FINANCIAL ANALYTICS

Course Project

Submitted To

Dr. Vivek Vijay

Associate Professor

Indian Institute of Technology Jodhpur

Submitted By

Sai Swaroop Kakarla (G24AI1026)

Purushothaman S (G24AI1042)

Mugundan B (G24AI1051)

Sri Charan Reddy (G24AI1095)

Aakanksha Nalamati (G24AI1111)

Pranav Srivastav Tenali Gnana (G24AI1114)

Indian Institute of Technology Jodhpur

Date: 25th June 2025

Index

Introduction

This project performs a comprehensive financial analysis of 10 selected assets from the stock market using Python. The analysis covers three years of historical price data, computes descriptive statistics, visualizes price trends and returns, performs correlation analysis, and constructs an optimal portfolio based on Modern Portfolio Theory (Markowitz Efficient Frontier). The objective is to understand the risk-return characteristics of the chosen assets and develop a diversified investment index that maximizes return for a given level of risk.

Portfolio Analysis

Portfolio analysis is the process of evaluating and assessing a collection of investments (a *portfolio*) to understand its overall performance, risk profile, and potential returns. In essence, it involves examining how the different assets in a portfolio (such as stocks, bonds, real estate, etc.) are combined and how that mix affects the portfolio's return and volatility. The goal is often to determine whether the portfolio is aligned with the investor's objectives and risk tolerance, and to identify ways to improve its risk-adjusted performance. By analysing a portfolio's composition and historical performance, investors can gauge whether they are on track to meet their financial goals or if adjustments (like rebalancing asset allocations or replacing underperforming assets) are needed

Practical Uses in Personal Investing

For individual investors, portfolio analysis translates theory into the practical management of one's savings and investments. One major application is **diversification for risk management**. The classic adage "Don't put all your eggs in one basket" captures this idea. By analysing their portfolio, personal investors ensure they hold a variety of asset classes (stocks, bonds, cash, perhaps real estate, or others) so that no single investment can make or break the entire portfolio. Diversification helps *reduce the volatility and potential losses* in a portfolio more than it necessarily boosts returns. For example, instead of trying to pick one winning stock (and risking picking a loser), an investor might hold dozens of stocks or use broad index funds. History is full of stories of people striking it rich on a single stock pick, but for every big winner there are many investments that flop – thus spreading money across many securities is crucial to avoid catastrophic loss if one investment fails. Through portfolio analysis, an individual can assess whether their asset mix matches their risk tolerance and goals: e.g. a young investor might analyse and find they can take more equity exposure for growth, whereas a retiree's analysis might prompt shift to more bonds for stability. Regular analysis also informs **rebalancing** (adjusting the weights of assets back to target levels) – for instance, if stock markets have a big run-up and a portfolio becomes stock-heavy relative to the investor's plan, portfolio analysis would flag this increased risk, indicating it may be time to sell some stocks and buy other assets to restore the intended risk level. Overall, in personal investing, portfolio analysis is used to achieve a well-balanced portfolio that harnesses diversification to protect against undue risk while aiming for reasonable returns aligned with one's financial objectives.

Practical Uses in Institutional Finance (Pension Funds and Endowments)

In the realm of **corporate and institutional finance**, portfolio analysis is equally – if not more indispensable. Large institutional investors such as pension funds, insurance companies, and university endowments manage billions of dollars with the dual mandate of growth and safety, and they rely on rigorous portfolio analysis to fulfil their obligations. A pension fund, for example, must ensure that its investment portfolio can generate sufficient returns to pay retirees their benefits in the future. This leads to sophisticated asset-liability management and portfolio analysis: institutions will set an asset allocation policy (spread of stocks, bonds, real assets, alternatives, etc.) and continually analyse performance, risks, and changing market conditions. **Diversification** and risk management are paramount for institutions – they often diversify not only across traditional assets but also into alternatives like private equity, hedge funds, real estate, or infrastructure. In recent years, many pension plans have actually **expanded into alternative investments to boost returns and reduce overall risk**, especially in the face of low interest rates and volatile equity markets The 2008–2009 financial crisis and the ensuing low-yield environment, for instance, prompted numerous funds to rethink the classic heavy reliance on stocks and bonds By conducting detailed portfolio analyses (including stress tests and scenario analyses), institutional investors can find an optimal balance that aims to meet long-term return targets while controlling risk to an acceptable level.

Real-world examples illustrate these practices well. Historically, many institutions followed a simple "60/40" portfolio model (60% equities, 40% bonds) as a balanced strategy. However, some forward-thinking endowments and funds used analysis to pursue better diversification. **Yale University's endowment**, under the leadership of David Swensen, famously applied MPT principles to revolutionize its portfolio: instead of sticking to the 60/40 mix, Yale's portfolio was diversified across domestic and international stocks, bonds, plus a substantial allocation to alternative assets like hedge funds, private equity, real estate, and even timber. By taking advantage of Yale's long-time horizon and emphasizing broad diversification (in what became known as the "Yale Model"), Swensen was able to greatly improve the endowment's risk-adjusted returns. In fact, Yale's endowment performance outpaced the traditional 60/40 portfolio by about 4% per year under his tenure, demonstrating the tangible benefits of robust portfolio analysis and diversification. Likewise, large pension funds (e.g. CalPERS in the U.S. or major Canadian plans) regularly perform portfolio analysis to adjust their allocations in response to economic changes – for example, increasing allocations to infrastructure or private credit when analysis shows these can improve returns or provide better hedges against inflation. In sum, institutional portfolio analysis is about ensuring that a fund's investment strategy can meet its specific goals (such as paying future liabilities or funding a university's budget) with an acceptable level of risk. This involves continuous monitoring and tweaking of the portfolio using quantitative models, risk metrics, and performance benchmarks to guide decisions like hiring asset managers, shifting allocations, or hedging certain risks.

Significance and Benefits of Portfolio Analysis

Whether for an individual investor or a large institution, conducting portfolio analysis is highly beneficial for informed decision-making. First and foremost, it imposes a **disciplined, strategic approach** to investing rather than ad-hoc choices. By systematically reviewing how a portfolio is performing relative to its goals and risk parameters, investors can *identify imbalances or problems early* and take corrective action. For example, analysis might reveal that a portfolio's strong recent returns came with an unacceptably high concentration in a single sector – prompting diversification before a downturn hits that sector. Portfolio analysis also emphasizes **risk-adjusted performance** – not just looking at returns in a vacuum, but considering the amount of risk taken to achieve those returns (through measures like the Sharpe ratio or alpha). This helps in comparing a portfolio against benchmarks and understanding if the returns are adequate for the risk incurred. Another key benefit is aligning investments with objectives: over time an investor's goals or circumstances may change (say, approaching retirement or, for a foundation, needing more cash flows), and ongoing analysis ensures the portfolio's asset mix is realigned accordingly. Broadly, portfolio analysis aids in **optimizing resource allocation, improving risk management, and guiding strategic investment decisions**. It brings to light the trade-offs between different investment choices (for instance, how adding a new asset class might lower overall volatility) and thus supports more rational, evidence-based decisions rather than emotional reactions to market swings. Ultimately, the significance of portfolio analysis lies in how it enables investors to make *informed adjustments* to their portfolios – enhancing the likelihood of achieving long-term financial goals while avoiding unnecessary risks. In a world of uncertain markets, having this analytical insight is invaluable: it provides clarity on where you stand and confidence in how to proceed, making portfolio analysis an indispensable part of prudent investment management.

Script execution and Description Explanation Cell wise

Cell 1: Title

```
# Portfolio Analysis - Python Notebook
```

Adds a title and short description to the notebook.

Cell 2: Install Libraries

```
!pip install yfinance pandas numpy matplotlib seaborn scipy PyPortfolioOpt
```

Installs all required libraries for data fetching, analysis, visualization, and portfolio optimization.

The second cell ensures that all necessary Python packages are installed. These include libraries for financial data fetching (yfinance), numerical computations (NumPy and pandas), plotting (matplotlib and seaborn), and portfolio optimization (PyPortfolioOpt). This step is essential for running the notebook in a fresh environment using Google Colab.

Cell 3: Import Libraries

```
import yfinance as yf
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from pypfopt import EfficientFrontier, risk_models, expected_returns
from datetime import datetime, timedelta
```

The third cell imports all the essential libraries required for the analysis. These include standard data manipulation and visualization packages, as well as **EfficientFrontier**, **expected_returns**, and **risk_models** from **PyPortfolioOpt**, which are used for constructing and **analyzing** the efficient frontier and optimizing the portfolio

Cell 4: Define Asset List

```
assets = ['AAPL', 'MSFT', 'GOOGL', 'AMZN', 'TSLA', 'META', 'NVDA', 'JPM', 'V', 'JNJ']
end_date = datetime.today()
start_date = end_date - timedelta(days=3*365)
```

In the fourth cell, the user defines the list of 10 assets to analyse. These tickers are large-cap U.S. equities (e.g., AAPL, MSFT, GOOGL), but can be customized as per user preference. The cell also calculates the start and end dates for data collection, spanning the most recent three years using datetime and time delta from Python’s standard library.

Cell 5: Download Price Data

```
data = yf.download(assets, start=start_date, end=end_date, auto_adjust=False)[' Close']
data.head()
```

The fifth cell fetches the historical price data for the selected assets using yfinance. download. Here, auto_adjust=False ensures that the "Adjusted Close" column is retained, which is important for calculating accurate historical returns accounting for dividends and splits. The result is a dataframe of adjusted closing prices which is previewed using. head() to confirm the structure of the data.

Getting Historical Data

```
[ ] data = yf.download(assets, start=start_date, end=end_date)['Close']
    data.head()
```

```
/tmp/ipython-input-5-2178217691.py:1: FutureWarning: YF.download() has changed argument auto_adjust default to True
  data = yf.download(assets, start=start_date, end=end_date)['Close']
[*********************100%***********************]  10 of 10 completed
```

Ticker	AAPL	AMZN	GOOGL	JNJ	JPM	META	MSFT	NVDA	TSLA	V
Date										
2022-06-22	133.212845	108.949997	110.822861	160.501617	106.114517	154.999664	246.867920	16.336439	236.086670	189.438141
2022-06-23	136.086731	112.440002	111.572853	164.081741	104.953674	157.883835	252.456161	16.201632	235.070007	192.194412
2022-06-24	139.423172	116.459999	117.271683	166.483704	108.086075	169.231583	261.077515	17.101334	245.706665	200.863861
2022-06-27	139.423172	113.220001	115.142952	166.328415	107.220062	168.565231	258.337036	16.844700	244.919998	198.957932
2022-06-28	135.269821	107.400002	111.339752	161.597595	106.704132	159.803314	250.135040	15.958980	232.663330	193.640945

Cell 6: Calculate Descriptive Statistics

```
returns = data.pct_change().dropna()
returns.describe()
```

Next, in the sixth cell, the notebook calculates the daily returns for each asset using the .pct_change() function and removes any resulting NaN values with. dropna(). This transformed dataset is then summarized using. describe (), which computes descriptive statistics including mean, standard deviation, min, max, and quartiles. This summary gives a quick insight into the distribution and volatility of each asset's returns.

Descriptive Analytics and Statistics and Calculation for Returns

```
[ ] returns = data.pct_change().dropna()
    returns.describe()
```

Ticker	AAPL	AMZN	GOOGL	JNJ	JPM	META	MSFT	NVDA	TSLA	V
count	751.000000	751.000000	751.000000	751.000000	751.000000	751.000000	751.000000	751.000000	751.000000	751.000000
mean	0.000706	0.001118	0.000752	-0.000034	0.001387	0.002360	0.001018	0.003477	0.001180	0.000853
std	0.017850	0.022270	0.020422	0.010758	0.015427	0.027818	0.016733	0.034200	0.039340	0.012619
min	-0.092456	-0.089791	-0.095094	-0.075916	-0.074838	-0.245571	-0.077156	-0.169682	-0.154262	-0.077374
25%	-0.008047	-0.011033	-0.010759	-0.005296	-0.006376	-0.010677	-0.007712	-0.016025	-0.020546	-0.005641
50%	0.001187	0.000344	0.001052	0.000340	0.001710	0.001372	0.001143	0.003503	0.000964	0.001602
75%	0.009581	0.013759	0.011570	0.005704	0.008945	0.014330	0.009800	0.022596	0.021612	0.007308
max	0.153288	0.121778	0.102244	0.060728	0.115445	0.232824	0.101337	0.243696	0.226900	0.078373

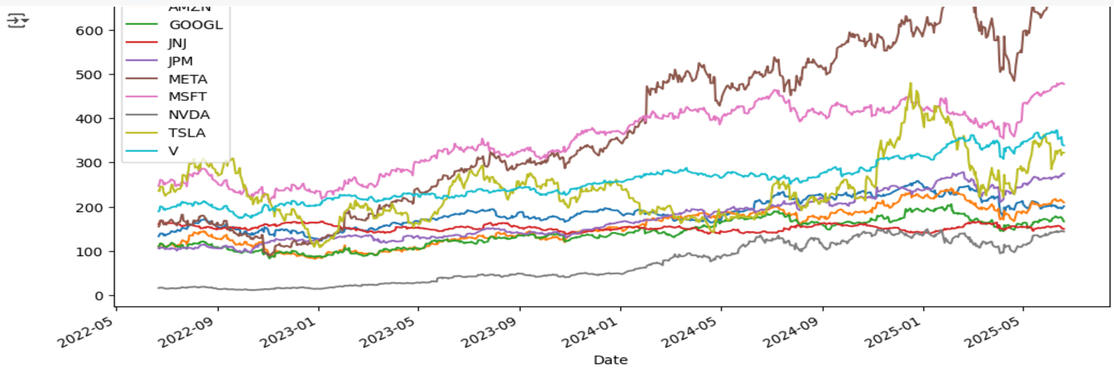
Cell 7: Time Series and Histogram Visuals

```
data.plot(figsize=(12, 6), title='Asset Prices Over Time')
plt.show()
returns.hist(bins=50, figsize=(12, 10))
plt.suptitle('Histogram of Asset Returns')
plt.show()
```

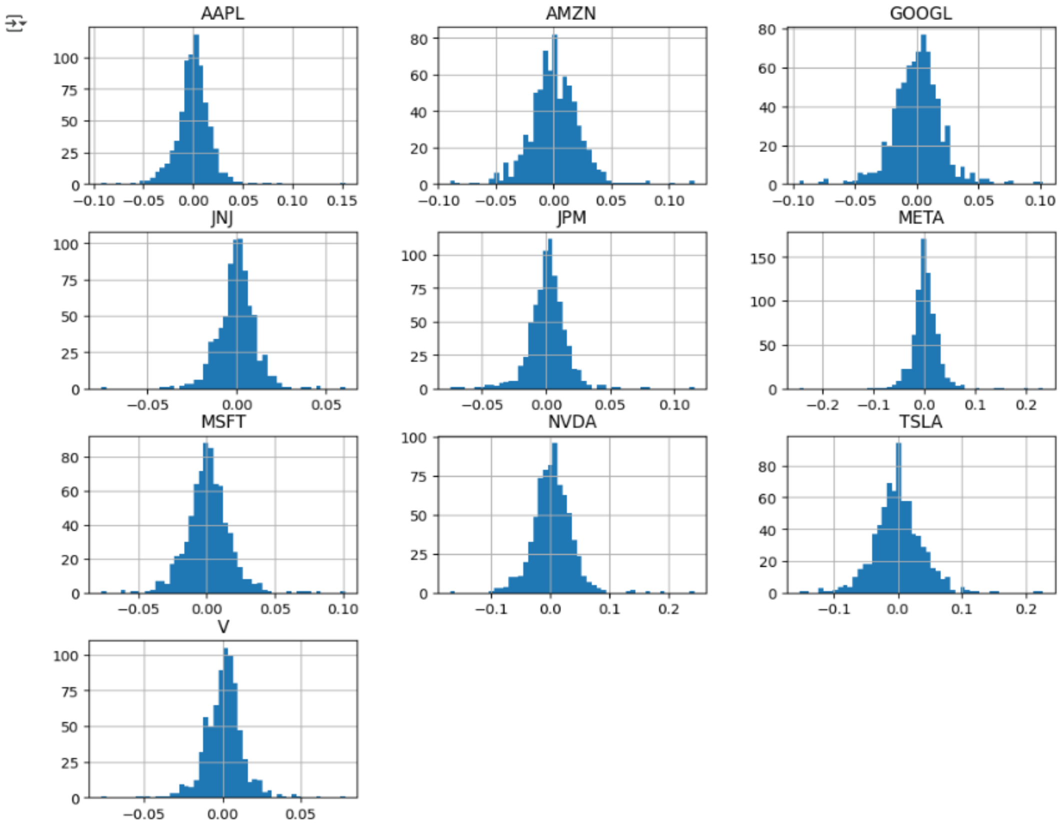
The seventh cell produces two important visualizations. The first is a time series plot of the adjusted closing prices for each asset, which helps visualize their trends and performance over time. The second is a set of histograms showing the distribution of returns for each asset. These histograms allow us to visually assess the spread and skewness of returns, helping identify which assets have more stable or volatile return distributions.

Data Visualizations and Time Series Analysis & Histograms based Data Visualization

```
[ ] data.plot(figsize=(12, 6), title='Asset Prices Over Time')
    plt.show()

    returns.hist(bins=50, figsize=(12, 10))
    plt.suptitle('Histogram of Asset Returns')
    plt.show()
```



Histogram of Asset Returns



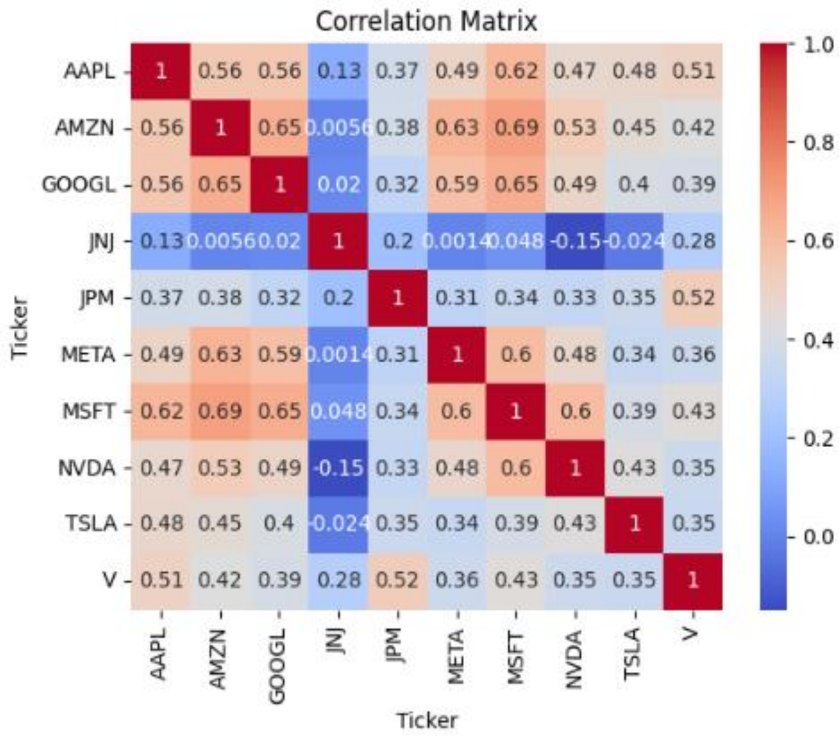
Cell 8: Correlation Matrix

```
corr_matrix = returns.corr()

sns.heatmap(corr_matrix, annot=True, cmap='coolwarm')

plt.title('Correlation Matrix')

plt.show()
```

In the eighth cell, a correlation matrix is computed using. corr() to analyse how the returns of each asset relate to one another. A heatmap is then generated using seaborn. Heatmap, which visually depicts the strength and direction of the correlations. This step is crucial for identifying diversification opportunities, as lower or negative correlations between assets can reduce overall portfolio risk.

Correlation Analysis

```
corr_matrix = returns.corr()
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```



Cell 9: Portfolio Optimization & Efficient Frontier

```
mu = expected_returns.mean_historical_return(data)

S = risk_models.sample_cov(data)

ef = EfficientFrontier(mu, S)

weights = ef.max_sharpe()

cleaned_weights = ef.clean_weights()

print(cleaned_weights)

ef.portfolio_performance(verbose=True)
```

Finally, in the ninth cell, the notebook applies **Modern Portfolio Theory** to construct an optimized portfolio. The expected returns are estimated using the mean_historical_return() function, and the sample covariance matrix is computed with sample_cov(). These inputs are passed to the Efficient Frontier class to find the asset weights that maximize the Sharpe ratio, effectively identifying the most efficient portfolio in terms of return per unit risk. The cleaned weights are printed to show the proportion of each asset in the optimal portfolio. The portfolio_performance() method then outputs the expected return, volatility (standard deviation), and Sharpe ratio of the optimized portfolio, summarizing its effectiveness.

Portfolio Optimization and Efficient Frontier

```
mu = expected_returns.mean_historical_return(data)
S = risk_models.sample_cov(data)

ef = EfficientFrontier(mu, S)
weights = ef.max_sharpe()
cleaned_weights = ef.clean_weights()
print(cleaned_weights)

ef.portfolio_performance(verbose=True)
```

```
OrderedDict([('AAPL', 0.0), ('AMZN', 0.0), ('GOOGL', 0.0), ('JNJ', 0.0), ('JPM', 0.49801), ('META', 0.14614), ('MSFT', 0.0), ('NVDA', 0.35585), ('TSLA', 0.0), ('V', 0.0)])
Expected annual return: 66.4%
Annual volatility: 29.8%
Sharpe Ratio: 2.23
(np.float64(0.6641856437834732),
 np.float64(0.2975372544605524),
 np.float64(2.2322772487353553))
```

Conclusion

This analysis explored the performance and risk characteristics of a diversified portfolio made from 10 market assets over the past three years. By analysing time series trends, statistical distributions, and correlations, we gained insights into individual asset behaviours. Using the Markowitz Efficient Frontier, we constructed a portfolio optimized for the highest Sharpe Ratio, demonstrating how quantitative finance tools can enhance investment decision-making. This model can be extended further by incorporating risk-free assets, calculating beta/alpha for CAPM, and visualizing the Capital Market Line and Security Market Line.

Git Link: https://github.com/PurushothamanShanmugam/Financial-Analytics-Project