Financial Portfolio Optimization

Team Name: Optiverse
Team Members: Aprajita Agarwal (BT2024258), Dikshita Mustoori (BT2024256)

Problem Statement

Investors want high returns with low risk but every asset behaves differently so choosing the right mix is hard. The Markowitz Mean–Variance framework models a portfolio using its expected return and risk. Expected return reflects average profit, while variance measures fluctuations. Markowitz showed that the best portfolios lie on the efficient frontier. Our goal is to use optimization to construct a portfolio that achieves a target return with the lowest possible risk.

Deriving the Optimization Formula

Step 1: Let w_i be the fraction of total capital invested in asset i.

$$\sum_{i=1}^{n} w_i = 1, \qquad w_i \geq 0.$$

Step 2: Each asset has an expected return μ_i and desired target return is R_{target}.

$$\mu^\top w \geq R_{\text{target}}.$$

Step 3: Risk is quantified using the covariance matrix Σ. We aim to minimize risk.

$$\min_{w} \; w^\top \Sigma w.$$

Final convex optimization model:

$$\begin{aligned} \min_{w} \quad & w^\top \Sigma w \\ \text{s.t.} \quad & \mu^\top w \geq R_{\text{target}}, \\ & \sum_{i=1}^{n} w_i = 1, \\ & w_i \geq 0 \quad \forall i. \end{aligned}$$

Proposed Approach

Step 1: Data Collection. We will download historical stock prices using Python's `yfinance` library. From these prices, we compute daily returns and estimate the expected returns vector μ and covariance matrix Σ.

Step 2: Model Construction. Using the Markowitz framework, we build the convex optimization model described above.

Step 3: Implementation. The optimization model will be modeled in Python using `CVXPY` and solved using solvers such as ECOS, OSQP, or Gurobi. We compute gradients, the Hessian, Lagrangian and KKT conditions to understand optimality.

Step 4: Evaluation and Visualization. By varying the target return, we compute the efficient frontier. We evaluate the resulting portfolios based on expected return, volatility, Sharpe ratio, and allocation distribution. Visualizations such as efficient frontier plots and portfolio composition graphs will be made.

FINANCIAL PORTFOLIO OPTIMIZATION

DATA COLLECTION

We gather historical stock prices using Python’s yfinance library and convert them into daily returns.From these returns, we compute average returns and the covariance matrix for modeling.

OPTIMIZATION MODEL SETUP

Using Markowitz Mean–Variance theory, we frame the problem as minimizing risk for a target return.
We build the mathematical model using returns, covariance, and portfolio weight constraints.

IMPLEMENTATION

We implement the model in Python using CVXPY, define the objective and constraints, and let a solver compute optimal weights.

EVALUATION & VISUALIZATION

We check how the optimized portfolio performs by plotting risk vs. return and generating the efficient frontier.
Graphs such as weight bars, risk-return plots, and backtesting curves help interpret results.