## 2.1 Lab Session 1: Portfolio Optimization

We consider the portfolio optimization problem of Example 1.

### 2.1.1 Uniform Portfolio.

Under Matlab (or Python), import the datasets from the SP, NYSE and HSI stock market indexes (provided in **log returns** and not linear returns). We shall denote $n$ the number of assets in the selected stock market.

For

$$w = w_{\text{uniform}} \equiv \frac{1}{n} 1_n$$

plot the values of the returns $x_t^\mathsf{T} w_{\text{uniform}}$ for the remaining 30 days. Then estimate, based on these 30 days, the obtained empirical variance (called *volatility* in finance).

### 2.1.2 Unconstrained Optimization.

We assume the stock market returns form a stationary ($n$-variate) time series. Estimate in particular the covariance matrix $C$ by means of the sample covariance matrix of the $n$ stock market (linear) returns $x_1, x_2, \ldots \in \mathbb{R}^n$ from 'day 1' to '30 days before the end of the recording' (be careful to use linear and not log returns).

Using the method of Lagrange multipliers, show that the problem

$$\operatorname{argmin}_{w \in \mathbb{R}^n} w^\mathsf{T} C w \text{ such that } \sum_{i=1}^{n} [w]_i = 1$$

has for unique solution

$$w_{\text{unconstrained}} = \frac{C^{-1} 1_n}{1_n^\mathsf{T} C^{-1} 1_n}.$$

Implement the formula and compare the values of the returns and the empirical variance now achieved to the uniform case. Comment also on the value of the entries of $w_{\text{unconstrained}}$.

### 2.1.3 Constrained Optimization.

The problem of the previous optimization is that it may return negative values for the entries of $w_{\text{unconstrained}}$, forcing "short selling". We now assume short selling is impossible (one can only buy and not sell on the market). We thus impose that $[w]_i \geq 0$ for all $i = 1, \ldots, n$.

As such, we now consider the constrained optimization problem with positivity constraints on the entries of $w$, that is

$$\operatorname{argmin}_{w \in \mathbb{R}^n} w^\mathsf{T} C w \text{ such that } \sum_{i=1}^{n} [w]_i = 1 \text{ and } [w]_i \geq 0.$$

#### Naive approach

We first propose to solve the problem by means of a simple 'naive' method consisting in proceeding to a gradient descent method on the unconstrained optimization problem mixed with projections to the constraint set.

Propose and implement such a method (or several approaches) and discuss their associated advantages and limitations.

#### Advanced optimization

We now proceed to a careful simplification of the problem by successively (i) *eliminating* the equality constraint (i.e., turning the problem into one with only inequality constraints) and (ii) solving the resulting inequality constrained optimization with a barrier method.

Explain why eliminating the equality constraint is convenient for the proposed barrier optimization approach. Could we have solved the problem differently? (think for instance of dual approaches)

We thus first proceed to the elimination of the equality constraint. To this end, show that we can recast the problem as

$$\operatorname{argmin}_{\tilde{w}\in\mathbb{R}^n} w^\top C w \text{ such that } w = P\tilde{w} + \frac{1}{n}1_n \text{ and } [w]_i \geq 0$$

for $P = I_n - \frac{1}{n}1_n 1_n^\top$, the projector operator on the orthogonal complement of span $(1_n)$.

Show however that the resulting cost function has a vanishing gradient for all vectors $\tilde{w}$ on a non-trivial subspace. What practical implementation problem can that lead to?

To avoid the problem, we now impose $[\tilde{w}]_1 = 1$. Show that the solution to the previous problem is equivalent to that of

$$\operatorname{argmin}_{\tilde{w}\in\mathbb{R}^{n-1}} w^\top C w \text{ such that } w = P\left[\begin{smallmatrix} 1 \\ \tilde{w} \end{smallmatrix}\right] + \frac{1}{n}1_n \text{ and } [w]_i \geq 0$$

and that the resulting cost function no longer has a zero-gradient subspace.

As such, the problem can now be solved by a barrier method by relaxing the constraint $[w]_i \geq 0$ for all $i = 1,\ldots,n$, by a log-barrier constraint $-\mu\sum_{i=1}^n \log([w]_i)$. Start by computing the gradient of the new cost function, that is, the gradient with respect to $\tilde{w}\in\mathbb{R}^{n-1}$ of

$$\left(P\left[\begin{smallmatrix} \frac{1}{n} \\ \tilde{w} \end{smallmatrix}\right] + \frac{1}{n}1_n\right)^\top C\left(P\left[\begin{smallmatrix} \frac{1}{n} \\ \tilde{w} \end{smallmatrix}\right] + \frac{1}{n}1_n\right) - \mu\sum_{i=1}^n \log\left(\left[P\left[\begin{smallmatrix} \frac{1}{n} \\ \tilde{w} \end{smallmatrix}\right] + \frac{1}{n}1_n\right]_i\right).$$

Implement the barrier method to solve the problem. The solution will be denoted $w_{\text{constrained}}$. Be especially careful to ensure that each gradient descent step does not break a barrier constraint (backtracking with decreasing step sizes could be used here). A thin control on the progression of the algorithm is also recommended to avoid early stopping.

Finally evaluate the returns $x_t^\top w_{\text{constrained}}$ achieved on the 30 last days, plot $w_{\text{constrained}}$ and compute the variance of the returns. Compare to the uniform and constrained solutions.

### 2.1.4 Visualization of the Solution

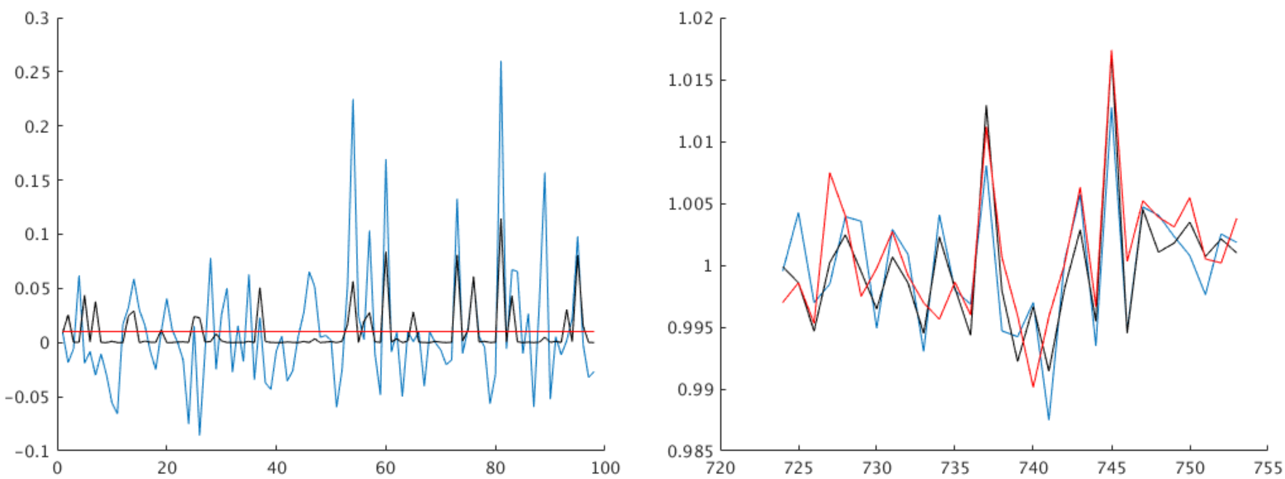


Figure 2.1: Portfolios allocated (left) and associated future returns (right) for the unconstrained optimization (blue), the constrained optimization (black) and the uniform allocation (red).