile than the market, and a beta less than 1 indicating that the portfolio is less volatile than the market.

The overall beta for the portfolio was 1.09106, which means that the portfolio's volatility is 9.106% greater than the market's volatility. During bullish market conditions, the portfolio's beta was 1.134165, indicating that it is 13.4165% more volatile than the market. Similarly, during bearish market conditions, the portfolio's beta was 1.055519, indicating that it is 5.5519% more volatile than the market. These results provide valuable insight into the portfolio's volatility and risk in relation to the market, and can be used to inform investment decisions.

```
> #CAPM.alpha(portfolioReturn, benchmarkReturns, .015/252)
> CAPM.jensenAlpha(portfolioReturn, benchmarkReturns, .015/252)
[1] 0.04061959
>
> SharpeRatio(portfolioReturn, Rf = .015/252, p = 0.95, FUN = "StdDev",
+             weights = NULL, annualize = FALSE)
                                portfolio.returns
StdDev Sharpe (Rf=0%, p=95%):          0.03226752
>
> table.AnnualizedReturns(portfolioReturn, Rf=.015/252, geometric=TRUE)
                            portfolio.returns
Annualized Return                      0.1209
Annualized Std Dev                     0.2593
Annualized Sharpe (Rf=1.5%)            0.4018
```

Figure 7: CAPM Alpha and tables of sumary result.(See Appendix 2)

The portfolio performance was analyzed using various metrics, including the Jensen Alpha, Sharpe Ratio, and annualized returns. The Jensen Alpha, a measure of a portfolio's excess return compared to the market, was found to be 0.04. The Sharpe Ratio, a measure of risk-adjusted return, was 0.03. The annualized return for the portfolio was 0.12, with an annualized standard deviation of 0.26 and an annualized Sharpe Ratio of 0.40. Overall, the portfolio shows a positive excess return compared to the market, but with a low risk-adjusted return.

4.2 Estimate Value at Risk (5%) for your portfolio and discuss how much each asset contributes to your estimated VaR.

In financial context, Value at Risk (VaR) is a measure of the potential loss of an investment portfolio over a given time period, such as a day or a week, with a certain level of confidence. VaR is typically expressed as a dollar amount or a percentage of the portfolio's value. In this part, the VaR is calculated with a probability level of 5%.

VaR is a commonly used risk management tool in finance, and it is often used by financial institutions and investment managers to evaluate and manage the risk of their portfolios. The VaR calculation can be used to identify the potential worst-case scenario for a portfolio, and it can help investment managers to make informed decisions about how to manage risk.

It is important to note that VaR is a statistical measure and it doesn't take into account the worst possible loss or tail event, for those kinds of event it is recommended to use other risk measures like Expected Shortfall (CVaR) or Tail Value at Risk (TVaR)

By using Rstudio (see appendix) , we had this result:

```
> # Print the VaR results
> print(PorVaR.Gaus)
$VaR
[1] 0.02564755

$contribution
  IBM.Close  DELL.Close  GOOG.Close  MSFT.Close  ORCL.Close
0.004214758 0.006161184 0.005272163 0.005405277 0.004594173

$pct_contrib_VaR
 IBM.Close DELL.Close GOOG.Close MSFT.Close ORCL.Close
 0.1643337  0.2402250  0.2055620  0.2107521  0.1791271
```

Figure 8: Result of Contribution of each asset (See Appendix 3)

The first line of output, $VaR, is the Value-at-Risk (VaR) of the portfolio, calculated using the Gaussian method, which is a commonly used method for VaR calculation. The value represents the potential loss of the portfolio, given a specified level of confidence (in this case, 5%).

The second line of output, $contribution, shows the dollar value of the VaR contributed by each individual component of the portfolio. It represents the estimated loss of each component of the portfolio in case of the worst 5% of scenarios.

The third line of output, $pct_contrib_VaR, shows the percentage of the VaR contributed by each individual component of the portfolio. It represents the estimated percentage of loss of each component of the portfolio in case of the worst 5% of scenarios.

In overall, the report shows the Value at Risk (VaR) for a portfolio consisting of five stocks: IBM, DELL, GOOG, MSFT, and ORCL. The VaR is calculated using the Gaussian method. The portfolio is assumed to be equally weighted among the five stocks.

The VaR for the portfolio is calculated to be 0.02564755. This means that there is a 5% chance that the portfolio will lose more than this amount on any given day.

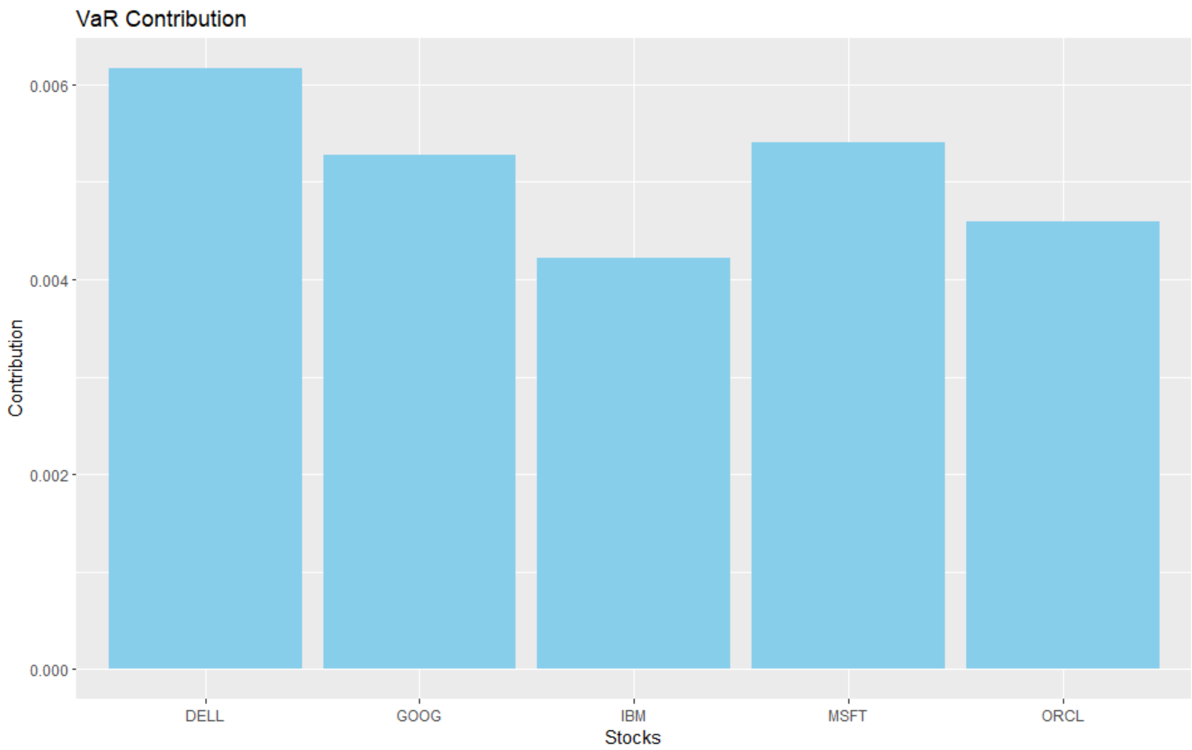


Figure 9: Graph of Contribution of each stocks

The report also shows the contribution of each stock to the overall VaR of the portfolio. IBM has the lowest contribution with 0.004214758 and DELL has the highest contribution with 0.00616.

In terms of percentage contribution, IBM has the lowest percentage contribution with 16.43337% while DELL has the highest percentage contribution with 24.02250%. This indicates that DELL contributes more to the overall risk of the portfolio than the other stocks.

In the financial context, the VaR is an important measure of risk and the contributions of individual stocks to the overall VaR can be used to identify the riskiest stocks in the portfolio.

This information can be used to make informed investment decisions and to construct a more diversified portfolio. Additionally, the percentage contributions to VaR can be used to determine how much of the portfolio's overall risk is due to each individual stock.

Task 5: Estimate the volatility of a single asset in your portfolio using ARCH/GARCH. Choosing best model and provide explanation.

This task is to figure out how volatile a single asset in a portfolio is by using the ARCH/GARCH model and its extensions. The purpose of this task is to better understand the level of risk associated with the asset and make more informed investment decisions. The ARCH/GARCH model, which stands for Autoregressive Conditional Heteroskedasticity/Generalized Autoregressive Conditional Heteroskedasticity, is a statistical model commonly used in finance and economics to analyze and predict changes in volatility over time. The task will be performed using the R programming language and the "rugarch" and “fGarch” packages, which provide a wide range of options for specifying the model and various diagnostic tools for assessing the fit of the model to the data. The result of this task will be an estimate of how volatile the asset in question is. This will help investors figure out how risky the investment is and make intelligent decisions about where to invest.

For this analysis, the daily closing prices of Microsoft (MSFT) stock from the past were used. To begin with, all the necessary libraries were imported. The data analysis was performed using the R programming language. The first step was to use the getSymbols.yahoo() function on Yahoo Finance to get information about past MSFT stock prices. This function was configured with the appropriate parameters, such as the symbol "MSFT", the date range from "2017-12-23" to "2022-12-23", a periodicity of "daily" and the auto.assign parameter was set to false.

The data was filtered to obtain the closing prices of the stock. A graph was then created to represent the MSFT stock prices.

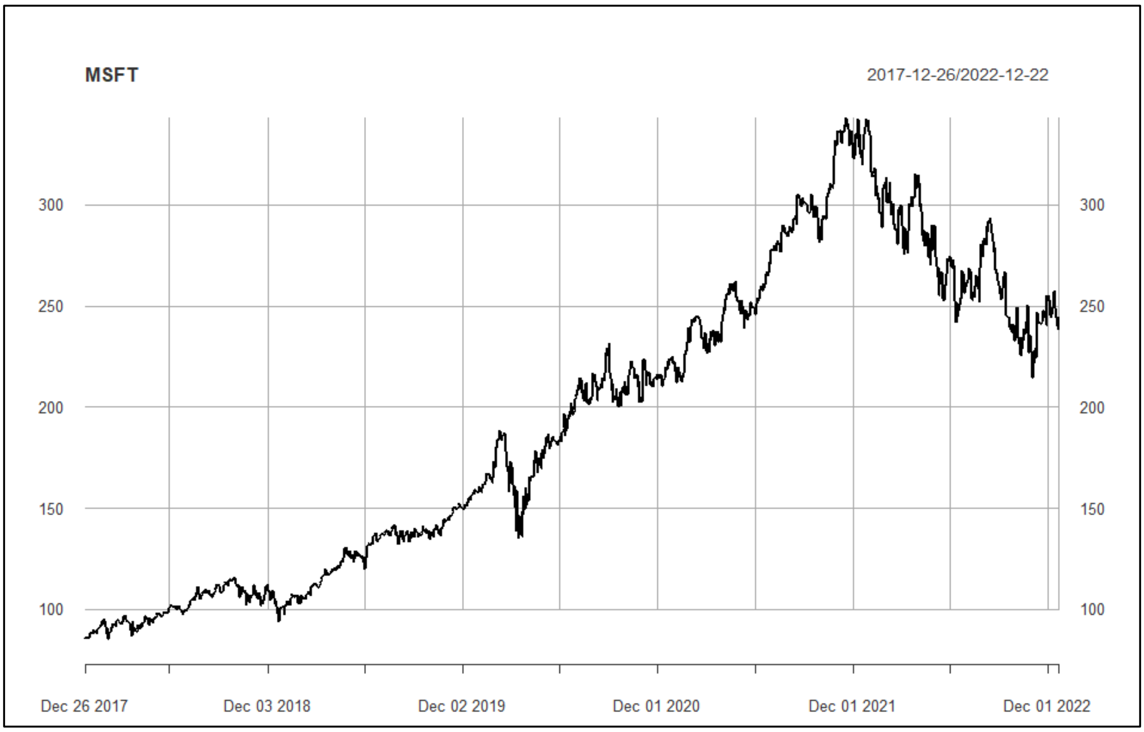


Figure 10: MSFT stock prices

And here is the graph about MSFT stocks's risk over time.

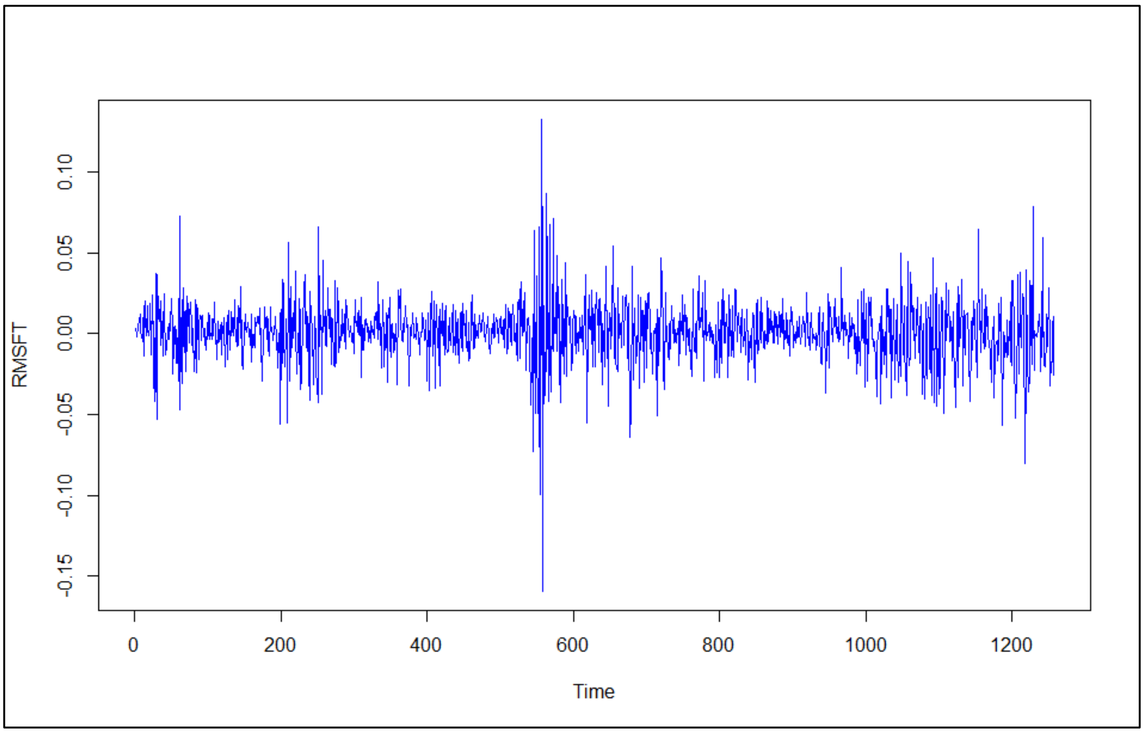


Figure 11 : MSFT stock's risk over time series

Afterwards, the MSFT data was transformed into a zoo object for plotting purposes. To test for data stationarity, a new time series object was created to represent the first difference of the natural log of the closing prices.

```
> # Check stationary of data
> adf.test(RMSFT)

        Augmented Dickey-Fuller Test

data:  RMSFT
Dickey-Fuller = -11.539, Lag order = 10, p-value = 0.01
alternative hypothesis: stationary
```

Figure 12: Stationarity of data(See Appendix 4)

The result indicates that the p-value is lower than 0.05 and "alternative hypothesis": stationary means data of MSFT stock can be used for ARCH/GARCH models.

Several GARCH models were fitted to the time series RMSFT using imported packages in R. The different models are specified by their orders: c(1,0), c(1,1), c(2,0), c(2,1), c(2,2), c(0,2), and c(1,2), respectively. It creates a list of these GARCH models and then extracts the AIC values from the models using the sapply() function and the AIC function. After that, it finds the model's index with the lowest AIC value using the which.min() function. Then, it selects the best model using the best_model_index and prints the best model's AIC value using the print() function. And then the best model is GARCH(1,1) because its AIC is lowest.

```
> print((best_model))

Call:
garch(x = RMSFT, order = c(1, 1), trace = FALSE)

Coefficient(s):
       a0         a1         b1  
1.199e-05  1.595e-01  8.153e-01  
```

Figure 13: The best GARCH model(See Appendix 4)

Here is a detail information about model GARCH(1,1):

```
Title:
 GARCH Modelling

Call:
 garchFit(formula = ~garch(1, 1), data = RMSFT, trace = F)

Mean and Variance Equation:
 data ~ garch(1, 1)
<environment: 0x00000217130df158>
 [data = RMSFT]

Conditional Distribution:
 norm

Coefficient(s):
        mu       omega      alpha1       beta1
1.3894e-03  1.1582e-05  1.6199e-01  8.1473e-01

Std. Errors:
 based on Hessian

Error Analysis:
        Estimate  Std. Error  t value Pr(>|t|)
mu     1.389e-03   4.001e-04    3.473 0.000515 ***
omega  1.158e-05   3.580e-06    3.236 0.001214 **
alpha1 1.620e-01   2.700e-02    5.999 1.98e-09 ***
beta1  8.147e-01   2.896e-02   28.131  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Log Likelihood:
 3355.547    normalized:  2.669488

Description:
 Sat Jan 14 21:51:41 2023 by user: Darius


Standardised Residuals Tests:
                                Statistic p-Value
 Jarque-Bera Test   R    Chi^2  105.6444  0
 Shapiro-Wilk Test  R    W      0.9846611 2.953828e-10
 Ljung-Box Test     R    Q(10)  12.37654  0.2606461
 Ljung-Box Test     R    Q(15)  15.54809  0.412704
 Ljung-Box Test     R    Q(20)  21.46514  0.3702236
 Ljung-Box Test     R^2  Q(10)  9.584502  0.4776702
 Ljung-Box Test     R^2  Q(15)  18.88241  0.2191121
 Ljung-Box Test     R^2  Q(20)  22.83044  0.2971969
 LM Arch Test       R    TR^2   17.06615  0.1471257

Information Criterion Statistics:
      AIC       BIC       SIC      HQIC
-5.332612 -5.316267 -5.332632 -5.326469
```

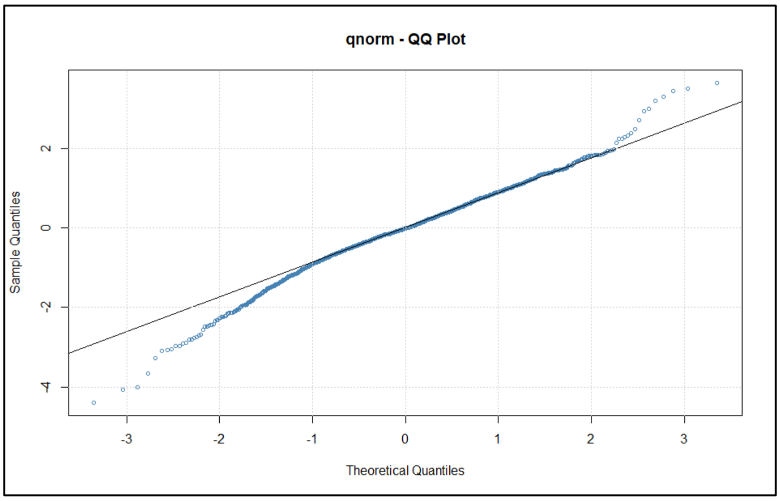
Figure 14: Summary of the model GARCH(1,1)

First, the summary of model GARCH(1,1) shows that p-value of omega, alpha1 and beta1 are very low, lower than 0.05. So the model has statistically significant.

Second, all of the Ljung-Box test indicate the p-value higher than 0.05. When the p-value is higher than 0.05, it indicates that there is not enough evidence to reject the null hypothesis. This means that there is a high probability that the residuals are independently distributed, and that there is no autocorrelation present in the residuals.

Overall, here is a good GARCH model.

Here are some figures from the built model:



The points align along a straight line, indicating that the sample data follows the same distribution as the theoretical distribution.

Figure 15: Q-Q plot

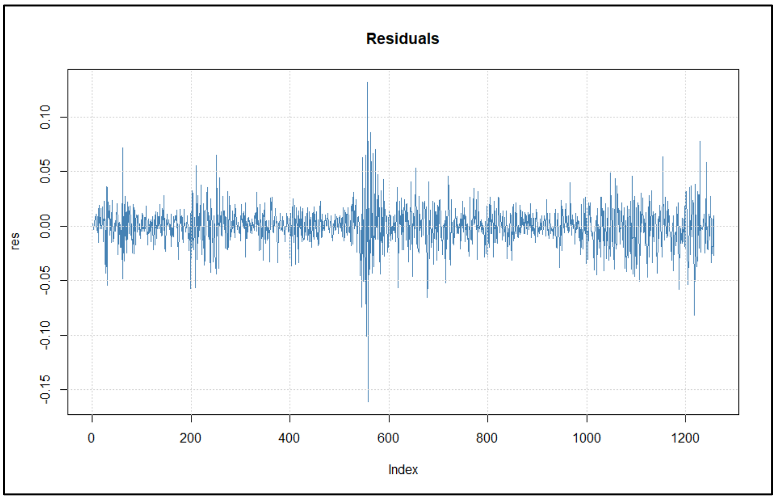


Figure 16: Residual

When the residuals of a model fall within a narrow range, typically around zero, it suggests that the model is a good fit for the data. In this case, the range of residuals being between -0.05 and 0.05, , so model is able to explain most of the variation in the data.

Subsequently, predictions were made regarding the Microsoft (MSFT) stock for the next five days. The data analysis revealed that there is a 99% probability that the MSFT stock will not decrease more than 2.076% on day T+1, 2.09% on the day T+2, 2.116% on the day T+3 and similar probabilities were calculated for the remaining days.

```
0-roll forecast [T0=1257-01-01]:
       Series    Sigma
T+1 0.001412 0.02076
T+2 0.001412 0.02096
T+3 0.001412 0.02116
T+4 0.001412 0.02136
T+5 0.001412 0.02156
```

Figure 17 : Forecast Volitality of MSFT in next 5 days

TASK 6: Summary of findings and discuss the implication of other relevant performance measurements.

The portfolio under consideration for investment is composed of a diverse set of companies, including IBM (a technology company), Dell Technologies (a technology product company), Google (an internet-related services and products company), Microsoft (a software and technology company), and Oracle (a computer technology corporation). Historical stock prices for these companies were collected over a five-year period (from December 23rd, 2017 to December 23rd, 2022) and downloaded from Yahoo Finance.

The expected returns and volatilities for each company, as well as the correlations between the asset returns, were calculated utilizing appropriate financial methods. The annual expected returns range from 9.245% for Dell Technologies to 20.52% for Google, and the annual volatilities range from 29.664% for Oracle to 39.405% for IBM. The correlations between the asset returns range from 0.43 for Dell Technologies and Oracle to 0.81 for Google and Microsoft.

The efficient frontier curve was also plotted, displaying the relationship between portfolio risk and return for different target returns. The slope of the efficient frontier is not steep, indicating a weak trade-off between risk and return. The location of the curve is in the lower-left corner, suggesting high-risk, low-return investments. Specific points, such as the tangency portfolio, the highest Sharpe ratio, occurs when the expected return equals 16%, and the minimum variance portfolio, which is the least risky, occurs when the expected return equals 8%.

The Sharpe ratios for a range of expected portfolio returns and volatilities were also calculated, with a range from 0.13 to 0.7. The equation of the Capital Market Line (CML) was also calculated, with a value of 0.16. The beta for each asset in the portfolio was also calculated, with an overall beta of 1.09106, indicating that the portfolio's volatility is 9.106% greater than the market's volatility.

Furthermore, the beta for each asset in the portfolio was calculated, and it was found that the overall beta for the portfolio was 1.09106, which means that the portfolio's volatility is 9.106% greater than the market's volatility. During bullish market conditions, the portfolio's beta was 1.134165, indicating that it is 13.4165% more volatile than the market. Similarly, during bearish market conditions, the portfolio's beta was 1.055519, indicating that it is 5.5519% more volatile than the market. These results provide valuable insight into the portfolio's volatility and risk in relation to the market, and can be used to inform investment decisions.

Lastly, the Value at Risk (VaR) for the portfolio was estimated, with IBM having the lowest contribution and Dell Technologies having the highest contribution. In terms of percentage contribution, IBM has the lowest percentage contribution with 16.43337% while Dell Technologies has the highest percentage contribution with 24.02250%. These results can assist in identifying potential risks and inform investment decisions.

Overall, these findings provide a potential investor with valuable information to choose the best efficient portfolio. The weak trade-off between risk and return suggests that the investor should be cautious and carefully weigh the potential returns against the potential risks. The Sharpe ratios, Capital Market Line, beta, and VaR can also help the investor make informed decisions about the portfolio's performance and risk management.

REFERENCES

Agmon, T. (1972). The relations among equity markets: A study of share price co-movements in the United States, United Kingdom, Germany and Japan. *The Journal of Finance, 27*(4), 839-855.

Andersen, T. G., Bollerslev, T., Christoffersen, P. F., & Diebold, F. X. (2006). Volatility and correlation forecasting. *Handbook of economic forecasting, 1*, 777-878.

Anderson, E. W., Ghysels, E., & Juergens, J. L. (2009). The impact of risk and uncertainty on expected returns. *Journal of financial economics, 94*(2), 233-263.

Black, F. (1993). Estimating expected return. *Financial analysts journal, 49*(5), 36-38.

De Loecker, J., Goldberg, P. K., Khandelwal, A. K., & Pavcnik, N. (2016). Prices, markups, and trade reform. *Econometrica, 84*(2), 445-510.

Martin, I. (2017). What is the Expected Return on the Market? *The Quarterly Journal of Economics, 132*(1), 367-433.

Morck, R., Yeung, B., & Yu, W. (2000). The information content of stock markets: why do emerging markets have synchronous stock price movements? *Journal of financial economics, 58*(1-2), 215-260.

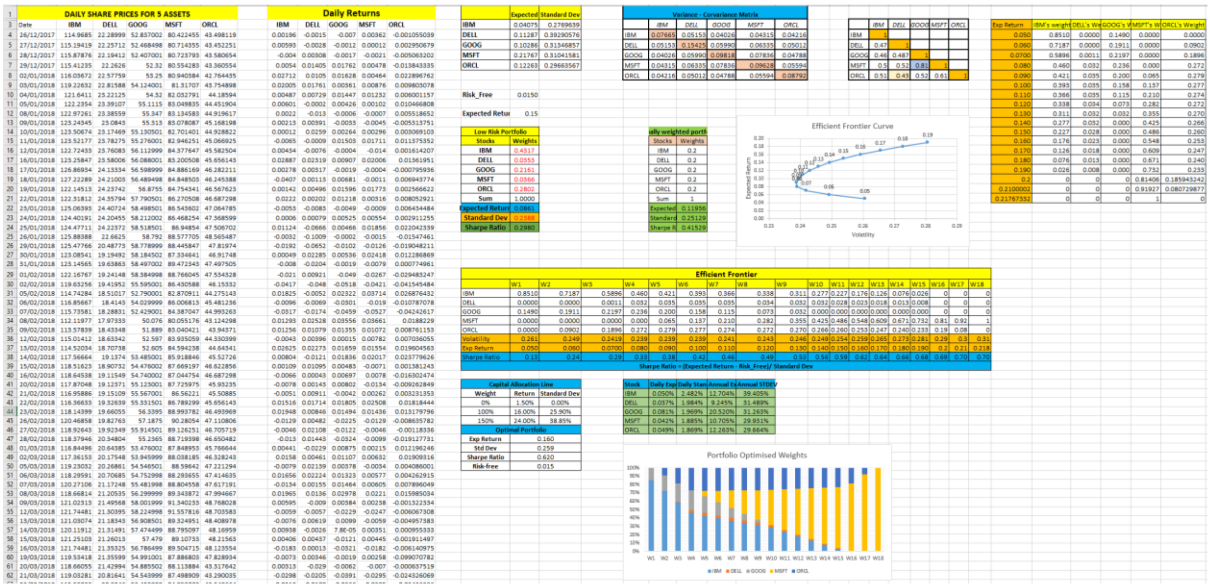
Muniesa, F. (2007). Market technologies and the pragmatics of prices. *Economy and society, 36*(3), 377-395.

Shiller, R. J. (1992). *Market volatility*: MIT press.

Vasilev, J., Turygina, V. F., Kosarev, A. I., & Nazarova, Y. Y. (2016). Mathematical optimization in environmental economics. Algorithm of gradient projection method. *International Multidisciplinary Scientific GeoConference: SGEM, 3*, 349-355.

APPENDIX

APPENDIX 1:Excel sheet for Task 1, Task 2 and Task 3:



Appendix 2:Code R for calculating Beta:

```r
# Importing libraries
library(dplyr)
library(quantmod)
library(PerformanceAnalytics)
library(imputeTS)
library(PortfolioAnalytics)
# Define the tickers and weights of the portfolio
tickers <- c("IBM", "DELL", "GOOG", "MSFT", "ORCL")
weights <- c(.2, .2, .2, .2, .2)
# Retrieve the historical prices for the tickers, the portfolioPrices object will be a matrix where the rows are
# the dates and the columns are the prices for each ticker
portfolioPrices <- NULL
for (Ticker in tickers)
  portfolioPrices <- cbind(portfolioPrices,
                           getSymbols.yahoo(Ticker, from="2017-12-23", to= "2022-12-23", periodicity = "daily",
                                            auto.assign=FALSE)[,4])
# Checking the first and last rows of the portfolioPrices matrix
head(portfolioPrices)
length(portfolioPrices)
tail(portfolioPrices)

# Retrieve the historical prices for the benchmark
benchmarkPrices <- getSymbols.yahoo("SPY",
                                    from="2017-12-23", to = "2022-12-23",
                                    periodicity = "daily",
                                    auto.assign=FALSE)[,4]

# Count the number of missing values in the benchmark prices
colSums(is.na(benchmarkPrices))

# Calculate the returns for the benchmark
# ROC function is used to calculate returns, type is set to "discrete"
ROC_benchmarkPrices <- ROC(benchmarkPrices, type = "discrete")

# Omit any rows with NA values
benchmarkReturns <- na.omit(ROC_benchmarkPrices)

#Rename Columns
colnames(portfolioPrices) <- tickers

#Get sum of NA per column
colSums(is.na(portfolioPrices))

#Plot
plot(portfolioPrices, legend = tickers)
```

```r

#Calculate Returns For DF
# ROC function is used to calculate returns, type is set to "discrete"
ROC_portfolioPrices <- ROC(portfolioPrices, type="discrete")

# Omit any rows with NA values
dailyReturns <- na.omit(ROC_portfolioPrices)

#Calculate Portfolio Returns
# Return.portfolio function is used to calculate returns, with the weights specified
portfolioReturn <- Return.portfolio(dailyReturns, weights=weights)

#Plot Performance
# chart.CumReturns function is used to plot cumulative returns
chart.CumReturns(portfolioReturn)

# charts.PerformanceSummary function is used to plot performance summary
charts.PerformanceSummary(portfolioReturn)

#Calculate Metrics with the risk-free rate =1.5% per year
# CAPM.beta function is used to calculate beta
CAPM.beta(portfolioReturn, benchmarkReturns, .015/252)

# CAPM.beta.bull function is used to calculate beta during bullish market
CAPM.beta.bull(portfolioReturn, benchmarkReturns, .015/252)

# CAPM.beta.bear function is used to calculate beta during bearish market
CAPM.beta.bear(portfolioReturn, benchmarkReturns, .015/252)

#CAPM.alpha(portfolioReturn, benchmarkReturns, .015/252)
CAPM.jensenAlpha(portfolioReturn, benchmarkReturns, .015/252)

SharpeRatio(portfolioReturn, Rf = .015/252, p = 0.95, FUN = "StdDev",
            weights = NULL, annualize = FALSE)

table.AnnualizedReturns(portfolioReturn, Rf=.015/252, geometric=TRUE)
```

Appendix 3: Code R for calculate how much each asset contributes to VaR

```r
# Load the quantmod, PerformanceAnalytics and PortfolioAnalytics libraries
library(quantmod)
library(PerformanceAnalytics)
library(PortfolioAnalytics)

# Define the tickers and weights for the portfolio
tickers <- c("IBM", "DELL", "GOOG", "MSFT", "ORCL")
weights <- c(.20, 0.20,.20,0.20,0.20)

# Create an empty matrix for portfolio prices
portfolioPrices <- NULL

# Loop through the tickers and get the closing prices for each
for (Ticker in tickers) {
  portfolioPrices <- cbind(portfolioPrices,
                           getSymbols.yahoo(Ticker, from="2017-12-23", to= "2022-12-23",
                                            periodicity = "daily", auto.assign=FALSE)[,4])
}

# Calculate the returns for the portfolio
portfolioReturns <- na.omit(ROC(portfolioPrices))

# Calculate the Value at Risk (VaR) for the portfolio using the Gaussian method, with a confidence level of 95%
PorVaR.Gaus <- VaR(portfolioReturns, p=0.05, weights=weights
                  ,portfolio_method = "component",method="gaussian")

# Print the VaR results
print(PorVaR.Gaus)

# Get the contributions to VaR for each stock
contribution <- PorVaR.Gaus$contribution

# Print the contributions
print(contribution)

# Calculate the percentage contributions to VaR for each stock
pct_contrib_VaR <- PorVaR.Gaus$pct_contrib_VaR*100

# Print the percentage contributions
print(pct_contrib_VaR)
```

```
#plot

library(ggplot2)
contribution_df <- data.frame(tickers, contribution)
colnames(contribution_df) <- c("Stock", "Contribution")
ggplot(contribution_df, aes(x = Stock, y = Contribution)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  ggtitle("VaR Contribution") + xlab("Stocks") + ylab("Contribution")

#plot percentage
pct_contrib_VaR_df = data.frame(tickers, pct_contrib_VaR)
print(pct_contrib_VaR_df)
ggplot(pct_contrib_VaR_df, aes(x = tickers, y = pct_contrib_VaR)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  geom_text(aes(label = paste0(round(Percentage_Contribution,2),"%"), y = Percentage_Contribution, x = Stock),
            position = position_stack(vjust = 0.5)) +
  ggtitle("VaR Percentage Contribution") + xlab("Stocks") + ylab("Percentage Contribution")
```

Appendix 4: Code R for using ARCH/GARCH

```
library(tseries)
library(fGarch)
library(rugarch)
library(gogarch)
library(PerformanceAnalytics)
library(zoo)
library(quantmod)
library(xts)

# Retrieve MSFT stock prices from Yahoo Finance
MSFT <-getSymbols.yahoo("MSFT", from="2017-12-23",
                              to ="2022-12-23", periodicity = "daily",
                              auto.assign=FALSE)[,4]
# Remove any missing values
na.omit(MSFT)

# Plot IBM stock prices
plot(MSFT)

# Create a zoo object for plotting
library(zoo)
MSFT<- zoo(MSFT)

# Create a new time series object for the first difference of the natural log of the closing prices
RMSFT<-ts(diff(log(MSFT$MSFT.Close)))

# Check stationary of data
adf.test(RMSFT)

# Plot the new time series
plot(RMSFT, col='BLUE')

# Fit several GARCH models to the time series
g10<-garch(RMSFT, order=c(1,0), trace=FALSE)
g11<-garch(RMSFT, order=c(1,1), trace=FALSE)
g20<-garch(RMSFT, order=c(2,0), trace=FALSE)
g21<-garch(RMSFT, order=c(2,1), trace=FALSE)
g22<-garch(RMSFT, order=c(2,2), trace=FALSE)
g02<-garch(RMSFT, order=c(0,2), trace=FALSE)
g12<-garch(RMSFT, order=c(1,2), trace=FALSE)
```

```r

# Create a list of the GARCH models
models <- list(g10, g11, g20, g21, g22, g02, g12)

# Extract the AIC values from the models
AIC_values <- sapply(models, AIC)

# Find the index of the model with the lowest AIC value
best_model_index <- which.min(AIC_values)

# Select the best model
best_model <- models[[best_model_index]]

# Print the best model's AIC value
print((best_model))

plot(g11)

#build model garch 1-1 with garchFit
m2 <- garchFit(~garch(1,1), data = RMSFT, trace = F)
summary(m2)
plot(m2)
plot(m2, which="all")
predict(m2, 5)
#build model with urgrachfit

specmodel = ugarchspec(mean.model = list(armaOrder = c(0,0)),
                        variance.model = list(model = 'iGARCH', garchOrder = c(1,1)))

m3  <- ugarchfit(data = RMSFT, spec = specmodel)
summary(m3)
plot(m3, which="all")
#forecast
ugarchforecast(m3, n.ahead = 5)
```