Final Report
Optimization in Portfolio Management

Xilin Chen, Yuan Huang

Abstract

In this research project, we focus on the Quadratic Programming (QP) forms of Markowitz portfolio optimization, and try to apply Lagrangian Method and Interior Point Method (through cvxopt) to find the optimal solutions from Markowitz (1952) Portfolio (QP Problem). On the basis of the Markowitz portfolio optimizations, we introduce the investment limitations from the real investment world, and evaluate their impact on the weighting scheme, and finally depict a case of minor impact and a case of major impact due to the addition of real-life constraint(s).

Introduction

Background

Portfolio construction is an extremely important aspect of investment management. Even if an investor chooses a group of assets or return streams to invest in, it is not a simple task to decide how much money to allocate to each asset or return stream. The expected return on the asset is certainly an important factor, but investors may also want to consider the co-dependence of investment risk and asset returns. The basic intuition is to involve a set of assets that attempt to determine the proportion of investment in each in order to minimize investment risk and maximize return on investment. However, higher return always yields higher risk.
Modern portfolio theory, introduced by Markowitz (1952), presents a mathematical framework for maximizing the expected return of a portfolio subject to a risk constraint which the risk is measured by the covariance matrix of asset returns. Specifically, the key insight is that by combining assets with different expected returns and volatility, one can decide on a mathematically optimal allocation which minimises the risk for a target return and the set of all such optimal portfolios is referred to as the efficient frontier. Although optimization problems are generally difficult, many portfolio optimization tasks can be reduced to convex optimization problems, which require the use of extensive theory and some efficient solution procedures (Boyd and Vandenberghe 2004).

Justification of Research Topic

One drawback of solving the problem of Markowitz optimization and the determination of the effective investment boundary is that when mathematical models are solvable when the number of marketable assets and market constraints are low. However, when the conditions and given the limitations of the real world, it will be a complex and difficult task. Over the years, more advanced mathematical knowledge and computerized machine learning, deep learning, etc. have been applied to solve such complex problems, thus enabling more help in solving problems that overcome the uncertainty and ambiguity of the environment.

Statement of Optimization Problem

Statement of Objective

Consider a portfolio, in which:

- the vector of weights for each asset: $\mathbf{w} = [w_1, w_2, ..., w_n]$;
- the vector of average daily return are $\boldsymbol{\mu} = [\mu_1, \mu_2, ..., \mu_n]$.

In Markowitz's portfolio analysis, **Return maximization** and **Risk minimization** are the key features:

Return: The average return across the assets is the average of the expected returns of each of the asset (n assets in total) in the portfolio

$$\tau = E[E(R_i)] = \frac{1}{n}\sum_{i=1}^{n} E(R_i) = \frac{1}{n}\sum_{i=1}^{n} \mu_i$$

Expected return of the portfolio is the weighted average of the expected returns of each of the asset in the portfolio (weighted by the weight of each asset in the portfolio):

$$E[R(\mathbf{w})] = \sum_{i=1}^{n} E(R_i) \cdot w_i = \boldsymbol{\mu}^T \mathbf{w}$$

for i = 1,2,3,...,n, where R_i represents the return of each asset.

Risk: The risk of a portfolio can be represents as the variance of the portfolio return. In the Markowitz's theory, the assets are weighted and their returns are correlated (as in the real market), and thus the calculation of the portfolio return variance takes into account the two-by-two covariance of the assets and the weights:

$$V(\mathbf{w}) = \mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w}$$

where $\boldsymbol{\Sigma}$ is the covariance matrix of the returns of the two two assets, in which the element $\sigma_{i,j} = Cov(Y_i, Y_j)$ for i, j $\in$ {1,2,3,...,n}, and Σ must be positive semi-definite.

To choose the best portfolio from a number of possible portfolios, each with different return and risk, two separate decisions are to be made, first, we need to determine a set of efficient portfolios and second select the best portfolio out of the efficient set. And we will further explain it in the solution algorithm sector.

Statement of Assumptions

There are several assumptions in the Markowitz portfolio optimization theory:

- The risk of a portfolio is based on the variability of returns from said portfolio.
- An portfolio analysis is based on the single period model of investment.
- Investors are rational in nature.
- Investors are risk averse.
- Investors prefer to increase consumption.

- The investor's utility function is concave and increasing, due to their risk aversion and consumption preference.

- Investors either maximize their portfolio return for a given level of risk or minimize their risk for a given return.

Model formulation

Classical Model: Convex Problem

The following problem A,B,and C are equivalent, but optimize the same question from different perspectives.

Problem A: Minimum Variance Portfolio (MVP)

First, consider the objective function is to minimize the risk, then the total return could be view as a constraint on weights. So the description of the optimization problem A is as follows:

$$\begin{aligned} \min \quad & \mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w} && (1) \\ \text{s.t.} \quad & \sum_i \mathbf{w}_i = 1 && (2) \\ & \boldsymbol{\mu}^T \mathbf{w} \geq \alpha && (3) \end{aligned}$$

where the first constraints asserts that the sum of the weights need to be 1 (fully invested) and the optimal portfolio return should be no less than α, which is determined under different situations.

Problem B: Maximum Return Portfolio (MRP)

The Markowitz portfolio problem could also describe as when take the risk as constraints (no more than β, also could be determined under different situations) and trying to maximize the return. The description of the optimization problem B is as follows:

$$\begin{aligned} \max \quad & \boldsymbol{\mu}^T \mathbf{w} && (4) \\ \text{s.t.} \quad & \sum_i \mathbf{w}_i = 1 && (5) \\ & \mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w} \leq \beta && (6) \end{aligned}$$

Problem C: Markowitz (1952) Portfolio

Markowitz incorporate the trade-off between "maximizing return" and "minimizing variance" into one optimization problem by parameterize a fixed ratio λ for risk, where λ captures the preferences of risk for different people, and the problem becomes maximizing (return - $\lambda\cdot$ variance). The description of the optimization problem C is as following:

$$\max \quad \boldsymbol{\mu}^T\mathbf{w} - \lambda\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w} \tag{7}$$

$$\text{s.t.} \quad \sum_i \mathbf{w}_i = 1 \tag{8}$$

Summary

The three portfolio optimization problems all contain the corresponding parameters τ, β, λ, whose values determine the inclination of the investor in the trade-off between risk and return.

Classical Model: Non-convex Problem

The aim for us to introduce the non-convex problem part is that by adapting certain mathematical rules, the non-convex problem could be transform to convex problem, hence, the solution algorithm we use to solve the convex problem could also useful for the non-convex problem under this context.

Maximum Risk-adjusted Return Portfolio

Risk-adjusted return can be deemed as the λ in the objective function of Markowitz (1952) Portfolio (Eq.7) if we assert $\boldsymbol{\mu}^T\mathbf{w} - \lambda\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w} = 0$. Then, the objective of the optimization moved to the maximization of λ, which allows higher flexibility:

$$\max_w \quad \frac{R(\mathbf{w}) - r_f}{\phi_k(\mathbf{w})} \tag{9}$$

$$\text{s.t.} \quad \mathbf{A}^T\mathbf{w} \geq \mathbf{B} \tag{10}$$

$$\phi_i(\mathbf{w}) \leq c \qquad \forall i \tag{11}$$

where $\mathbf{w}$ is the weight vector, $R(\mathbf{w})$ is the return of the portfolio weighted according to $\mathbf{w}$, r_f is the risk-free return, $\phi_i(\mathbf{w})$ is the value of the risk measures of the portfolio weighted according to $\mathbf{w}$ (there might be various risk measures, while only one of the risk measures, $\phi_k(*)$, will be used in the objective function), $\mathbf{A}^T\mathbf{w} \geq \mathbf{B}$ is a series of linear constraints on the weighting, and $\phi_i(\mathbf{w}) \leq c$ is a series of linear constraint on the risk measures.

Maximum Sharpe Ratio Portfolio (MSRP)

A popular set-up of maximum risk-adjusted return portfolio optimization is the maximization of Sharpe ratio of the portfolio (Sharpe 1966), in which **Maximum Sharpe Ratio Portfolio (MSRP)**, which inherits the structure of Markowitz's Modern Portfolio Theory (mean-variance portfolio analysis) which defines return and risk with **expected return** ($\mathbf{w}^T\boldsymbol{\mu}$) and **volatility** ($\sqrt{\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}}$), where $\boldsymbol{\Sigma}$ is the **covariance matrix** (1952), but replaces the fixed $\frac{1}{\lambda}$

($\lambda > 0$) to a Sharpe Ratio (depending on $\mathbf{w}$) which is the objective. Accordingly, the optimization problem will be:

$$\max_{\mathbf{w}} \quad \frac{\mathbf{w}^T \boldsymbol{\mu} - r_f}{\sqrt{\mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w}}} \tag{12}$$

$$\text{s.t.} \quad \mathbf{1}^T \mathbf{w} = 1 \tag{13}$$

This problem belongs to fractional programming (FP) problems, which is not convex. Non-convex programming is usually hard to solve, while the maximization of the Sharpe ratio is a special subset of FP: **concave-convex single-ratio FP**, which can be solved rather easily by transforming them into convex problems (See Appendix A).

Model Convexization: QP-form of MSRP

By Schaible transform (See Appendix C): We first introduces two auxiliary variable $\tilde{\boldsymbol{w}}$, t:

$$\tilde{\boldsymbol{w}} = \frac{\mathbf{w}}{\sqrt{\mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w}}} \quad t = \frac{1}{\sqrt{\mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w}}} \tag{14}$$

Through this method, the concave-convex single-ratio **FP** problem of MSRP optimization can be reformulated to a **QP**:

$$\max_{\tilde{\boldsymbol{w}}, t} \quad \boldsymbol{\mu}^T \tilde{\boldsymbol{w}} - r_f t \tag{15}$$

$$\text{s.t.} \quad \tilde{\boldsymbol{w}}^T \boldsymbol{\Sigma} \tilde{\boldsymbol{w}} \leq 1 \tag{16}$$

$$\mathbf{1}^T \tilde{\boldsymbol{w}} = t \tag{17}$$

$$t \geq 0 \tag{18}$$

And after we get the optimal solutions of $\tilde{\boldsymbol{w}}$ and t, we can get the optimal $\mathbf{w}$ by:

$$\mathbf{w} = \frac{\tilde{\boldsymbol{w}}}{t}$$

Adjustment: Real-life Constraints

In the real world of investment, there are a couple of restrictions on security trading in some markets (e.g. Chinese stock market) or for some sort of investors (e.g. pension funds) to avoid excessively high trading frequency and thus control the level of risk and price fluctuation. These polcies are not in the theoretical framework of Modern Portfolio Theory.

One of the most harsh restriction is no-shorting policy. The other policies include leverage control, maximal/minimal single-asset position control, sparsity control (remove the assets with tiny weights), turnover limit (defines the maximal increase/decrease in the position of a single asset, which is a control of single-asset turnover) and, most commonly, the transaction fee, which is a control of total turnover. Considering these restrictions, we code them into constraints and get the general model:

$$
\begin{aligned}
\max_{\mathbf{w}} \quad & \lambda \boldsymbol{\mu}^T \mathbf{w} - \mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w} && && (19)\\
\text{s.t.} \quad & \|\mathbf{w_i} - \mathbf{w}_{i,0}\| \leq \gamma_1 && \text{Single-asset turnover} && (20)\\
& \|\mathbf{w} - \mathbf{w}_0\|_1 \leq \gamma_2 && \text{Total turnover} && (21)\\
& \|\mathbf{w}\|_1 \leq \gamma_3 && \text{Leverage} && (22)\\
& \|\mathbf{w}\|_\infty \leq \gamma_4 && \text{Single-asset max. position} && (23)\\
& \|\mathbf{w}\| \leq \gamma_5 && \text{Sparsity} && (24)\\
& \mathbf{w}^T \mathbf{1} = 1 && \text{Full investment} && (25)\\
& \mathbf{w} \geq \mathbf{0} && \text{No-shorting} && (26)
\end{aligned}
$$

where $\|x\|_1 = \sum_{i=1}^n \|x\|$, $\|x\|_\infty = \max_{i=1}^n \|x\|$ (Palomer 30-33).

Solution algorithm

Baseline Model solved by Lagrange

This is a convex quadratic problem (QP) with only one linear constraint which admits a closed-form solution. Recording to the model formulation part, we state the baseline optimization as following(We add a $\frac{1}{2}$ here at the risk measure part, because it is helpful for us to get more neat results after we take the first order condition, whereas, this term is not affected the final solution):

$$
\begin{aligned}
\max \quad & \boldsymbol{\mu}^T \mathbf{w} - \frac{1}{2} \lambda \mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w} && (27)\\
\text{s.t.} \quad & \boldsymbol{e}^T \mathbf{w} = 1 && (28)
\end{aligned}
$$

Where $\boldsymbol{e}$ is the ones vector.

The corresponding Lagrange is following:

$$L(\mathbf{w}, \lambda) = \mu^T \mathbf{w} - \frac{1}{2} \lambda \mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w} - \beta(\boldsymbol{e}^T \mathbf{w} - 1)$$

And the first order conditions are linear:

$$L_w = \mu - \boldsymbol{\Sigma} \mathbf{w} - \beta \boldsymbol{e} = 0 \tag{29}$$

$$L_{beta} = \boldsymbol{e}^T \mathbf{w} - 1 = 0 \tag{30}$$

We formulate this as a linear system as:

$$\begin{pmatrix} \lambda\mathbf{\Sigma} & \boldsymbol{e} \\ \boldsymbol{e}^T & 0 \end{pmatrix} \cdot \begin{pmatrix} \mathbf{w} \\ \beta \end{pmatrix} = \begin{pmatrix} \mu \\ 1 \end{pmatrix}$$

Such that

$$\begin{pmatrix} \mathbf{w}^* \\ \beta^* \end{pmatrix} = \begin{pmatrix} \lambda\mathbf{\Sigma} & \boldsymbol{e} \\ \boldsymbol{e}^T & 0 \end{pmatrix}^{-1} \cdot \begin{pmatrix} \mu \\ 1 \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} \cdot \begin{pmatrix} \mu \\ 1 \end{pmatrix}$$

By applying the block matrix inversion theorem, we could know that $\mathbf{w}^*$ is given by the first row of the inverted matrix times the constraint vector, where

$$\mathbf{w}^* = c_{11}\mu + c_{12}$$

By looking up the two inversions from wiki,

$$c_{11} = \frac{1}{\lambda}\mathbf{\Sigma}^{-1} - \frac{1}{\lambda}\mathbf{\Sigma}^{-1}\mathbf{e}(\mathbf{e}^T\frac{1}{\lambda}\mathbf{\Sigma}^{-1}\mathbf{e})^{-1}\mathbf{e}^T\frac{1}{\lambda}\mathbf{\Sigma}^{-1} \tag{31}$$

$$c_{12} = \frac{1}{\lambda}\mathbf{\Sigma}^{-1}\mathbf{e}(\mathbf{e}^T\frac{1}{\lambda}\mathbf{\Sigma}^{-1}\mathbf{e})^{-1} = \frac{\mathbf{\Sigma}^{-1}\mathbf{e}}{\mathbf{e}^T\mathbf{\Sigma}^{-1}\mathbf{e}} \tag{32}$$

By introducing the financial definitions of the characteristics of a portfolio:

$$a = \mathbf{e}^T\mathbf{\Sigma}^{-1}\mathbf{e} \tag{33}$$

$$b = \mathbf{e}^T\mathbf{\Sigma}^{-1}\mu \tag{34}$$

$$\mathbf{w}_{MVP} = \frac{\mathbf{\Sigma}^{-1}\mathbf{e}}{\mathbf{e}^T\mathbf{\Sigma}^{-1}\mathbf{e}} \tag{35}$$

$$\mathbf{w}_{Tangency} = \frac{\mathbf{\Sigma}^{-1}\mu}{\mathbf{e}^T\mathbf{\Sigma}^{-1}\mu} \tag{36}$$

Substituting these term into $c_{11}\mu$ and thus:

$$\mathbf{w}^* = c_{11}\mu + c_{12} = \mathbf{w}_{MVP} + \frac{1}{\lambda}b(\mathbf{w}_{Tangency} - \mathbf{w}_{MVP})$$

And finally, we note that $E(MVP) = \frac{b}{a}$ and $Var(MVP) = \frac{1}{a}$, so that we replace b in the final outcome:

$$\mathbf{w}^* = c_{11}\mu + c_{12} = \mathbf{w}_{MVP} + \frac{1}{\lambda}\frac{\mu_{MVP}}{\sigma^2_{MVP}}(\mathbf{w}_{Tangency} - \mathbf{w}_{MVP})$$

Interior Point Methods: Python package `cvxopt`

The alternative method we use for the situation with real-life constraints in `cvxopt`, a python package used to solve convex optimization problems.

$$\min \quad \frac{1}{2}\mathbf{x}^T P\mathbf{x} - \mathbf{q}^T\mathbf{x} \tag{37}$$

$$\text{s.t.} \quad \mathbf{G}\mathbf{x} \preceq \mathbf{h} \tag{38}$$

$$A\mathbf{x} = \mathbf{b} \tag{39}$$

For quadratic problems (QP), `cvxopt` applies interior point method (IPM) to solve them, taking $\{\mathbf{P},\mathbf{q},(\mathbf{G}),(\mathbf{h}),(\mathbf{A}),(\mathbf{b})\}$ as input. In our case, if $\lambda = 1/2$:

$$\min \quad \frac{1}{2}\mathbf{w}^T\mathbf{\Sigma}\mathbf{w} - \boldsymbol{\mu}^T\mathbf{w} \tag{40}$$

$$\text{s.t.} \quad \mathbf{1}^T\boldsymbol{w} = 1 \tag{41}$$

$$\mathbf{w} \geq \mathbf{0} \tag{42}$$

$\boldsymbol{P} = \mathbf{\Sigma}$, $\boldsymbol{q}$ = -$\boldsymbol{\mu}$, $\boldsymbol{G} = -\boldsymbol{I}$, $\boldsymbol{h} = \mathbf{0}$, $\boldsymbol{A} = \mathbf{1}$, b = 1

Data Source and Description

Data Source

The dataset we collecting, cleaning, processing is from the platform SimFin. SimFin is a platform for free fundamental financial data. Data in SimFin can be exported quickly and access via API.

Data Description

The dataset we conduct here is mainly use to conduct the computational results from the real-world data instead of the simulated data. From the free database we could use, we randomly get eight assets for five years daily data from the U.S. stock market which are from 2016-05-09 to 2021-05-06. We use the adjusted close price as the price of the stock on that paticular day. The eight assets we choose are: AAL (American Airlines Group Inc.), AAMC (Altisource Asset Management Corp), AAME (Atlantic American Corp.), AAOI (Applied Optoelectronics, Inc.), AAON (AAON Inc.), AAP (Advance Auto Parts Inc.), AAPL (Apple Inc.), ZTS(Zoetis). We shift the day and construct the daily return for each asset and we also build a function to construct the monthly return. We drop the missing value and we also tend to full in the missing value by using the average adjust close price for one day before and after. And totally, the shape of the dataframe is 1286 rows times 8 columns. The detailed data could be visualize and view in the python file we provided in the Appendix.

Data Availability

The main drawback of mean-variance optimization is that the theoretical treatment requires knowledge of the expected returns and the future risk-characteristics which characterized as covariance of the assets. However, in real-life cases, we can derive estimates of the expected return and covariance based on historical data, and the closer our estimates are to the real values, the better our portfolio will be. To calculate expected return, we may use the sample mean of the historical return.

Computational results and discussion

Simulation: Randomly-weighted portfolios

According to the part in the model formulation, we have one basic Markowitz optimization baseline model and then we have two other model with constraints that we add as an innovation part for our project. First, based on the baseline model, we randomly generate several portfolios with different expected returns and expected volatility. As Figure 2 shows, we generate 3000 observations of the portfolios.

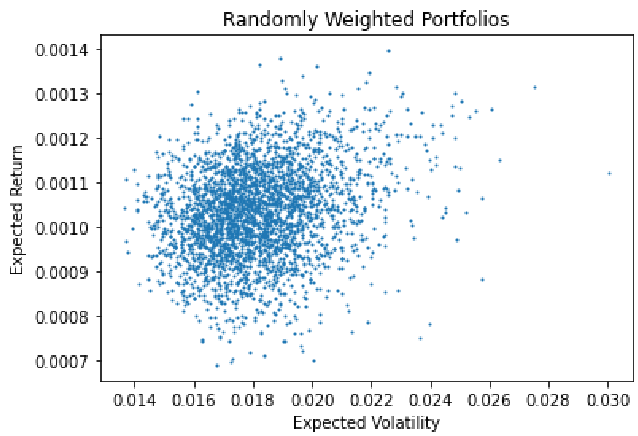

Figure 1: Randomly Weighted Portfolios(n=3000)

Efficient Frontier and GMVP

Furthermore, based on the solution algorithm of the baseline model, we further generate the Efficient Frontier for this set of portfolios which are shown in Figure 3. The yellow line represents the efficient portfolio and we constructed them by solving the baseline model. And the point in the Efficient Frontier but with the minimum expected volatility is the GMVP(global minimum variance portfolio). Usually, when we want to get the optimal solution of the Markowitz optimization, we actually trying to find the MVP. The optimal solution for the weights are solved as:
$(1.03e-07, 1.33e-01, 2.73e-02, 2.62e-07, 1.19e-07, 4.95e-08, 8.40e-01, 4.13e-07)$

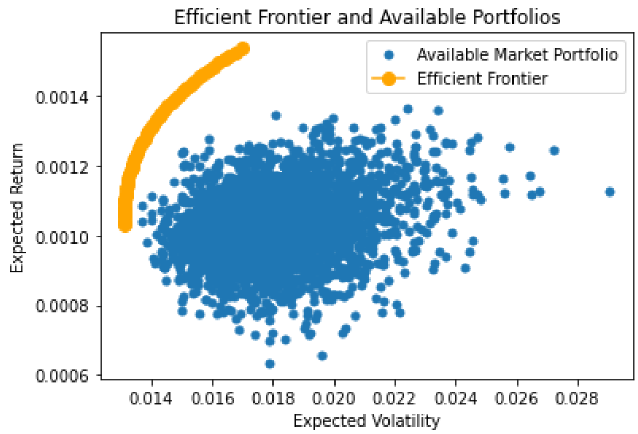

Figure 2: Efficient Frontier and Available Portfolios

Adjusted Model

Single-asset maximal position control

Next, we have update version of the baseline model, we further add on constrains for allowing only the long position which we need to require all the weights should be larger or equal to zero and at the same time,we limiting the weighting of each stock to no more than 30 percentage as an example. And furthermore:

$$\begin{aligned}
\max_{\mathbf{w}} \quad & \lambda \boldsymbol{\mu}^T \mathbf{w} - \mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w} && (43)\\
\text{s.t.} \quad & \|\mathbf{w}\|_\infty \leq 0.3 && \text{Single-asset max. position} \quad (44)\\
& \mathbf{w}^T \mathbf{1} = 1 && \text{Full investment} \quad (45)\\
& \mathbf{w} \geq \mathbf{0} && \text{No-shorting} \quad (46)
\end{aligned}$$

And the optimal solution are
$(3.55e-07, 2.59e-01,1.41e-01,1.11e-06,1.92e-06,3.87e-07,3.00e-01, 3.00e-01)$
which all the weights are positive so that the additional constraints fulfilled.

Single-asset turnover and total turnover control

Furthermore, by considering in the real-life case, when we buy or sell the stock, we need to pay a charge fee for the service which the fee is determined by the amount of the stocks we exchange. By considering this, we limit the upper limit of the total position transfer to which is less than 50 percentage, as well as the single asset position transfer to which is less than 10 percentage. As this problem need us to take into the consideration of the initial weights, we just assume that in the initial position, we have those eight assets all equal to 0.125 shares.

$$
\begin{aligned}
\max_{\mathbf{w}} \quad & \lambda \boldsymbol{\mu}^T \mathbf{w} - \mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w} && && (47) \\
\text{s.t.} \quad & \|\mathbf{w_i} - \mathbf{w}_{i,0}\| \leq 0.1 && \text{Single-asset turnover} && (48) \\
& \|\mathbf{w} - \mathbf{w}_0\|_1 \leq 0.5 && \text{Total turnover} && (49) \\
& \mathbf{w}^T \mathbf{1} = 1 && \text{Full investment} && (50) \\
& \mathbf{w} \geq \mathbf{0} && \text{No-shorting} && (51)
\end{aligned}
$$

The optimal solution are
$(2.50e-02, 1.62e-01, 1.38e-01, 7.50e-02, 1.25e-01, 2.50e-02, 2.25e-01, 2.25e-01)$
And through the results, we could see that those asset which are categories as "bad" assets through the share in the baseline model, actually decreasing the share from 0.125 to less than 0.125 in this adjusted model 3 as we could turnover the "bad" assets to get more profit.

Comparison of the adjusted models

By summing up the results of the optimal weights for different models, we could find that the best performance of the model belongs to the baseline and remaining the others as sub-optimals. That is because as we adding more and more constraints, it is hard to get the optimal position as the baseline model can achieve. Also, from the Table 1, the column SA-MP shows that the weights are all less than 30 percentage which means the contriants are effective. As for the model of the SA/T-MT model, we add the turnover constraints, as the initial weights is the first column indicates as Original. And through the change of the initial weight and the optimal weights after we limit the single asset turnover should less than 10 percentage and total turnover limit is 50 percentage, we could find that the "bad" assets such as AAL, AAOI, and AAP are actually decrease the shares through the turnover process and the "good" assets like AAMC, AAPL, and ZTS are actually increasing the shares.

Table 1: Comparison of the results from adjusted models

Asstes	Original	Baseline	SA-MP	SA/T-MT
AAL	12.5%			2.5%
AAMC	12.5%	13.3%	25.9%	16.2%
AAME	12.5%	2.7%	14.1%	13.8%
AAOI	12.5%			7.5%
AAON	12.5%			12.5%
AAP	12.5%			2.5%
AAPL	12.5%	84%	30%	22.5%
ZTS	12.5%		30%	22.5%
Objective Value		-0.001399	-0.001199	-0.001092

Conclusion

This final project covered a couple of portfolio optimization models, with a focus on the Quadratic Programming problems, and tried using Lagrangian and Interior Point Methods to solve them. By subsuming the real-life investment restrictions into constraints, our model can better fit to the needs of investors, which is essential for them. According to our computational examples, the introduction of any investment restriction can lead to significant differences in the weight vector, and thus incorporating the corresponding constraints in the portfolio management problem is necessary.

Appendix A: Concave-convex Single-ratio Fractional Programming

Concave-convex single-ratio FP is the FP problems in the form of:

$$\max_{\mathbf{x}} \quad \frac{f(\mathbf{x})}{g(\mathbf{x})} \tag{52}$$

$$\text{s.t.} \quad \mathbf{x} \in \mathcal{F}. \tag{53}$$

where numerator $f(\mathbf{x}) \geq 0$ is concave and denominator $g(\mathbf{x}) \geq 0$ is convex; $\mathcal{F}$ is the feasible region.Problems of these kinds can be converted into convex problems, such as SOCP (e.g. via Dinkelbach transform) and QP (e.g. via Schaible transform).

Appendix B: Dinkelbach transform

Dinkelbach transform introduces an auxiliary variable α:

$$\alpha^{(k)} = \frac{f(\mathbf{x}^{(k)})}{g(\mathbf{x}^{(k)})} \tag{54}$$

where k is the iteration index. Through this method, the concave-convex single-ratio FP problem can be reformulated to a **SOCP** through iterative Dinkelbach algorithm where $\mathbf{x}^{(k+1)}$ is updated as:

$$\max_{\mathbf{x}} \quad f(\mathbf{x}) - \alpha^{(k)} \cdot g(\mathbf{x}) \tag{55}$$

$$\text{s.t.} \quad \mathbf{x} \in \mathcal{F} \tag{56}$$

Appendix C: Schaible transform

Schaible transform introduces two auxiliary variable $\boldsymbol{\alpha}$, β:

$$\boldsymbol{\alpha} = \frac{\mathbf{x}}{g(\mathbf{x})} \quad \beta = \frac{1}{g(\mathbf{x})} \tag{57}$$

Through this method, the concave-convex single-ratio FP problem can be reformulated to a **QP**:

$$\max_{\boldsymbol{\alpha},\beta} \quad \beta \cdot f\left(\frac{\boldsymbol{\alpha}}{\beta}\right) \tag{58}$$

$$\text{s.t.} \quad \beta \cdot g\left(\frac{\boldsymbol{\alpha}}{\beta}\right) \leq 1 \tag{59}$$

$$\beta \geq 0 \tag{60}$$

$$\mathbf{x} = \frac{\boldsymbol{\alpha}}{\beta} \in \mathcal{F}. \tag{61}$$

Appendix D: Python Code

The code could be reproduce through Google Colab and the link is: https://colab.research.google.com/drive/13uiNyonJkKIbEC85SiGcslkU48v3vBsC?usp=sharing

The following codes are the main function we use to get the numerical results.

```
def Markowitz_orgi(data):
    '''
    the function need to return the optimal portfolio weights
    corresponding sigmas for a desired optimal portfolio return
    Params:
    - data: T x N matrix of observed data
    where T represents the date and N represents the number of assets
    '''
    returns = pd.DataFrame(data)
    cov = np.matrix(np.cov(returns.T))
    N = returns.shape[1]
    mu = returns.mean(axis=0)


    # constraint matrices for quadratic programming
    P = opt.matrix(cov)
    q = -opt.matrix(mu)
    G = -opt.matrix(np.eye(N))
    A = opt.matrix(1.0, (1,N))
    b = opt.matrix(1.0)
    h = opt.matrix(0.0, (N ,1))


    opt.solvers.options['show_progress'] = False

    '''
    - calculate portfolio weights, every weight vector is of size Nx1
    - find optimal weights with solver.qp(P, q, G, h, A, b)
    which we use it from the cvxopt directly
    - we will further explain how this package works
    and the algorithm in the final report
    '''
    optimal_weights = [solvers.qp(P, q, G, h, A, b)['x']]

    return optimal_weights



def Markowitz_update1(data):
```

```python
    returns = pd.DataFrame(data)
    cov = np.matrix(np.cov(returns.T))
    N = returns.shape[1]
    mu = returns.mean(axis=0)


    #constraints update with limiting the weighting of each stock to no more than 30%

    w_limit = 0.3
    G1 = np.eye(N)
    h1 = w_limit * np.ones([N, 1])

    G2 = -np.eye(N)
    h2 = np.zeros([N, 1])
    G = np.vstack((G1, G2))
    h = np.vstack((h1, h2))
    """
    we may also further can limit the total weight of stocks
    in a single industry to no more than a certain number
    Take 30 percentage as an example, but it need to get the data with industry label

    mk = 0.3
    type = np.array(industry_label)
    G3 = np.zeros([12, len(industry_label)])
    for i in range(12):
        G3[i] = (type==i)
    h3 = mk * np.ones([12, 1])
    """
    # constraint matrices for quadratic programming
    P = opt.matrix(cov)
    q = -opt.matrix(mu)
    G = opt.matrix(G)
    A = opt.matrix(1.0, (1,N))
    b = opt.matrix(1.0)
    h = opt.matrix(h)

    # solving process
    opt.solvers.options['show_progress'] = False

    optimal_weights = [solvers.qp(P, q, G, h, A, b)['x']]

    return optimal_weights

def Markowitz_update2(data):

    returns = pd.DataFrame(data)
```

```
mu = np.matrix(returns.mean())
cov = np.matrix(np.cov(returns.T))
N = returns.shape[1]

#constraints update part
m = 0.3
G1_x = np.eye(N)
G1_y = np.zeros_like(G1_x)
G1_z = np.zeros_like(G1_x)
G1 = np.hstack((G1_x, G1_y, G1_z))
h1 = m * np.ones([N, 1])

G2_x = -np.eye(N)
G2_y = np.zeros_like(G2_x)
G2_z = np.zeros_like(G2_x)
G2 = np.hstack((G2_x, G2_y, G2_z))
h2 = np.zeros([N, 1])
## Limit the transfer value of individual stocks which is smaller than 10%
G3_x = np.eye(N)
G3_y = -np.eye(N)
G3_z = np.zeros_like(G3_x)
G3 = np.hstack((G3_x, G3_y, G3_z))

G4_x = -np.eye(N)
G4_y = np.zeros_like(G4_x)
G4_z = -np.eye(N)
G4 = np.hstack((G4_x, G4_y, G4_z))

#set the initial weights as [0.125,0.125,0.125,0.125,0.125,0.125,0.125,0.125]
h3 = np.array([0.125]*8).reshape(8,1)
h4 = -np.array([0.125]*8).reshape(8,1)

G5 = np.hstack((G1_y, G1_x, G1_z))
h5 = 0.1 * np.ones([N, 1])

G6 = -G5
h6 = np.zeros([N, 1])

G7 = np.hstack((G1_y, G1_z, G1_x))
h7 = 0.1 * np.ones([N, 1])

G8 = -G7
h8 = np.zeros([N, 1])
## Limit the upper limit of the total position transfer to 50%
G9_y = np.ones([1, N])
G9_z = np.ones([1, N])
```

```python
G9_x = np.zeros([1, N])
h9 = np.array([0.5])
G9 = np.hstack((G9_x, G9_y, G9_z))

G = np.vstack((G1, G2, G3, G4, G5, G6, G7, G8, G9))
h = np.vstack((h1, h2, h3, h4, h5, h6, h7, h8, h9))
# equality constraints
A_x = np.ones([1, N])
A_y = np.zeros_like(A_x)
A_z = np.zeros_like(A_x)
A = np.hstack((A_x, A_y, A_z))
# setting the objective function
P_x = cov
P = np.zeros([3*N, 3*N])
P[0:N, 0:N] = cov

q = np.zeros([3*N, 1])
q[0:N] = -mu.T

# trun it into cvxopt matrices
P = opt.matrix(P)
q = opt.matrix(q)
A = opt.matrix(A)
b = opt.matrix(1.0)
G = opt.matrix(G)
h = opt.matrix(h)

# solving process
opt.solvers.options['show_progress'] = False

#calculate portfolio weights, every weight vector is of size Nx1
optimal_weights = [solvers.qp(P, q, G, h, A, b)['x']]
#sol = solvers.qp(P, q, G, h, A, b)

return optimal_weights
#return sol
```

Reference

Boyd, S., Vandenberghe, L. Convex Optimization. *Cambridge University Press* (2004), ISBN: 0521833787

Dinkelbach, Werner. "On Nonlinear Fractional Programming." *Management science* 13.7 (1967): 492-498.

Markowitz, H. (1952). Portfolio Selection. *Journal of Finance* 7(1), (1952): 77–91.https://EconPapers.repec.org/RePEc:bla:jfinan:v:7:y:1952:i:1:p:77-91

Palomar, Daniel P. "Portfolio Optimization with R." *Hong Kong University of Science and Technology.*

Renato Bruni, Francesco Cesarone, Andrea Scozzari, Fabio Tardella, Real-world datasets for portfolio selection and solutions of some stochastic dominance portfolio models. *Data in Brief* Volume 8, 2016, Pages 858-862, ISSN 2352-3409, https://doi.org/10.1016/j.dib.2016.06.031.

Schaible, Siegfried. "Parameter-free Convex Equivalent and Dual Programs of Fractional Programming Problems." *Zeitschrift für Operations Research* 18.5 (1974): 187-196.

Sharpe, William F. "Capital asset prices: A theory of market equilibrium under conditions of risk." *The Journal of Finance* 19.3 (1964): 425-442.

Sharpe, William F. "Mutual Fund Performance." *The Journal of business* 39.1 (1966): 119-138.