New York University Shanghai

# Portfolio Optimization and Non-Linear Programming

**BUSF-SHU 210: Business Analytics (Spring 2019)**

## NLP Formulation of Portfolio Optimization

Recall that our objective is to maximize mean-variance utility:

$$\text{Expected Annual Return } - \lambda * \text{Variance of Annual Return}$$

Hence, we first calculate the expected annual return and the variance of annual return separately.

Some initial analysis of the data set `Portfolio.csv` yields that the expected annual return rate of each stock can be summarized in the following table:

| Index | Stock | Expected Annual Return | Standard Deviation of Annual Return |
|---|---|---|---|
| 1 | AAPL | 0.114 | 0.039 |
| 2 | AMZN | 0.103 | 0.030 |
| 3 | DIS | 0.092 | 0.032 |
| 4 | WFM | 0.085 | 0.029 |
| 5 | WMT | 0.078 | 0.022 |
| 6 | Bond | 0.05 | 0 |

We use $\mu_i$ to denote the expected annual return of Stock $i$ (Stock 0 is the bond), and $\sigma_i$ to denote the standard deviation of annual stock return of Stock $i$. We also use the monthly stock return data to estimate the correlation between the stock returns, summarized in the following table:

| | AAPL | AMZN | DIS | WFM | WMT | Bond |
|---|---|---|---|---|---|---|
| AAPL | 1 | 0.160 | 0.163 | -0.260 | 0.399 | 0 |
| AMZN | 0.160 | 1 | 0.029 | 0.272 | -0.193 | 0 |
| DIS | 0.163 | 0.029 | 1 | 0.173 | 0.124 | 0 |
| WFM | -0.260 | 0.272 | 0.173 | 1 | 0.125 | 0 |
| WMT | 0.399 | -0.193 | 0.124 | 0.125 | 1 | 0 |
| Bond | 0 | 0 | 0 | 0 | 0 | 1 |

We use $\rho_{ij}$ to denote the correlation between the annual return of Stock $i$ and Stock $j$.

### Decision Variables

The decision variables of the portfolio optimization problem are the quantities of each stock in the portfolio:

- $x_i$: The quantity of Stock $i$ in the portfolio.

We define the investment portfolio as $x = (x_1, x_2, x_3, x_4, x_5, x_6)'$.

### Objective Function

To characterize the objective function, we first calculate the expected annual return of portfolio $x$:

$$\text{Mean Annual Return} = \sum_{i=1}^{6} \mu_i x_i$$

To calculate the variance of annual return for the portfolio $x$, we define the random variable $Z_i$ as the return per unit Stock $i$. Thus, the total annual return of portfolio $x$ is

$$\sum_{i=1}^{6} x_i Z_i$$

Hence, we can calculate the variance of total annual return is

$$\begin{aligned}
\text{Var}(\sum_{i=1}^{6} x_i Z_i) =& \mathbb{E}(\sum_{i=1}^{6} x_i Z_i - \sum_{i=1}^{6} \mu_i x_i)^2 \\
=& \mathbb{E}[\sum_{i=1}^{6} x_i (Z_i - \mu_i)]^2 \\
=& \sum_{i=1}^{6} \sum_{j=1}^{6} \mathbb{E}[x_i x_j (Z_i - \mu_i)(Z_j - \mu_j)] \\
=& \sum_{i=1}^{6} \sum_{j=1}^{6} [\sigma_i \sigma_j \rho_{ij} x_i x_j],
\end{aligned}$$

where the third inequality follows from the multinomial expansion of $[\sum_{i=1}^{6} x_i (Z_i - \mu_i)]^2$ and the last follows from the definition of correlation that $\rho_{ij} = \mathbb{E}[(Z_i - \mu_i)(Z_j - \mu_j)]/(\sigma_i \sigma_j)$. Therefore, the objective function i.e., the mean-variance utility is

$$MV(x) = \sum_{i=1}^{6} \mu_i x_i - \lambda(\sum_{i=1}^{6} \sum_{j=1}^{6} \sigma_i \sigma_j \rho_{ij} x_i x_j)$$

### Constraints

Besides the non-negativity constraint, the only thing left is the budget constraint:

$$\sum_{i=1}^{6} x_i = 1$$

For the convenience of solving the model in $R$, we express the budget constraint as:

$$B(x) = \sum_{i=1}^{6} x_i - 1 = 0$$

| $\lambda$ | Mean Return | SD Return | $x_1^*$ | $x_2^*$ | $x_3^*$ | $x_4^*$ | $x_5^*$ | $x_6^*$ |
|---|---|---|---|---|---|---|---|---|
| 0 | 11.4% | 3.9% | 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 11.4% | 3.9% | 1 | 0 | 0 | 0 | 0 | 0 |
| 10 | 10.9% | 2.80% | 0.595 | 0.405 | 0 | 0 | 0 | 0 |
| 50 | 9.87% | 1.79% | 0.265 | 0.312 | 0.151 | 0.197 | 0.0746 | 0 |
| 100 | 8.90% | 1.40% | 0.134 | 0.263 | 0.133 | 0.140 | 0.215 | 0.116 |
| 500 | 5.78% | 0.28% | 0.0267 | 0.0527 | 0.0265 | 0.0280 | 0.0429 | 0.823 |
| $1,000$ | 5.39% | 0.14% | 0.0134 | 0.0263 | 0.0133 | 0.0140 | 0.0215 | 0.912 |

Table 1: Mean and Standard Deviation of Annual Return

Putting everything together, we formulate the optimization model for the portfolio optimization problem as

$$\max\ MV(x) = \sum_{i=1}^{6} \mu_i x_i - \lambda(\sum_{i=1}^{6}\sum_{i=1}^{6} \sigma_i \sigma_j \rho_{ij} x_i x_j)$$

Subject to

$$B(x) = \sum_{i=1}^{6} x_i - 1 = 0$$

where $\mu_i$ is the expected annual return of Stock $i$, $\sigma_i$ is the standard deviation of annual return for Stock $i$, $\rho_{ij}$ is the correlation between the annual returns of Stock $i$ and Stock $j$, and $\lambda$ is the parameter that captures the weight of return and risk for the investor.

We can also write the objective function and constraint in matrix form:

$$MV(x) = -\lambda \cdot x' \cdot Cov \cdot x - \mu' x,$$

$$B(x) = (1,1,1,1,1,1)x - 1,$$

where $Cov_{6\times 6}$ is the covariance matrix ($Cov_{ij} = \rho_{ij}\sigma_i\sigma_j$) and $\mu = (\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \mu_6)'$ is the vector representing the average return.

We use the package "`cvxopt`" and "`scipy`" in Python to solve the portfolio optimization problem. The solution to this model is given in the Jupyter Notebook `Convex-Optimization.ipynb` As we can see from Table 1, for different weights of risk and return, the optimal invest strategies are drastically different. The key message here is that: As $\lambda$ increases, the investor cares more and more about controlling risk relative to obtaining high return, so both the mean and the standard deviation of annual return will decrease.

## Gradient and Jacobian for Multi-Variable Functions

A lot of algorithms that solve non-linear program models rely on exploiting the gradient of the objective function and the Jacobian of the constraint functions.

## Gradient

Gradient is the generalization of derivative for multi-variable functions, which is useful to locally approximate a non-linear function with a linear one. Specifically, let $f(x) = f(x_1, x_2, x_3, \cdots, x_n)$ be a differentiable function defined on $\mathbb{R}^n$. Then, the gradient of $f(x)$ is defined as a vector-valued function:

$$\text{Grad}[f(x)] = \left(\frac{\partial f(x)}{\partial x_1}, \frac{\partial f(x)}{\partial x_2}, \cdots, \frac{\partial f(x)}{\partial x_n}\right),$$

where $\frac{\partial f(x)}{\partial x_i}$ is the partial derivative of $f(x)$ with respect to $x_i$. In other words, it is the derivative of $f(x)$ with respect to $x_i$, holding all other variables $x_1$, $x_2$, $x_3$, $\cdots$, $x_{i-1}$, $x_{i+1}$, $\cdots$, $x_n$ fixed.

Now we give some examples.

- If $f_1(x) = (3x_1 + 2x_2)^2$, the gradient of $f_1(x)$ is
$$\text{Grad}[f_1(x)] = (18x_1 + 12x_2, 12x_1 + 8x_2)$$
- If $f_2(x) = \exp(2x_1 + x_2)$, the gradient of $f_2(x)$ is
$$\text{Grad}[f_2(x)] = (2\exp(2x_1 + x_2), \exp(2x_1 + x_2))$$
- If $f_3(x) = (\log(x_1 + 2x_2))^2$, the gradient of $f_3(x)$ is
$$\text{Grad}[f_3 x] = \left(\frac{2\log(x_1 + 2x_2)}{x_1 + 2x_2}, \frac{4\log(x_1 + 2x_2)}{x_1 + 2x_2}\right)$$
- For $MV(x) = \sum_{i=0}^5 \mu_i x_i - \lambda(\sum_{i=0}^5 \sum_{i=0}^5 \sigma_i \sigma_j \rho_{ij} x_i x_j)$, its gradient is
$$\text{Grad}[MV(x)] = (\mu_1 - 2\lambda(\sum_{j=1}^{6} \sigma_1 \sigma_j \rho_{1j} x_j), \mu_2 - 2\lambda(\sum_{j=1}^{6} \sigma_2 \sigma_j \rho_{2j} x_j), \cdots, \mu_0 - 2\lambda(\sum_{j=1}^{6} \sigma_0 \sigma_j \rho_{0j} x_j))$$

## Jacobian

Jacobian is the generalization of Gradient for vector-valued functions, which is useful to locally approximate a non-linear vector-valued function with a linear one. Specifically, let

$$g(x) = (g_1(x), g_2(x), \cdots, g_m(x))'$$

be a differentiable function defined on $\mathbb{R}^n$ (i.e., $x = (x_1, x_2, \cdots, n)$). Then, the Jacobian of function $g(x)$ is defined as a matrix-valued function:

$$\text{Jac}[g(x)] = \begin{pmatrix} \frac{\partial g_1(x)}{\partial x_1} & \frac{\partial g_1(x)}{\partial x_2} & \cdots & \frac{\partial g_1(x)}{\partial x_n} \\ \frac{\partial g_2(x)}{\partial x_1} & \frac{\partial g_2(x)}{\partial x_2} & \cdots & \frac{\partial g_2(x)}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial g_m(x)}{\partial x_1} & \frac{\partial g_m(x)}{\partial x_2} & \cdots & \frac{\partial g_m(x)}{\partial x_n} \end{pmatrix}$$

where $\frac{\partial g_i(x)}{\partial x_j}$ is the partial derivative of (real-number-valued) function $g_i(x)$ with respect to $x_j$. In other words, it is the derivative of $g_i(x)$ with respect to $x_j$, holding all other variables $x_1$, $x_2$, $x_3$, $\cdots$, $x_{j-1}$, $x_{j+1}$, $\cdots$, $x_n$ fixed.

Now we give some examples.

- If $g(x) = (g_1(x), g_2(x))'$, where $g_1(x)x_1 + x_2$ and $g_2(x) = 2x_1 - x_2$, the Jacobian of $g(x)$ is

$$\text{Jac}[g(x)] = \begin{pmatrix} 1 & 1 \\ 2 & -1 \end{pmatrix}$$

- If $g(x) = (g_1(x), g_2(x))'$, where $g_1(x) = \exp(x_1 + x_2)$ and $g_2(x) = (x_1 - x_2)^2$, the Jacobian of $g(x)$ is

$$\text{Jac}[g(x)] = \begin{pmatrix} \exp(x_1 + x_2) & \exp(x_1 + x_2) \\ 2x_1 - 2x_2 & 2x_2 - 2x_1 \end{pmatrix}$$

- For $B(x) = \sum_{i=1}^{6} x_i - 1$, its Jacobian is

$$\text{Jac}[B(x)] = (1, 1, 1, 1, 1, 1)$$