VaR Approaches

Historical

```
In [1]:   # Importing libraries

          import numpy as np
          import pandas as pd
          import yfinance as yf

          from tabulate import tabulate

          from scipy.stats import norm

          import warnings
          warnings.filterwarnings("ignore")
```

```
In [2]:   # Getting Historical data of Nifty from Yahoo Finance

          df = yf.download('^NSEI', '2021-01-01', '2023-08-14')
          df = df[['Close']]
          df['returns'] = df.Close.pct_change()
```
```
[**********************100%%**********************]  1 of 1 completed
```

```
In [3]:   Historical_VaR_99 = df['returns'].quantile(0.01)
          print(Historical_VaR_99)
```
```
-0.02646235627345307
```

Variance Covariance

```
In [4]:   mean = np.mean(df['returns'])
          std_dev = np.std(df['returns'])
```

```
In [5]:   VarianceCovariance_VaR_99 = norm.ppf(0.01, mean, std_dev)
          print(VarianceCovariance_VaR_99)
```
```
-0.02172905748881931
```

Montecarlo Simulation

```
In [6]:   np.random.seed(42)
          n_sims = 1000000
```

```
In [7]:   simulated_returns = np.random.normal(mean, std_dev,n_sims)
          Simulated_VaR_99 = np.percentile(simulated_returns,1)
          print(Simulated_VaR_99)
```
```
-0.02176656965405816
```

Loading [MathJax]/extensions/Safe.js