Exploring Value-at-Risk (VaR)

- **VALUE-AT-RISK (VaR) IN A FINANCIAL CONTEXT:**

Value at Risk is the highest anticipated loss at a specified level of confidence over a specified time period.

Value-at-risk is a statistical indicator of how risky financial institutions or asset portfolios are. It is described as the highest monetary loss that may reasonably be anticipated over a specified time horizon with a particular level of confidence. For instance, if the 99% one-month VAR is $1 million, there is 99% confidence that the portfolio won't lose more than $1 million during the course of the following month.

Since we must distinguish between normal and abnormal risk as well as market and nonmarket risk, the measure is occasionally defined more specifically as the potential loss in value from "normal market risk" as opposed to total risk.

- **METHODS THAT CAN BE IMPLEMENTED TO ESTIMATE VALUE AT RISK:**

The three primary approaches to calculate VaR are as follows:

1. HISTORICAL METHOD:

Using the previous returns of an item as a basis.

A start date and an end date are given by the investor or analyst, which prompts a number of scenarios that demonstrate the historical value that is at risk. The price of the security in this situation is a variable, along with market volatility.

This method provides a thorough view of the prospective gains and losses in the portfolio while precisely computing linear and non-linear possibilities.

2. PARAMETRIC METHOD:

The value at risk is calculated using the historical mean and standard deviation under the assumption that returns have a normal distribution.

This approach, sometimes referred to as the variance-covariance approach, makes the assumption that the returns produced by a given portfolio are normally distributed and can be fully explained by the standard deviation and expected returns.

The formula for Value at Risk: Market price times volatility equals VaR.

Using a specific confidence level, volatility is used to denote a multiple of standard deviation (SD). A 95% confidence level will therefore reveal volatility of 1.65 standard deviations.

3. MONTE CARLO METHOD:

Simulations are used in the Monte Carlo approach.

This approach makes use of a non-linear pricing model and gauges the degree of risk by projecting several future possibilities. This approach works well when there are numerous issues with risk measurement. Additionally, it offers a comprehensive and in-depth distribution of the portfolio's gains and losses, which may or may not be symmetrical. Comparatively speaking to other computation techniques, it takes more time.

- **IMPLEMENTING HISTORICAL AND PARAMETRIC METHOD IN R STUDIO TO CALCULATE VaR (VALUE AT RISK)**

STEP 1:

We will generate a list of random numbers (here we are generating daily returns of a company's stock) using R studio

Open and start R Studio

To retrieve stock price information from Yahoo Finance and calculate returns, load the quantmod library. If the package hasn't previously been installed, we'll use '**install.packages("quantmod")**' to do so before loading it with **'library(quantmod)'**.

The **'getSymbols'** function will be used to retrieve historical stock price information for the company. then modify the date range as necessary, replacing "AAPL" with the ticker symbol of the firm you are interested in.

We will use the **'dailyReturn'** function to determine the daily returns.

Then we will print the daily returns calculated.

The input (BLUE TEXT) and output (BLACK TEXT) is shown below in R Studio window:

```
Console   Terminal ×   Background Jobs ×
R 4.3.1 · ~/
> library(quantmod)
> getSymbols("AAPL", from = "2023-08-01", to = "2023-08-10")
[1] "AAPL"
> returns <- dailyReturn(AAPL)
> print(returns)
           daily.returns
2023-08-01  -0.003210379
2023-08-02  -0.015489999
2023-08-03  -0.007321652
2023-08-04  -0.048020049
2023-08-07  -0.017253691
2023-08-08   0.005311696
2023-08-09  -0.008954397
> 
```

STEP 2:

For the data mentioned earlier, we will now use R Studio to calculate the Value at Risk (VaR) using the historical technique:

A vector or data frame holding the historical returns must first be created. Since we supplied a list of returns in this instance, we can construct the following vector:

The amount of confidence we desire for our VaR estimate must then be specified. For instance, we can set the following if we want to calculate the VaR at the 95% confidence level.

Find the appropriate quantile of the historical returns and use that to calculate the VaR. The quantile function can be used for this.

Then we will print the VaR.

The historical VaR will now be calculated and displayed in R Studio at the selected confidence level (for example, 95%).

```
Console   Terminal ×   Background Jobs ×
R 4.3.1 · ~/
> library(quantmod)
> getSymbols("AAPL", from = "2023-08-01", to = "2023-08-10")
[1] "AAPL"
> returns <- dailyReturn(AAPL)
> print(returns)
           daily.returns
2023-08-01  -0.003210379
2023-08-02  -0.015489999
2023-08-03  -0.007321652
2023-08-04  -0.048020049
2023-08-07  -0.017253691
2023-08-08   0.005311696
2023-08-09  -0.008954397
> returns <- c(-0.003210379, -0.015489999, -0.007321652, -0.048020049, -0.017253691, 0.00
5311696, -0.008954397)
> confidence_level <- 0.95
> var <- quantile(returns, 1 - confidence_level)
> cat("Historical VaR at", confidence_level * 100, "% confidence level:", var, "\n")
Historical VaR at 95 % confidence level: -0.03879014
> 
```

The input (BLUE TEXT) and output (BLACK TEXT) is shown below in R Studio window:

```
> returns <- c(-0.003210379, -0.015489999, -0.007321652, -0.048020049, -0.017253691, 0.00
5311696, -0.008954397)
> confidence_level <- 0.95
> var <- quantile(returns, 1 - confidence_level)
> cat("Historical VaR at", confidence_level * 100, "% confidence level:", var, "\n")
Historical VaR at 95 % confidence level: -0.03879014
```

STEP 3:

For the data mentioned earlier, we will now use R Studio to calculate the Value at Risk (VaR) using the Parametric technique:

We must make some assumptions about the distribution of our data in order to calculate the VaR using the parametric approach. Typically, we can assume that the returns are distributed normally.

Within this code:

'returns' is a vector that includes all of the previous returns.

'confidence_level' is the desired level of confidence, for example, 0.95 for a 95%.

We compute the mean (**'mean_return'**) and standard deviation **('std_dev').**

Using the '**qnorm'** function, we determine the quantile of the standard normal distribution at the given confidence level **('z_score').**

Then, using the formula **'mean_return + z_score * std_dev**', the parametric VaR is calculated.

The parametric VaR is then printed at the chosen confidence level.

Console Terminal × Background Jobs ×

R 4.3.1 · ~/

```
> library(quantmod)
> getSymbols("AAPL", from = "2023-08-01", to = "2023-08-10")
[1] "AAPL"
> returns <- dailyReturn(AAPL)
> print(returns)
           daily.returns
2023-08-01  -0.003210379
2023-08-02  -0.015489999
2023-08-03  -0.007321652
2023-08-04  -0.048020049
2023-08-07  -0.017253691
2023-08-08   0.005311696
2023-08-09  -0.008954397
> returns <- c(-0.003210379, -0.015489999, -0.007321652, -0.048020049, -0.017253691, 0.00
5311696, -0.008954397)
> confidence_level <- 0.95
> var <- quantile(returns, 1 - confidence_level)
> cat("Historical VaR at", confidence_level * 100, "% confidence level:", var, "\n")
Historical VaR at 95 % confidence level: -0.03879014
> returns <- c(-0.003210379, -0.015489999, -0.007321652, -0.048020049, -0.017253691, 0.00
5311696, -0.008954397)
> confidence_level <- 0.95
> mean_return <- mean(returns)
> std_dev <- sd(returns)
> z_score <- qnorm(1 - confidence_level)
> var_parametric <- mean_return + z_score * std_dev
> cat("Parametric VaR at", confidence_level * 100, "% confidence level:", var_parametric,
"\n")
Parametric VaR at 95 % confidence level: -0.04147873
> 
```

The input (BLUE TEXT) and output (BLACK TEXT) is shown below in R Studio window:

```
> returns <- c(-0.003210379, -0.015489999, -0.007321652, -0.048020049, -0.017253691, 0.00
5311696, -0.008954397)
> confidence_level <- 0.95
> mean_return <- mean(returns)
> std_dev <- sd(returns)
> z_score <- qnorm(1 - confidence_level)
> var_parametric <- mean_return + z_score * std_dev
> cat("Parametric VaR at", confidence_level * 100, "% confidence level:", var_parametric,
"\n")
Parametric VaR at 95 % confidence level: -0.04147873
> 
```

- **CONDITIONAL VALUE-AT-RISK (CVaR) or EXPECTED SHORTFALL:**

A risk measure known as Conditional Value at Risk (CVaR), often referred to as Expected Shortfall (ES) or Tail Value at Risk (TVaR), estimates the expected loss of an investment or portfolio in the case of adversity in the market.

Conditional Value-at-Risk calculates the overall losses that VaR does not take into account. VaR must first be calculated in order to calculate ES. When the loss is more than the VaR, ES is the projected loss during time T.

In order to estimate the expected loss in the tail of the loss distribution, CVaR is calculated as the average of the losses that exceed the VaR threshold.

When determining the risk of assets with abnormal return distributions, such as those with fat tails or skewness, CVaR is particularly helpful. By taking into account both the possibility and magnitude of extreme losses, it offers a more complete understanding of tail risk.

Additionally, risk management and portfolio optimization both heavily rely on CVaR. It aids financial institutions and investors in better understanding possible losses during volatile market conditions, allocating capital effectively, and developing risk management plans.

Several financial applications, such as portfolio optimization, stress testing, risk budgeting, and risk monitoring and reporting, frequently use CVaR.

- **ESTIMATING CONDITONAL VALUE AT RISK (CVaR):**

There are various ways to calculate CVaR, and the method chosen is frequently determined by the data that is available and the distributional assumptions made. Here are some popular techniques for calculating CVaR:

Historical Approach: The historical method involves estimating CVaR based on past data. You get the average of the worst returns above a given quantile by sorting previous returns in descending order. By averaging the bottom 5% of prior returns, for instance, you may determine the CVaR at a 95% confidence level.

Parametric Approach: The parametric approach makes the assumption that returns adhere to a predetermined distribution, frequently the normal distribution. You construct the quantile of the distribution at the specified confidence level after estimating the mean and standard deviation of returns. The anticipated shortfall above that quantile is considered to represent the CVaR.

Simulation using Monte Carlo: Monte Carlo simulation entails creating a variety of return scenarios depending on a selected model or distribution. The CVaR is calculated by simulating a large number of potential return paths, figuring out the losses for each path, and then averaging the worst losses over the selected confidence level.

Stress testing: Stress testing entails simulating several extreme scenarios, such as market collapses or economic downturns, on the portfolio or investment. Based on the results of these stress tests, you examine how the portfolio performs in certain situations and estimate the CVaR.

Non-Parametric Approaches: Non-parametric approaches do not rely on a particular return distribution assumption. They rely on empirical data instead. Kernel density estimation is a popular non-parametric technique that involves directly estimating the probability density function of returns using historical data and then computing CVaR based on this estimated distribution.

Analytical Models: Analytical models may be used to calculate CVaR for particular asset classes or risk characteristics. These models may include variables like volatility and can be tailored to particular classes of financial instruments.

- **Calculation of CVaR:**

Discrete Distribution: CVaR is determined as the weighted average of the losses over the chosen confidence level when the probability distribution is discrete.

Continuous Distribution: In the case of a continuous probability distribution, CVaR is calculated as the sum of the losses times the corresponding probabilities, over the specified degree of confidence.

Numerical Methods (Monte Carlo Simulation): By modelling a large number of alternative market situations and estimating the average loss in the worst-case scenarios, Monte Carlo simulation can be used to estimate CVaR for complicated distributions or big portfolios.

- **ADVANTAGES OF CVaR OVER VaR:**

- VaR does not regulate scenarios beyond VaR
- CVaR accounts for losses above VaR
- Deviation and Risk are different risk management concepts
- VaR has superior mathematical features than CVaR
- Risk management using CVaR functions can be done extremely effectively
- The Standard Deviation faces stiff competition from CVaR Deviation.

- The CVaR explicitly tackles tail-end risks by focusing on the average loss magnitude beyond the VaR threshold, whereas VaR measures the downside risk up to a certain point but does not inherently account for the severity of losses beyond the threshold.

- In contrast to CVaR, which extends VaR by computing the average of losses above the VaR threshold, VaR is computed by finding the loss value at a given confidence level, frequently using statistical techniques like percentiles or simulation approaches.

- Although VaR is sensitive to changes in the distribution of returns close to the threshold level, it is less effective at capturing the behavior of severe losses than CVaR, which is especially pertinent when reducing extreme losses is a top concern.

- **IMPLEMENTING A CVaR CALCULATION IN R:**

Historical method has been used to calculate CVaR in R Studio.

We must assume some things about the distribution of our data in order to calculate Conditional Value at Risk (CVaR) using the parametric technique in R Studio, such as assuming a normal distribution.

In the following code:

'returns' is a vector in this code that holds the past returns.

'**confidence_level'** the desired level of confidence, for example, 0.95 for a 95%

We compute the mean average (**'mean_return'**) and standard deviation (**'std_dev'**).

Using the **'qnorm'** function, we determine the quantile of the standard normal distribution at the given confidence level **('z_score').**

'-mean_return - (z_score * std_dev)' is then used to generate the parametric CVaR.

The parametric CVaR is then printed at the chosen confidence level.

The input (BLUE TEXT) and output (BLACK TEXT) is shown below in R Studio window:

```
> returns <- c(-0.003210379, -0.015489999, -0.007321652, -0.048020049, -0.017253691, 0.
005311696, -0.008954397)
>
> confidence_level <- 0.95
> mean_return <- mean(returns)
> std_dev <- sd(returns)
> z_score <- qnorm(1 - confidence_level)
> cvar_parametric <- -mean_return - (z_score * std_dev)
>
> cat("Parametric CVaR at", confidence_level * 100, "% confidence level:", cvar_paramet
ric, "\n")
Parametric CVaR at 95 % confidence level: 0.04147873
> 
```

Environment | History | Connections | Tutorial

Import Dataset | 50 MiB | List

R | Global Environment

Data	
AAPL	An xts object on 2023-08-01 / 2023-08-09…
Values	
confidence_level	0.95
cvar_parametric	0.041478727829935
mean_return	-0.0135626387142857
returns	num [1:7] -0.00321 -0.01549 -0.00732 -0.04…
std_dev	0.0169717770981168
var	Named num -0.0388
var_parametric	-0.041478727829935
z_score	-1.64485362695147

Exploratory Analysis:

I ran the following code for the exploratory analysis of the data:

robo <- read.csv("ROBO2 (3).csv")

str(robo)

summary(robo)

- From 2017 to 2022, the ROBO ETF's daily prices, returns, and trading volumes are included in the data. The daily closing price will then be plotted to show the overall trend.

I ran the following code afterwards:

plot(robo$price, type="I", xlab="Date, ylab="price (GBP)", main="ROBO Daily Closing Price")

- Over the time period, the price exhibits an overall rising trend. Specifically in early 2018, early 2020 during the COVID pandemic, and late 2022, there are times of instability and drawdowns.

Then I ran the following code for further analysis:

hist(robo$today, breas=40, main="Distribution of Daily Returns of prices", xlab="Returns")

- The returns are centered about 0, as is customary for financial return data, and have broader tails.

The estimations of VaR and CVaR change significantly over time. They frequently rise during times of market turbulence and declines, signaling higher anticipated tail risk. However, the actual returns frequently surpass the VaR levels, suggesting violations; they don't seem to anticipate future returns consistently.

This indicates that VaR may have limits as a practical risk management tool. In genuine financial time series like this one, the assumptions of normally distributed returns and static risk levels are frequently broken. Although CVaR offers more details about severity, it continues to rely on previous data, which may not accurately forecast future risk. Making trading and investment decisions may need the use of supplemental or alternative risk indicators. Risk forecasting could benefit from models that are more flexible and futuristic.

In conclusion, VaR and CVaR offer some understanding of historical tail risk but seem to have limits in terms of forecasting for this ETF. Due to the fluctuation, it is difficult to gauge financial risk using only historical return distributions.