# 4 Robust MVO

## 4.1 Robust Optimization

In the problems we have seen so far, we are estimating our input parameters from raw data. Estimated parameters will always have a degree of uncertainty. If the uncertainty is too large, it may render the solution to our optimization problem useless. Robust optimization deals with this issue by accepting that our inputs are noisy, and incorporating their uncertainty as deterministic variability into the optimization problem.

Example:

$$\begin{aligned} \min_x \quad & 2x_1 + 3x_2 \\ \text{s.t.} \quad & \underbrace{2}_{\text{noisy}} x_1 + \underbrace{1}_{\text{noisy}} x_2 \geq 1 \\ \Rightarrow \text{ Solution: } x^* = & \begin{bmatrix} 0.5 \\ 0 \end{bmatrix}, \quad \text{Obj. Value: } z^* = 1 \end{aligned}$$

However, what if our coefficients are noisy? Suppose the real problem is:

$$\begin{aligned} \min_x \quad & 2x_1 + 3x_2 \\ \text{s.t.} \quad & 1.99x_1 + 0.99x_2 \geq 1 \end{aligned}$$

Even though these are small changes, $x^*$ is now infeasible.

Uncertainty set

$$U = \left\{ \begin{bmatrix} 1.99 \\ 0.99 \end{bmatrix}, \underbrace{\begin{bmatrix} 2 \\ 1 \end{bmatrix}}_{\text{Nominal values}}, \begin{bmatrix} 2.01 \\ 1.01 \end{bmatrix} \right\}$$

The robust counterpart (RC) to our nominal problem is

$$\begin{aligned} \min_x \quad & 2x_1 + 3x_2 \\ \text{s.t.} \quad & 1.99x_1 + 0.99x_2 \geq 1 \\ & 2x_1 + 1x_2 \geq 1 \\ & 2.01x_1 + 1.01x_2 \geq 1 \\ \rightarrow \text{Solution: } x^* = & \begin{bmatrix} 0.5025 \\ 0 \end{bmatrix}, \quad z^* = 1.005 \end{aligned}$$

where the extra 0.005 in $z^*$ is the cost of robustness.

Any feasible solution for the RC will satisfy all constraints, thereby ensuring our solution is braced against uncertainty.

## 4.2 General Robust Optimization

Nominal LP:

$$\begin{aligned} \min_{x} \quad & c^T x \\ \text{s.t.} \quad & Ax \geq b \\ & c,\ x \in \mathbb{R}^n,\ b \in \mathbb{R}^m,\ A \in \mathbb{R}^{m \times n} \end{aligned}$$

Robust Counterpart (RC):

$$\begin{aligned} \min_{x} \quad & c^T x \\ \text{s.t.} \quad & Ax \geq b \\ & \forall\ (c, A, b) \in U \\ & U \subset \mathbb{R}^n \times \mathbb{R}^{m \times n} \times \mathbb{R}^m \end{aligned}$$

RC is equivalent to:

$$\left.\begin{aligned} \min_{t,x} \quad & t \\ \text{s.t.} \quad & t \geq c^T x \\ & Ax \geq b \end{aligned}\right\} \quad \forall\ (c, A, b) \in U$$

Goal: We want a constraint-level representation of robustness

$$\rightarrow \quad t \geq c^T x \quad \forall\ c$$

Idea: Formulate a constraint-wise RC

For each constraint $a_i x \geq b_i$ in $Ax \geq b$, where $a_i$ is the $i^{\text{th}}$ row of $A$, we can write $a_i x \geq b_i\ \forall\ (a_i, b_i) \in U_i$.

$\rightarrow$ $U_i$ is the restriction of $U$ to elements only relevant to the realization of coefficients $a_i$ and $b_i$ corresponding to the $i^{\text{th}}$ constraint.

Consider

$$\left.\begin{aligned} \min_{x} \quad & 2x_1 - 3x_2 \\ \text{s.t.} \quad & a_{11}x_1 + a_{12}x_2 \geq b_1 \\ & a_{21}x_1 + a_{22}x_2 \geq b_2 \end{aligned}\right\} \quad \text{Nominal LP}$$

and let the constraint coefficients be noisy, governed by the uncertainty set

$$\begin{bmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \end{bmatrix} \in U, \qquad \text{Let } (a_i, b_i) = (\underbrace{a_{i1}, a_{i2}}_{a_i}, b_i)$$

$$U = \left\{ \begin{bmatrix} 0.95 & 1.95 & 0.95 \\ 2.95 & 1.95 & 1.95 \end{bmatrix}, \begin{bmatrix} 1 & 2 & 1 \\ 3 & 2 & 2 \end{bmatrix}, \begin{bmatrix} 1.05 & 2.05 & 1.05 \\ 3.05 & 2.05 & 2.05 \end{bmatrix} \right\}$$

$$U_1 = \{(0.95, 1.95, 0.95),\ (1, 2, 1),\ (1.05, 2.05, 1.05)\}$$
$$U_2 = \{(2.95, 1.95, 1.95),\ (3, 2, 2),\ (3.05, 2.05, 2.05)\}$$

So now the constraint-wise RC is

$$\begin{aligned} \min_{t,x} \quad & t \\ \text{s.t.} \quad & t \geq 2x_1 - 3x_2 \\ & a_{11}x_1 + a_{12}x_2 \geq b_1 \quad \forall\ (a_{11}, a_{12}, b_1) \in U_1 \\ & a_{21}x_1 + a_{22}x_2 \geq b_1 \quad \forall\ (a_{21}, a_{22}, b_2) \in U_2 \end{aligned}$$

We can have 2 possible types of scenarios:

- $U$ is finite, then we can model the RC as an LP with a finite number of constraints (as shown in the example above).
- $U$ is infinite, then we can model the RC as a semi-infinite LP.

### 4.3 $U$ infinite case

Suppose the nominal matrix (e.g., the matrix estimated from some raw data) is the following

$$\begin{bmatrix} 1 & 2 & 1 \\ 3 & 2 & 2 \end{bmatrix}$$

We want to consider all possible matrices that result from perturbing each element in the nominal matrix by adding a number in [-0.5, 0.5], e.g.,

$$\begin{bmatrix} 0.5 & 2.1 & 1.5 \\ 3.3 & 1.78 & 2.13 \end{bmatrix}$$

$$U = \left\{ \begin{bmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 2 & 1 \\ 3 & 2 & 2 \end{bmatrix} + \sum_{j=1}^{6} \zeta_j P_j}_{\text{nominal + perturbation}}, \quad \underbrace{-1 \leq \zeta_j \leq 1}_{\text{Uncountably infinite}} \quad , \quad j = 1, ..., 6 \right\}$$

with the perturbation set

$$B = \{\zeta \in \mathbb{R}^6 \mid\ -1 \leq \zeta_j \leq 1,\ j = 1, ..., 6\}.$$

and

$$P_1 = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad P_2 = \begin{bmatrix} 0 & 0.5 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad P_3 = \begin{bmatrix} 0 & 0 & 0.5 \\ 0 & 0 & 0 \end{bmatrix}$$
$$P_4 = \begin{bmatrix} 0 & 0 & 0 \\ 0.5 & 0 & 0 \end{bmatrix}, \quad P_5 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.5 & 0 \end{bmatrix}, \quad P_6 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}$$

How to formulate this RC? Extract $U_1$, $U_2$ from this so we can get a constraint-wise form

$$U_1 = \left\{(a_1, b_1) = (1, 2, 1) + \sum_{j=1}^{3} \zeta_1^j P_1^j \mid \zeta_1 \in B_1\right\}$$

where

$$P_1^1 = (0.5, 0, 0), \quad P_1^2 = (0, 0.5, 0), \quad P_1^3 = (0, 0, 0.5)$$
$$B_1 = \left\{\zeta_1 \in \mathbb{R}^3 \mid -1 \leq \zeta_1^j \leq 1,\ j = 1, 2, 3\right\}$$

and we do the same for $U_2$.

What we want: 'Robustify' each constraint at once

$$\begin{aligned} a_i x \geq b_i \quad &\rightarrow \quad a_i x \geq b_i \ \forall\ (a_i, b_i) \in U_i \\ &\rightarrow \quad a_i x \geq b_i \ \forall\ \left\{(a_i, b_i) = (a_i^0, b_i^0) + \sum_{j=1}^{J_i} \zeta_i^j P_i^j \mid \zeta_i \in B_i\right\} \end{aligned}$$

where $J_i$ is the number of coefficients in constraint $i$ deemed to be noisy.

Theorem:

Suppose $B_i = \left\{\zeta_i \in \mathbb{R}^{J_i} \mid \underbrace{||\zeta_i||_\infty \leq 1}_{\max |\zeta_i^j| \leq 1}\right\}$, then the RC is an LP.

$$\begin{aligned} & a_i x \geq b_i \ \forall\ \left\{(a_i, b_i) = (a_i^0, b_i^0) + \sum_{j=1}^{J_i} \zeta_i^j \underbrace{(a_i^j, b_i^j)}_{P_i^j = (a_i^j, b_i^j)} \mid \zeta_i \in B_i\right\} \\ \iff & \left(a_i^0 + \sum_{j=1}^{J_i} \zeta_i^j a_i^j\right) x \geq \left(b_i^0 + \sum_{j=1}^{J_i} \zeta_i^j b_i^j\right) \quad \forall\ \zeta_i \in B_i \\ \iff & \sum_{j=1}^{J_i} \zeta_i^j \left(a_i^j x - b_i^j\right) \geq b_i^0 - a_i^0 x \quad \forall\ \zeta_i \in B_i \ \left(\text{i.e., } |\zeta_i^j| \leq 1,\ j = 1, ..., J_i\right) \end{aligned}$$

which must hold for

$$\min_{-1 \leq \zeta_i^j \leq 1} \quad \sum_{j=1}^{J_i} \zeta_i^j \left(a_i^j x - b_i^j\right) \geq b_i^0 - a_i^0 x$$

Then minimum value of the left-hand side (LHS) is

$$-\sum_{j=1}^{J_i} |a_i^j x - b_i^j|$$

so we have that

$$-\sum_{j=1}^{J_i} |a_i^j x - b_i^j| + a_i^0 x \geq b_i^0$$

Finally, all we need to do is remove the absolute value from constraint $i$ above,

$$\begin{aligned} y_i^j &\geq a_i^j x - b_i^j \\ y_i^j &\geq -a_i^j x + b_i^j \\ \Rightarrow -\sum_{j=1}^{J_i} y_i^j + a_i^0 x &\geq b_i^0 \end{aligned}$$

and we do this for all constraints.

## 4.4 Parameter Uncertainty

Why do we need robustness? The 'true' value governing an observed distribution is typically unknown, and parameters estimated from raw historical data will always have a degree of uncertainty.

In general, estimated variances and covariances are reasonably reliable. However, this is usually not the case for the estimated expected returns. This uncertainty is not due to faulty data, but rather because of a fundamental limitation of the process of computing estimates from data.

Let's consider our yearly return on a stock, $r_y$, which is a random variable. This can be written in terms of monthly returns

$$1 + r_y = (1 + r_1)(1 + r_2)...(1 + r_{12}),$$

where these monthly returns $r_t$ are also random variables. If the monthly returns are small, we can approximate this as

$$1 + r_y \approx 1 + r_1 + r_2 + ... + r_{12} = 1 + \sum_{t=1}^{12} r_t$$

Now, let's assume that these monthly returns have the same statistical properties and are mutually uncorrelated. In other words, $r_t \sim \mathcal{N}(\bar{r}, \sigma)$ for all $t$'s. Then

$$\begin{aligned} \bar{r}_y &= \mathbb{E}[r_y] = \mathbb{E}[\sum_{t=1}^{12} r_t] = 12\bar{r} \\ \sigma_y^2 &= \text{var}(r_y) = \text{var}\Big(\sum_{t=1}^{12} r_t\Big) = \sum_{t=1}^{12} \text{var}(r_t) = 12\sigma^2 \\ \rightarrow \sigma_y &= \sqrt{12}\sigma \end{aligned}$$

Consider a stock with mean yearly return $\bar{r}_y = 12\%$ and yearly standard deviation $\sigma_y = 15\%$. Thus, by our previous relationship, we have

$$\begin{aligned} \bar{r} &= \frac{\bar{r}_y}{12} = \frac{12\%}{12} = 1\% \\ \sigma &= \frac{\sigma_y}{\sqrt{12}} = \frac{15\%}{\sqrt{12}} = 4.33\% \end{aligned}$$

When we use yearly values, the ratio of our standard deviation to the mean was

$$\frac{\sigma_y}{\bar{r}_y} = \frac{15}{12} = 1.25$$

whereas now that we are using monthly values we have

$$\frac{\sigma}{\bar{r}} = \frac{4.33}{1} = 4.33$$

The shorter our frequency of measurement, the greater this ratio will be and the more uncertain our estimated expected returns become.

So, should we stick to using low frequency measurements, such as yearly estimates? Not necessarily!

## 4.5 Mean Blur

Using less frequent observations (e.g., yearly) reduces the quality of our estimates. Let's assume the asset returns are governed by a distribution with 'true' parameters $\bar{r}$ and $\sigma$. We want to estimate these parameters from some historical data

$$\widehat{\bar{r}} = \frac{1}{N}\sum_{t=1}^{N} r_t$$

The estimate of the expected returns, $\widehat{\bar{r}}$, is itself random. If we were to choose a different set of $N$, observations we would get a different estimate. If we take the expectation of this estimate we get

$$\mathbb{E}[\widehat{\bar{r}}] = \mathbb{E}\Big[\frac{1}{N}\sum_{t=1}^{N} r_t\Big] = \frac{1}{N}\mathbb{E}\Big[\sum_{t=1}^{N} \bar{r}\Big] = \bar{r}$$

On the other hand, if we take the variance of this estimate we get

$$\sigma^2_{\text{est}} = \text{var}(\widehat{\bar{r}}) = \text{var}\Big(\frac{1}{N}\sum_{t=1}^{N} r_t\Big) = \frac{1}{N^2}\sum_{t=1}^{N} \text{var}(r_t) = \frac{\sigma^2}{N}$$

In other words, this estimate is governed by its own distribution, $\widehat{\bar{r}} \sim \mathcal{N}(\bar{r}, \sigma^2/N)$.

Going back to the previous example, if $\bar{r} = 1\%$ and $\sigma = 4.33\%$, and we used one year's worth of monthly observations, then the standard deviation of our estimate (also known of the standard error) is

$$\sigma_{\text{est}} = \frac{4.33\%}{\sqrt{12}} = 1.25\%$$

so the standard deviation of our estimated mean is larger than the mean itself. We would need 4 years worth of data to reduce this standard deviation by half.

Alternatively, we could increase the number of data points by switching from monthly to a weekly or daily values, but this would bring us back to our previous problem where we get a worse ratio of

standard deviation to mean value.

Diagram: Parameter uncertainty for daily and yearly values

## 4.6 Estimation of $\sigma$

As we have discussed, the variance and covariance estimates are more accurate than those of expected returns. Our sample variance is

$$s^2 = \frac{1}{N-1}\sum_{t=1}^{N}(r_t - \widehat{\bar{r}})^2$$

As with the estimated expected returns, we also have that $\mathbb{E}[s^2] = \sigma^2$. We can also calculate the variance of the sample variance

$$\text{var}(s^2) = \frac{2\sigma^4}{N-1}$$
$$\text{stdev}(s^2) = \frac{\sqrt{2}\sigma^2}{\sqrt{N-1}}$$

which means that the standard deviation of our estimated variance is $\sqrt{2/(N-1)}$ of the true variance. This is not bad for a reasonably large number of observations $N$.

## 4.7 Robust MVO

So far we have stated that a problem with MVO is the use of estimated expected returns, which tend to be noisy. The Black–Litterman model deals with this issue by removing the estimated expected returns from the optimization problem, and instead uses estimated market equilibrium returns.

Another alternative is to accept that estimated expected returns $\mu \in \mathbb{R}^n$ are noisy, and incorporate this uncertainty to formulate a robust counterpart. The estimated covariance matrix $Q \in \mathbb{R}^{n\times n}$ is generally well-behaved and a good estimate. Moreover, MVO is more sensitive to estimation errors in the expected returns, and these can have an impact an order of magnitude larger than errors in the covariance matrix.

Consider the following problem

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \lambda x^T Q\, x - \mu^T x \\ \text{s.t.} \quad & 1^T x = 1 \\ & x \geq 0 \end{aligned}$$

with the following two estimates of $\mu$ for $n = 3$:

$$\mu_A = \begin{bmatrix} 7.15\% \\ 7.16\% \\ 7\% \end{bmatrix}, \quad \mu_B = \begin{bmatrix} 7.16\% \\ 7.15\% \\ 7\% \end{bmatrix},$$

$$Q = \begin{bmatrix} 0.05 & 0.04 & 0.03 \\ 0.04 & 0.06 & 0.03 \\ 0.03 & 0.03 & 0.07 \end{bmatrix}$$

$$\lambda = 0.02$$

If we optimize this portfolio, we get the following optimal portfolios

| | Portfolio A | Portfolio B |
|---|---|---|
| Asset 1 | 58.3% | 75.0% |
| Asset 2 | 41.7% | 25.0% |
| Asset 3 | 0.0% | 0.0% |

A small change in our estimated parameters can have a significant impact on our optimal portfolio.

Diagram: Feasible region and optimal value instability after adding a constraint

Now consider our typical MVO problem

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & x^T Q\, x \\ \text{s.t.} \quad & \mu^T x \geq R \quad \Big\} \text{ Noisy linear constraint} \\ & 1^T x = 1 \\ & (x_i \geq 0, \quad i = 1,\ ...,\ n) \end{aligned}$$

We can introduce an uncertainty set around the estimated expected returns. We can say that the vector of 'true' expected returns lies within a box uncertainty set

$$\mu_{\text{true}} \in \mathcal{U}(\mu) = \{\mu_{\text{true}} \in \mathbb{R}^n : |\mu_i^{\text{true}} - \mu_i| \leq \delta_i,\ i = 1, ..., n\}$$

where $\mu_{\text{true}}$ is the vector of 'true' (but unknown) expected returns, and $\delta_i$ is the maximum 'distance' we assume exists between our estimate and the 'true' expected returns.

Diagram: Box uncertainty set

A typical way to size this distance is to use the standard error from estimation. As we saw before when studying parameter uncertainty, the standard error of our estimated $\mu$ is $\Theta^{1/2} \in \mathbb{R}^{n \times n}$, where

$$\Theta = \frac{\text{diag}(Q)}{N} \quad \rightarrow \quad (\Theta^{1/2})_{ii} = \frac{\sigma_i}{\sqrt{N}}, \text{ and } (\Theta^{1/2})_{ij} = 0 \text{ for } i \neq j$$

The distances $\delta_i$ can be set proportionally to the standard errors,

$$\delta_i = \varepsilon_1 (\Theta^{1/2})_{ii}$$

where $\varepsilon_1$ is a sizing parameter, usually providing some sort of probabilistic guarantee. If we assume the error follows a normal distribution, we can use the inverse of the cumulative distribution function to determine $\varepsilon_1$. For example, if we wish to have a 95% confidence level that $\mu_{\text{true}}$ lies within our uncertainty set, then

$$\varepsilon_1 = 1.96 \;\Rightarrow\; \delta_i = 1.96 (\Theta^{1/2})_{ii}.$$

Our robust MVO would then look like this

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & x^T Q \; x \\ \text{s.t.} \quad & \mu^T x - \delta^T |x| \geq R \quad \Big\} \text{ Penalize our target return constraint} \\ & 1^T x = 1 \\ & (x_i \geq 0, \quad i = 1, \; ..., \; n) \end{aligned}$$

where $\delta_i = \varepsilon_1 (\Theta^{1/2})_{ii} = \varepsilon_1 \sigma_i / \sqrt{N}$.

The above problem follows the same derivation of the RC LP we saw before. We can replace the absolute value term by adding an auxiliary variable $y \in \mathbb{R}^n$ such that

$$\begin{aligned} & y_i \geq x_i \quad \text{for } i = 1, ...n \\ & y_i \geq -x_i \quad \text{for } i = 1, ...n \\ & \mu^T x - \delta^T y \geq R \end{aligned}$$

## 4.8 Ellipsoidal Uncertainty Set

Realizations in the corners of the box constraints might be too extreme. Instead, we might want to create an ellipsoidal uncertainty set,

$$\mu_{\text{true}} \in \mathcal{U}(\mu) = \{\mu_{\text{true}} \in \mathbb{R}^n : (\mu_{\text{true}} - \mu)^T \Theta^{-1} (\mu_{\text{true}} - \mu) \leq \varepsilon_2^2\}$$

where we have an upper bound on the distance between our estimated expected returns $\mu$ and the 'true' expected returns $\mu_{\text{true}}$.

This measure of distance can be thought of as the linear algebra version of the square of a standard score

$$z_i = \frac{r_i - \mu_i}{s_i} \;\rightarrow\; z_i^2 = \frac{(r_i - \mu_i)^2}{s_i^2} \;\rightarrow (r - \mu)^T \Theta^{-1} (r - \mu).$$

The uncertainty set can be visualized as

Diagram: Ellipsoidal uncertainty set

In this case our $\varepsilon_2^2$ is a measure of maximum distance from our estimated expected returns where we believe the true expected returns will lie, and is related to the confidence level around our estimate.

We can add the corresponding robust penalty $\varepsilon_2 ||\Theta^{1/2} x||_2$ to our target return constraint, where $||\cdot||_2$ is the Euclidean norm of a vector and can also be thought of as a measure of distance. This term is equivalent to

$$\varepsilon_2 ||\Theta^{1/2} x||_2 \quad \iff \quad \varepsilon_2 \sqrt{x^T \Theta x}$$

Finally, we have our robust MVO problem

$$\begin{aligned}
\min_{\boldsymbol{x}} \quad & x^T Q\, x \\
\text{s.t.} \quad & \mu^T x - \varepsilon_2 ||\Theta^{1/2} x||_2 \geq R \quad \Big\} \text{ Penalize our target return constraint} \\
& 1^T x = 1 \\
& (x_i \geq 0, \quad i = 1,\ ...,\ n)
\end{aligned}$$

One way to set the value of $\varepsilon_2$ is to use a confidence region of $\alpha$. For a confidence level of 95%, we have $\alpha = 0.95$,

$$\varepsilon_2^2 = \chi_n^2(\alpha),$$

where $\chi^2_n$ is the inverse cumulative distribution function of the chi-squared distribution with $n$ degrees of freedom (corresponding to the number of assets in our portfolio).

Typically, this is the preferred formulation of the robust MVO problem. Note that, by construction, we do not need to take the absolute value of $x$ since we are concerned with the sum of squared deviations.

## 4.9 Resampling

Portfolio resampling is an alternative 'robust' MVO technique developed with a practitioner's needs in mind by Michaud and Michaud (1998). The concept deviates away from traditional robust optimization techniques, but works well in practice.

Going back to our example, a 0.01% difference between our estimated expected return can have a significant impact on the resulting portfolio. However, what if we accepted this instability, and argue that a "good" fit for both scenarios would be to find the average portfolio weights from both sets of estimates.

Consider the two alternative estimates from our previous example

$$\mu_A = \begin{bmatrix} 7.15\% \\ 7.16\% \\ 7\% \end{bmatrix}, \quad \mu_B = \begin{bmatrix} 7.16\% \\ 7.15\% \\ 7\% \end{bmatrix}, \quad \lambda = 0.02$$

$$Q = \begin{bmatrix} 0.05 & 0.04 & 0.03 \\ 0.04 & 0.06 & 0.03 \\ 0.03 & 0.03 & 0.07 \end{bmatrix}$$

And, we get the following optimal portfolios

| | Portfolio A | Portfolio B | $\rightarrow$ | Portfolio C |
|---|---|---|---|---|
| Asset 1 | 58.3% | 75.0% | | 66.65% |
| Asset 2 | 41.7% | 25.0% | | 33.35% |
| Asset 3 | 0.0% | 0.0% | | 0.0% |

where we average portfolios A and B to get portfolio C, which is partially braced against realizations that would better fit A or B.

Portfolio resampling takes this one step further by computing the average of dozens (or even hundreds) of portfolios. The algorithm to compute this pseudo-robust portfolio is the following

1. Estimate the asset expected returns $\mu \in \mathbb{R}^n$ and covariance matrix $Q \in \mathbb{R}^{n \times n}$.
2. Collect a sample of $T$ observations by drawing randomly generated values from your distribution $r_t \sim \mathcal{N}(\mu, Q)$ for $t = 1, ..., T$, with each $r_t \in \mathbb{R}^n$. Typically, $T \approx 100$.

3. Use your $T$ randomly generated observations to estimate a new set of expected returns $\mu' \in \mathbb{R}^n$ and covariance matrix $Q' \in \mathbb{R}^{n \times n}$. Do not use the factor model when estimating $\mu'$ and $Q'$, instead use the randomly generated data directly.

4. Optimize your portfolio using $\mu'$ and $Q'$ to yield a portfolio $x'$. Save this $x'$.

5. Repeat steps 2 to 4 many times (about 100 times). Calculate the 'robust' portfolio $x$ by taking the average of all your sample portfolios.

The concept of randomly drawing samples and optimizing is akin to Monte Carlo methods, which we will study soon.