

LEIDEN UNIVERSITY

MASTER THESIS

Robust Mean-Variance Optimization

Author:
T.A. DE GRAAF

Supervisors:
Dr. O.W VAN GAANS
(M.I. Leiden)

Dr. M. VAN DER SCHANS
(Ortec Finance)

December 2, 2016



Universiteit
Leiden



LEIDEN UNIVERSITY

Abstract

Robust Mean-Variance Optimization

by T.A. DE GRAAF

Mean-Variance optimization is widely used to find portfolios that make an optimal trade-off between expected return and volatility. The method, however, struggles with a robustness problem since the portfolio weights are very sensitive towards change of the input parameters. There is a vast literature on methods that tries to solve this problem and we discuss two of these methods: resampling and shrinkage. In addition to the methods from the literature, we develop a new method which we call maximum distance optimization.

The resampling method attempts to obtain more robust portfolios by changing the optimization procedure. The shrinkage method attempts to obtain more robust portfolios by making the estimation of the input parameters more robust. The maximum distance optimization method explores a region closely beneath the efficient frontier and determines what kind of portfolios are nearly optimal, but have very different portfolio weights. First, we show that any convex combination of these near-optimal portfolios is also near optimal. Second, we show that the set of near-optimal portfolios is robust. Apart from the robustness, the advantage of this method is that we now obtain a whole scope of solutions, instead of a single portfolio, which Mean-Variance optimization provides. Since the region is robust, the investor or consultant can use his own qualitative arguments to select a preferred portfolio from this region.

Contents

Introduction	1
1 Mean-Variance optimization	3
1.1 Formulation	3
1.2 Analysis of Mean-Variance optimization	5
1.3 Robustness problems	9
1.3.1 The effect of errors in returns, variances, and correlations	9
1.3.1.1 Conclusion	17
2 Resampling	19
2.1 Formulation	20
2.2 Example	22
2.2.1 The diversification effect of resampling	23
2.2.1.1 Conclusion	27
2.2.2 The robustness effect of resampling	27
2.2.2.1 Conclusion	35
2.2.3 Convexity in the resampled frontier	35
2.2.3.1 Conclusion	39
3 Shrinkage	41
3.1 Formulation	41
3.2 Data	42
3.3 Linear shrinkage	43
3.3.1 Constant correlation target	43
3.3.2 The robustness effect of shrinkage	47
3.3.2.1 Conclusion	52
4 Maximum distance optimization	53
4.1 Maximum distance towards convex hull	54
4.2 Support Vector Machines	55
4.3 Formulation of maximum distance optimization	57
4.4 Basin hopping	60
4.4.1 Starting point	60
4.4.2 Number of iterations	61
4.4.3 Stepsize	61
4.4.4 Temperature	61
4.4.5 Local minimization algorithm	62
4.4.6 Basin hopping pseudocode	62
4.5 Example	63
4.5.1 Extra stop criterion	63
4.5.2 Results	64
4.5.3 Convex combination portfolio	70
4.5.4 The diversification effect of MDO	71

4.5.4.1	Conclusion	72
4.5.5	Norms for determining the robustness of MDO	72
4.5.6	The robustness effect of MDO	74
4.5.6.1	Conclusion	78
5	Applications	79
5.1	Shrinkage	79
5.2	Resampling	79
5.2.1	Advice	80
5.3	Maximum distance optimization	80
5.3.1	Advice	81

Introduction

Investors can invest their money in a wide variety of asset classes. All asset classes have their own characteristics and investors want to make smart decisions. They aim to optimize their portfolio by taking into account the historical performances of the asset classes and decide what proportion of their investment-budget they want to invest in which asset class.

In 1952, Harry Markowitz introduced a groundbreaking method regarding portfolio optimization in his paper 'Portfolio Selection' [8]. Despite an initial lack of interest, the ideas he present have come to build foundations on the nowadays widely used *Mean-Variance optimization* method. In 1990, Markowitz, together with William Sharpe and Merton Miller, even received the Nobel Memorial Prize in Economic Sciences for their contribution in the theory of financial economics [9].

Mean-Variance optimization uses expected return (mean) as a measure for profit and expected volatility (variance) as a risk measure. From historical performances, we can estimate the expected risk and the expected return of an asset class, as well as the correlation between the asset classes. Based on these performances, Mean-Variance optimization can be used to construct portfolios that have an optimal balance between required expected return and the level of risk that an investor is willing to take.

There are two disadvantages to Mean-Variance optimization. The most important one is that this method is sensitive towards adjustments of the input. For example, a small change of the expected return of one asset class, can result in a totally different optimal portfolio allocations. Investors do not want to change their investment strategy drastically, due to reasons as liquidity problems and transaction costs. Hence, it is undesirable that slightly different modeling assumptions lead to completely different allocation decisions. In this thesis, we are searching for a way to tackle this problem, that is, a way to make the Mean-Variance optimization method more robust. The second disadvantage is that in Mean-Variance optimization, we can obtain portfolios that are little diversified. The concept of not putting all eggs into one basket is a conventional wisdom that investors like to follow, hence we want to obtain more diversified portfolios.

In this thesis, we discuss three methods that try to solve the robustness problem of Mean-Variance optimization. In Chapter 2, we discuss a method called resampling [10]. In Chapter 3, we discuss a method called shrinkage [6]. In Chapter 4, we develop a new method which we call maximum distance optimization. In Chapter 5, we discuss the applications of these three methods and formulate our advice to Ortec Finance.

Chapter 1

Mean-Variance optimization

Mean-Variance optimization is a widely used method for portfolio optimization. For different risk aversion levels, we obtain portfolios with an optimal tradeoff between expected volatility and expected return. It is known that Mean-Variance optimization struggles with a robustness problem: small changes of the estimated input parameters can result in different optimal portfolio weights. A second problem that we can encounter is that we can obtain optimal portfolios that are little diversified. In this section, we first give a formal definition of Mean-Variance optimization and then we show the robustness and diversification problems that we encounter.

1.1 Formulation

Consider $N \in \mathbb{N}$ asset classes of which we assume that their returns, $\mathbf{r} \in \mathbb{R}^N$, are normally distributed, with mean $\boldsymbol{\mu} \in \mathbb{R}^N$, and covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^N \times \mathbb{R}^N$. We know that $\boldsymbol{\Sigma}$ is semi positive definite by definition. Here, we assume that it is positive definite. In reality, no asset class is an exact linear combination of other asset classes, so this is a reasonable assumption. Let $\mathbf{w} \in \mathbb{R}^N$ denote the *weight vector*, or *portfolio*, then $\mu_0 = \mathbf{w}^\top \boldsymbol{\mu}$ represents the *expected portfolio return*, $\sigma_0^2 = \mathbf{w}^\top \boldsymbol{\Sigma} \mathbf{w}$ represents the *expected portfolio risk*, and $r_0 = \mathbf{w}^\top \mathbf{r}$ represents the *real portfolio return*, which we assume to be normally distributed with $r_0 \sim \mathcal{N}(\mu_0, \sigma_0^2)$. Sometimes the expected portfolio return and the expected portfolio risk are referred to as ‘portfolio return’ and ‘portfolio risk’, or just as ‘return’ and ‘risk’, respectively. Since we are not able to predict the future, the reader knows from the context what is meant. Except for the designations above, we denote the weight vector, \mathbf{w} , also in other ways: we say ‘weights’, or ‘solution’. In the latter case, from the context it is clear that we consider a weight vector that is a solution to Mean-Variance (MV) optimization, or another optimization program.

Unless stated otherwise, we only regard risky assets. We can choose from a wide range of asset classes, all with their own characteristics. Some asset classes have generally a low risk, like German government bonds. However, one usually obtains little reward for taking little risk. Other asset classes, like U.S. equity, have a high volatility, hence, they are considered to be risky. But, in return for taking more risk, one usually gets more return. Often, by investing in different asset classes with comparable risk and return levels, we can reduce the overall portfolio risk. The question is: what asset class should get how much weight, to optimize the return and risk level? For a range of different return and risk levels, the MV optimization method produces corresponding optimal portfolios.

We can choose a level of return that we require and minimize the corresponding level of risk. Or, we can choose the level of risk we are willing to take and maximize the level

of return. A third option is that we choose a level of risk aversion and optimize the corresponding ratio between the risk and return level. We make these different ways of formulating MV optimization explicit in the following definitions. Later in this section, we show under which circumstances these definitions are equivalent.

Let us fix N , r , $\boldsymbol{\mu}$, and $\boldsymbol{\Sigma}$ as at the beginning of this subsection. We look for a weight vector \boldsymbol{w} which is optimal in terms of expected volatility and return.

Definition 1.1.1. The *minimum risk formulation* of MV optimization is given by

$$\min_{\boldsymbol{w} \in \mathbb{R}^N} \boldsymbol{w}^\top \boldsymbol{\Sigma} \boldsymbol{w}, \text{ subject to } \boldsymbol{w}^\top \boldsymbol{\mu} = \mu_0, \boldsymbol{w}^\top \mathbf{1} = 1. \quad (1.1)$$

Definition 1.1.2. The *maximum return formulation* of MV optimization is given by

$$\max_{\boldsymbol{w} \in \mathbb{R}^N} \boldsymbol{w}^\top \boldsymbol{\mu}, \text{ subject to } \boldsymbol{w}^\top \boldsymbol{\Sigma} \boldsymbol{w} = \sigma_0^2, \boldsymbol{w}^\top \mathbf{1} = 1. \quad (1.2)$$

Definition 1.1.3. Let $\lambda \in [0, \infty)$, the *risk aversion formulation* of MV optimization is given by

$$\min_{\boldsymbol{w} \in \mathbb{R}^N} \lambda \boldsymbol{w}^\top \boldsymbol{\Sigma} \boldsymbol{w} - \boldsymbol{w}^\top \boldsymbol{\mu}, \text{ subject to } \boldsymbol{w}^\top \mathbf{1} = 1. \quad (1.3)$$

There is one constraint that we left out of these definitions, but which is in some cases required. This constraint prevents weights from attaining a negative value: $\boldsymbol{w} \geq \mathbf{0}$. In financial terms, a negative weight of an asset class is called *selling short* or *going short*. Short selling is a risky business and forbidden for many investors. The constraint that prevents us from short selling can make the analysis more complex; that is why it is sometimes left out. In the definitions below, the constraint is also left out, but one can add it if required.

The *feasible set* corresponding to the expected portfolio return and risk is the set parameterized by $\{(\sigma_0^2, \mu_0) : \text{there exists } \boldsymbol{w} \in \mathbb{R}^N \text{ such that } \sigma_0^2 = \boldsymbol{w}^\top \boldsymbol{\Sigma} \boldsymbol{w}, \mu_0 = \boldsymbol{w}^\top \boldsymbol{\mu}, \text{ and } \boldsymbol{w}^\top \mathbf{1} = 1\}$. To define the optimal portfolios in an unambiguous way, we first have to look at the solution with minimal variance. Let $\boldsymbol{w}_{\text{GMV}}$ be such that $\boldsymbol{w}_{\text{GMV}} = \arg \min_{\boldsymbol{w} \in \mathbb{R}^N} \boldsymbol{w}^\top \boldsymbol{\Sigma} \boldsymbol{w}$ and $\boldsymbol{w}_{\text{GMV}}^\top \mathbf{1} = 1$, then $\boldsymbol{w}_{\text{GMV}}$ is called the *global minimum variance (GMV) portfolio*. Define $\mu_{\text{GMV}} = \boldsymbol{w}_{\text{GMV}}^\top \boldsymbol{\mu}$ as the *global minimum (expected) return* and $\sigma_{\text{GMV}}^2 = \boldsymbol{w}_{\text{GMV}}^\top \boldsymbol{\Sigma} \boldsymbol{w}_{\text{GMV}}$ as the *global minimum (expected) risk*.

Definition 1.1.4. A weight vector \boldsymbol{w} is called *efficient* or *optimal*, if \boldsymbol{w} is a solution to (1.1) for some $\mu_0 \geq \mu_{\text{GMV}}$, or if it is a solution to (1.2) for some $\sigma_0^2 \geq \sigma_{\text{GMV}}^2$, or if it is a solution to (1.3) for some $\lambda \in [0, \infty)$.

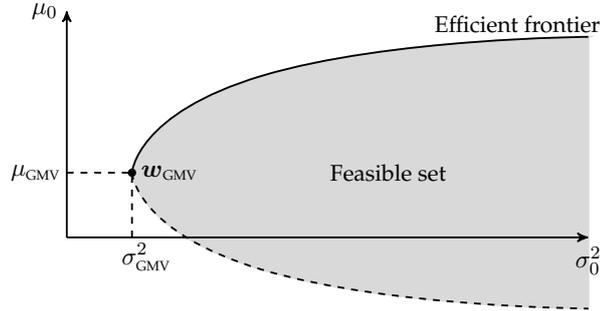
Define $W = \{\boldsymbol{w} : \boldsymbol{w} \text{ is an efficient weight vector}\}$ as the *set of efficient weights*. In Lemma 1.2.7, we show that, regarding only efficient weights, the three formulations, (1.1), (1.2), and (1.3), are equivalent. Hence, the definition of an optimal weight vector is unambiguous.

Definition 1.1.5. The *efficient frontier* is given by

$$\{(\sigma_0^2, \mu_0) : \sigma_0^2 = \boldsymbol{w}^\top \boldsymbol{\Sigma} \boldsymbol{w}, \mu_0 = \boldsymbol{w}^\top \boldsymbol{\mu}, \boldsymbol{w} \in W\}.$$

Figure 1.1 gives a graphical interpretation of the definitions above.

FIGURE 1.1: The efficient frontier, together with the global minimum variance portfolio and the feasible set.



The amount of risk one is willing to take is represented by the risk aversion parameter. If $\lambda = 0$, we do not care about risk at all and we are only interested in maximizing the expected return, $w^\top \mu$. If λ is big, we are more focused on low risk than on high return. By taking λ in a range from zero to infinity, we get optimal portfolios, ranging from μ_{GMV} and σ_{GMV} , to $\max_{n \leq N} \mu_n$ and corresponding risk. It seems reasonable to want a portfolio somewhere in between these extremes. By looking at the efficient frontier and determining what level of risk and return fits our needs, we can figure out what risk aversion parameter suits us. However, if we include other asset classes for example, it might be possible that the same risk aversion level puts us in another risk and/or return position.

1.2 Analysis of Mean-Variance optimization

In this section, we use the minimum risk formulation (1.1). We list a set of lemmas and corollaries that help us understand MV optimization better and that are useful in later analyses. Note that we can invert the covariance matrix Σ , because we assume that it is positive definite and not just semi positive definite.

Lemma 1.2.1. *Assume we are allowed to go short. Let $\mu_0 \in \mathbb{R}_{>\mu_{\text{GMV}}}$. The efficient portfolios w produced by MV optimization satisfy $w = g + h\mu_0$, with*

$$g = \frac{1}{ac - b^2} \Sigma^{-1} [c\mathbf{1} - b\mu]$$

$$h = \frac{1}{ac - b^2} \Sigma^{-1} [a\mu - b\mathbf{1}]$$

$$a = \mathbf{1}^\top \Sigma^{-1} \mathbf{1}$$

$$b = \mathbf{1}^\top \Sigma^{-1} \mu$$

$$c = \mu^\top \Sigma^{-1} \mu.$$

Proof. We use Lagrange multipliers. Define $F(w, \phi_1, \phi_2) = w^\top \Sigma w - \phi_1(w^\top \mu - \mu_0) - \phi_2(w^\top \mathbf{1} - 1)$. Differentiating and setting equal to 0, gives

$$2\Sigma w - \phi_1 \mu - \phi_2 \mathbf{1} = 0$$

$$w^\top \mu - \mu_0 = 0$$

$$w^\top \mathbf{1} - 1 = 0.$$

The first equation can be written as

$$\mathbf{w} = \frac{1}{2} \Sigma^{-1} [\boldsymbol{\mu}, \mathbf{1}] \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix}. \quad (1.4)$$

Using a , b , and c , defined as above, we obtain

$$\begin{aligned} \mu_0 &= \mathbf{w}^\top \boldsymbol{\mu} = \frac{1}{2} \phi_1 c + \frac{1}{2} \phi_2 b \\ 1 &= \mathbf{w}^\top \mathbf{1} = \frac{1}{2} \phi_1 b + \frac{1}{2} \phi_2 a. \end{aligned}$$

Rewrite this as

$$\frac{1}{2} \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} = \begin{bmatrix} c & b \\ b & a \end{bmatrix}^{-1} \begin{bmatrix} \mu_0 \\ 1 \end{bmatrix}.$$

Putting this again in equation (1.4) gives

$$\begin{aligned} \mathbf{w} &= \Sigma^{-1} [\boldsymbol{\mu}, \mathbf{1}] \begin{bmatrix} c & b \\ b & a \end{bmatrix}^{-1} \begin{bmatrix} \mu_0 \\ 1 \end{bmatrix} \\ &= \mathbf{g} + \mathbf{h} \mu_0. \end{aligned} \quad \square$$

Note that the elements in \mathbf{g} sum up to 1 and the elements of \mathbf{h} sum up to zero:

$$\begin{aligned} \mathbf{1}^\top \mathbf{g} &= \frac{1}{ac - b^2} (a \mathbf{1}^\top \Sigma^{-1} \mathbf{1} - b \mathbf{1}^\top \Sigma^{-1} \boldsymbol{\mu}) = 1 \\ \mathbf{1}^\top \mathbf{h} &= \frac{1}{ac - b^2} (a \mathbf{1}^\top \Sigma^{-1} \boldsymbol{\mu} - b \mathbf{1}^\top \Sigma^{-1} \mathbf{1}) = 0. \end{aligned}$$

Also, note that the positive definiteness of Σ implies that $a, c > 0$.

Corollary 1.2.2. *Assume we are allowed to go short. If we only consider efficient portfolios, then the portfolio return, as a function of the portfolio risk, is given by*

$$\mu_0(\sigma_0^2) = \frac{-\mathbf{g}^\top \Sigma \mathbf{h} + \sqrt{(\mathbf{g}^\top \Sigma \mathbf{h})^2 - (\mathbf{h}^\top \Sigma \mathbf{h})(\mathbf{g}^\top \Sigma \mathbf{g} - \sigma_0^2)}}{\mathbf{h}^\top \Sigma \mathbf{h}}.$$

Proof. We have the following functions. The risk as function of the weight vector: $\sigma_0^2(\mathbf{w}) : \mathbb{R}^N \rightarrow \mathbb{R}$, $\mathbf{w} \mapsto \mathbf{w}^\top \Sigma \mathbf{w}$, the return as a function of the weight vector: $\mu_0(\mathbf{w}) : \mathbb{R}^N \rightarrow \mathbb{R}$, $\mathbf{w} \mapsto \mathbf{w}^\top \boldsymbol{\mu}$, and from Lemma 1.2.1, the weight vector as a function of the required return: $\mathbf{w}(\mu_0) : \mathbb{R} \rightarrow \mathbb{R}^N$, $\mu_0 \mapsto \mathbf{g} + \mathbf{h} \mu_0$. Compute $\sigma_0^2(\mathbf{w}(\mu_0)) : \mathbb{R} \rightarrow \mathbb{R}^N \rightarrow \mathbb{R}$, $\mu_0 \mapsto \mathbf{g} + \mathbf{h} \mu_0 \mapsto (\mathbf{g} + \mathbf{h} \mu_0)^\top \Sigma (\mathbf{g} + \mathbf{h} \mu_0)$. We obtain

$$\begin{aligned} \sigma_0^2(\mu_0) &= \mathbf{w}^\top(\mu_0) \Sigma \mathbf{w}(\mu_0) \\ &= \mu_0^2 \mathbf{h}^\top \Sigma \mathbf{h} + \mu_0 (\mathbf{g}^\top \Sigma \mathbf{h} + \mathbf{h}^\top \Sigma \mathbf{g}) + \mathbf{g}^\top \Sigma \mathbf{g}. \end{aligned}$$

Note that this is a convex parabola, since Σ is positive definite. We obtain $\mu_0(\mathbf{w}(\sigma_0^2)) : \mathbb{R} \rightarrow \mathbb{R}^N \rightarrow \mathbb{R}$, $\sigma_0^2 \mapsto \mathbf{g} + \mathbf{h} \mu_0 \mapsto \mu_0$, by inverting the equation above:

$$\mu_0(\sigma_0^2) = \frac{-\mathbf{g}^\top \Sigma \mathbf{h} \pm \sqrt{(\mathbf{g}^\top \Sigma \mathbf{h})^2 - (\mathbf{h}^\top \Sigma \mathbf{h})(\mathbf{g}^\top \Sigma \mathbf{g} - \sigma_0^2)}}{\mathbf{h}^\top \Sigma \mathbf{h}}.$$

We leave out the ‘-’ variant as we are only interested in the highest possible return. \square

Lemma 1.2.3. Σ^{-1} is positive definite.

Proof. Let $\mathbf{z} \in \mathbb{R}^N$, and take $\mathbf{x} = \Sigma^{-1}\mathbf{z}$. Then

$$\mathbf{z}^\top \Sigma^{-1} \mathbf{z} = (\Sigma \Sigma^{-1} \mathbf{z})^\top \Sigma^{-1} \mathbf{z} = (\Sigma^{-1} \mathbf{z})^\top \Sigma^\top \Sigma^{-1} \mathbf{z} = \mathbf{x}^\top \Sigma^\top \mathbf{x} = \mathbf{x}^\top \Sigma \mathbf{x} > 0.$$

□

Lemma 1.2.4. For a, b , and c , defined as in Lemma 1.2.1, we have: $ac - b^2 \geq 0$.

Proof. The positive definite matrix Σ^{-1} generates an inner product: $\langle \mathbf{x}, \mathbf{y} \rangle_{\Sigma^{-1}} = \mathbf{x}^\top \Sigma^{-1} \mathbf{y}$, for $\mathbf{x}, \mathbf{y} \in \mathbb{R}^N$. Use Cauchy-Schwarz to show that

$$b^2 = (\mathbf{1}^\top \Sigma^{-1} \boldsymbol{\mu})^2 = (\langle \mathbf{1}, \boldsymbol{\mu} \rangle_{\Sigma^{-1}})^2 \leq (\|\mathbf{1}\|_{\Sigma^{-1}} \|\boldsymbol{\mu}\|_{\Sigma^{-1}})^2 = (\mathbf{1}^\top \Sigma^{-1} \mathbf{1})(\boldsymbol{\mu}^\top \Sigma^{-1} \boldsymbol{\mu}) = ac.$$

□

Lemma 1.2.5. Assume we are allowed to go short. If we only consider efficient portfolios, then the functions, $\sigma_0^2(\mu_0)$ and $\mu_0(\sigma_0^2)$, in terms of a, b , and c , are given by

$$\begin{aligned} \sigma_0^2(\mu_0) &= \frac{1}{ac - b^2} (a\mu_0^2 - 2b\mu_0 + c) \\ \mu_0(\sigma_0^2) &= \frac{b}{a} + \frac{1}{a} \sqrt{(-b^2 + ac)(a\sigma_0^2 - 1)}. \end{aligned} \quad (1.5)$$

Proof. Multiply equation (1.4) by $[\boldsymbol{\mu}, \mathbf{1}]^\top$:

$$\begin{aligned} \frac{1}{2} \begin{bmatrix} \boldsymbol{\mu}^\top \\ \mathbf{1}^\top \end{bmatrix} \Sigma^{-1} [\boldsymbol{\mu}, \mathbf{1}] \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} &= \begin{bmatrix} \boldsymbol{\mu}^\top \\ \mathbf{1}^\top \end{bmatrix} \mathbf{w} \\ \frac{1}{2} \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} &= \begin{bmatrix} c & b \\ b & a \end{bmatrix}^{-1} \begin{bmatrix} \mu_0 \\ 1 \end{bmatrix} \end{aligned}$$

using

$$\begin{bmatrix} \boldsymbol{\mu}^\top \\ \mathbf{1}^\top \end{bmatrix} \Sigma^{-1} [\boldsymbol{\mu}, \mathbf{1}] = \begin{bmatrix} c & b \\ b & a \end{bmatrix}.$$

Put the attained equation for $[\phi_1, \phi_2]^\top$ back in equation (1.4):

$$\mathbf{w} = \Sigma^{-1} [\boldsymbol{\mu}, \mathbf{1}] \begin{bmatrix} c & b \\ b & a \end{bmatrix}^{-1} \begin{bmatrix} \mu_0 \\ 1 \end{bmatrix}. \quad (1.6)$$

Use $\sigma_0^2 = \mathbf{w}^\top \Sigma \mathbf{w}$ to see that

$$\begin{aligned} \sigma_0^2 &= \left[\Sigma^{-1} [\boldsymbol{\mu}, \mathbf{1}] \begin{bmatrix} c & b \\ b & a \end{bmatrix}^{-1} \begin{bmatrix} \mu_0 \\ 1 \end{bmatrix} \right]^\top \Sigma \left[\Sigma^{-1} [\boldsymbol{\mu}, \mathbf{1}] \begin{bmatrix} c & b \\ b & a \end{bmatrix}^{-1} \begin{bmatrix} \mu_0 \\ 1 \end{bmatrix} \right] \\ &= \frac{1}{ac - b^2} (a\mu_0^2 - 2b\mu_0 + c). \end{aligned}$$

Write μ_0 as a function of σ_0^2 :

$$\mu_0(\sigma_0^2) = \frac{b}{a} \pm \frac{1}{a} \sqrt{(-b^2 + ac)(a\sigma_0^2 - 1)}.$$

Again we leave out the ‘-’ variant, as we want maximum return.

□

Corollary 1.2.6. *Assume we are allowed to go short. We obtain $\sigma_{\text{GMV}}^2 = 1/a$, $\mu_{\text{GMV}} = b/a$, with weight vector $\mathbf{w}_{\text{GMV}} = (1/a)\boldsymbol{\Sigma}^{-1}\mathbf{1}$.*

Proof. An optimum is attained when $d\sigma_0^2(\mu_0)/d\mu_0 = 0$, that is, when $(ac - b^2)/(2a\mu_0 - 2b) = 0$. Use Lemma 1.2.4 to see that we are dealing with a minimum:

$$\frac{d^2\sigma_0^2(\mu_0)}{d\mu_0^2} = \frac{2a}{ac - b^2} > 0.$$

Hence, $\mu_{\text{GMV}} = b/a$ and $\sigma_{\text{GMV}}^2 = 1/a$. From equation (1.6) we obtain:

$$\begin{aligned} \mathbf{w}_{\text{GMV}} &= \frac{1}{ac - b^2} \boldsymbol{\Sigma}^{-1} [\boldsymbol{\mu}, \mathbf{1}] \begin{bmatrix} a & -b \\ -b & c \end{bmatrix} \begin{bmatrix} \frac{b}{a} \\ 1 \end{bmatrix} \\ &= \frac{1}{a} \boldsymbol{\Sigma}^{-1} \mathbf{1}. \end{aligned} \quad \square$$

Lemma 1.2.7. *Assume we are allowed to go short. Formulations, (1.1), (1.2), and (1.3), are equivalent if we consider only efficient portfolios.*

Proof. Let \mathbf{w}_1 and \mathbf{w}_2 be efficient portfolios, with $\mathbf{w}_1 \neq \mathbf{w}_2$. First, we show that we have

$$\mathbf{w}_1^\top \boldsymbol{\Sigma} \mathbf{w}_1 < \mathbf{w}_2^\top \boldsymbol{\Sigma} \mathbf{w}_2 \text{ if and only if } \mathbf{w}_1^\top \boldsymbol{\mu} < \mathbf{w}_2^\top \boldsymbol{\mu}. \quad (1.7)$$

Suppose $\mathbf{w}_1^\top \boldsymbol{\mu} < \mathbf{w}_2^\top \boldsymbol{\mu}$, then function (1.5) implies that $\sigma_0^2(\mathbf{w}_1^\top \boldsymbol{\mu}) < \sigma_0^2(\mathbf{w}_2^\top \boldsymbol{\mu})$. Similarly, when we assume $\mathbf{w}_1^\top \boldsymbol{\Sigma} \mathbf{w}_1 < \mathbf{w}_2^\top \boldsymbol{\Sigma} \mathbf{w}_2$, we get $\mathbf{w}_1^\top \boldsymbol{\mu} < \mathbf{w}_2^\top \boldsymbol{\mu}$.

We now prove the equivalence of the three formulations. Suppose \mathbf{w}_1 is the solution to (1.1), given some $\mu_0 \geq \mu_{\text{GMV}}$. For formulation (1.2), we fix $\sigma_0^2 = \mathbf{w}_1^\top \boldsymbol{\Sigma} \mathbf{w}_1$. We know that $\mathbf{w}_1^\top \mathbf{1} = 1$, hence $\mathbf{w}_1^\top \boldsymbol{\mu} \leq \max_{\mathbf{w}} \mathbf{w}^\top \boldsymbol{\mu}$. Also, according to (1.7), there can not be an efficient portfolio \mathbf{w}_2 , such that $\mathbf{w}_2^\top \boldsymbol{\mu} > \mathbf{w}_1^\top \boldsymbol{\mu} = \mu_0$ and $\mathbf{w}_2^\top \boldsymbol{\Sigma} \mathbf{w}_2 = \sigma_0^2$. Hence, \mathbf{w}_1 is the solution to (1.2), for $\sigma_0^2 = \mathbf{w}_1^\top \boldsymbol{\Sigma} \mathbf{w}_1$.

One can use the same line of reasoning to show that a solution to (1.2), is also a solution to (1.1). What is left to show is that the risk aversion formulation is also equivalent to formulations, (1.1) and (1.2).

Suppose \mathbf{w}_1 is the solution to (1.1) and (1.2), for some $\mu_0 = \mathbf{w}_1^\top \boldsymbol{\mu}$ and $\sigma_0^2 = \mathbf{w}_1^\top \boldsymbol{\Sigma} \mathbf{w}_1$. Using function (1.5), the risk aversion formulation becomes equivalent to minimizing:

$$\lambda \frac{1}{ac - b^2} (a\mu_0^2 - 2b\mu_0 + c) - \mu_0.$$

Differentiate to μ_0 , set it equal to 0, and solve it for λ . We get

$$\lambda = \frac{ac - b^2}{2a\mu_0 - 2b}.$$

Suppose that \mathbf{w}_2 is such that $\lambda \mathbf{w}_2^\top \boldsymbol{\Sigma} \mathbf{w}_2 - \mathbf{w}_2^\top \boldsymbol{\mu} < \lambda \mathbf{w}_1^\top \boldsymbol{\Sigma} \mathbf{w}_1 - \mathbf{w}_1^\top \boldsymbol{\mu}$. Make a distinction between the return of \mathbf{w}_1 and \mathbf{w}_2 , by defining them as $\mu_{0,1}$ and $\mu_{0,2}$, respectively. We get

$$\begin{aligned} \lambda \frac{1}{ac - b^2} (a\mu_{0,2}^2 - 2b\mu_{0,2} + c) - \mu_{0,2} &< \lambda \frac{1}{ac - b^2} (a\mu_{0,1}^2 - 2b\mu_{0,1} + c) - \mu_{0,1} \\ (\mu_{0,2} - \mu_{0,1})^2 &< 0. \end{aligned}$$

This gives a contradiction. Hence, \mathbf{w}_1 is the solution to (1.3) for $\lambda = (ac - b^2)/(2a\mu_0 - 2b)$.

Lastly, let $\lambda \in [0, \infty)$ and suppose w_1 is the solution to (1.3), i.e.,

$$\lambda w_1^\top \Sigma w_1 - w_1^\top \mu \leq \lambda w^\top \Sigma w - w^\top \mu, \text{ for all } w \text{ such that } w^\top \mathbf{1} = 1, w \neq w_1.$$

Let $\mu_0 = w_1^\top \mu$. Suppose that w_2 is such that $w_2^\top \mu = \mu_0$, and $w_2^\top \Sigma w_2 \leq w_1^\top \Sigma w_1$. Then, we get $\lambda w_2^\top \Sigma w_2 - \mu_0 \leq \lambda w_1^\top \Sigma w_1 - \mu_0$, which gives us a contradiction to w_1 being optimal. So w_1 is the solution to (1.1), and therefore also to (1.2). \square

1.3 Robustness problems

In this section, we show what problems we encounter in MV optimization, with regard to robustness. The main reason for wanting the MV method to be more robust is that we do not want to change a portfolio drastically due to small changes in estimated parameters. Suppose we have a current portfolio that is obtained from MV optimization and we have a large share in some asset class A. Suppose there is another asset class, B, that behaves similarly but slightly worse, that is, suppose the correlation with A is high, the risk level is the same, and the expected return is almost as good as that of A. If in some time period B is outperforming A, it is possible that repeating the MV optimization procedure, based on the new performances, results in a totally different portfolio: a small share in A and a large share in B. What should we do in this case? If we change our portfolio, we are forced to accept the transaction costs, hoping that this was just a one time occurrence. Or we can keep our original portfolio, not knowing whether it is still representative or not. Clearly, we want to avoid such a situation. Making the MV procedure more robust will help, because then the sudden change in performance of asset classes has less influence on the outcome. Also, having initially a more diversified portfolio will help, because then we would probably already have split the weights between asset class A and B, to some ratio.

In the analyses below, we only discuss the robustness problem and omit any diversification issues. This is because diversification might help in becoming more robust, as we argue above, but tackling the robustness problem is our number one priority.

1.3.1 The effect of errors in returns, variances, and correlations

We make the robustness problem explicit through an example based on a paper of Chopra and Ziemba [3]. This paper claims that variations of the expected return vector have a much bigger effect than variations in variances or the correlation matrix. Table 1.1 gives an overview of the expected returns and covariance matrix of 10 securities, rounded up to two digits. These securities are: Aluminium Co. of America, American Express Co., Boeing Co., Chevron Co., Coca Cola Co., E.I. Du Pont De Nemours & Co., Minnesota Mining and Manufacturing Co., Procter & Gamble Co., Sears, Roebuck & Co., and United Technologies Co. These securities are randomly selected from 29 Dow Jones Industrial Average securities, of which monthly observations from 1980 till 1989 are available.

Note that we look at one asset class: equity. One can perform the MV optimization method on all different kind of selections. Usually we search for a combination of different, safe and risky, asset classes. But if we want to choose between some selected assets of the same asset class, we can also perform MV optimization on these individual assets.

TABLE 1.1: Expected returns and covariance matrix of 10 securities.

Securities	Covariance matrix										Return
1. Alcoa	77.98	27.34	30.65	12.23	12.05	31.17	26.72	10.83	23.09	36.99	1.56
2. Amex	27.34	71.55	45.68	15.79	25.26	33.87	27.40	22.70	42.09	40.66	1.95
3. Boeing	30.65	45.68	100.8	19.20	25.75	27.97	26.23	21.89	39.24	54.16	1.91
4. Chev	12.23	15.79	19.20	74.33	2.94	21.40	11.66	11.13	11.91	27.00	1.58
5. Coke	12.05	25.26	25.75	2.94	35.86	14.90	16.76	20.10	20.99	21.48	2.16
6. Du Pont	31.17	33.87	27.97	21.40	14.90	47.29	24.67	19.37	31.99	31.87	1.6
7. MMM	26.72	27.40	26.23	11.66	16.76	24.67	33.83	19.80	26.40	28.81	1.49
8. P&G	10.83	22.70	21.89	11.13	20.10	19.37	19.80	31.79	22.62	22.10	1.62
9. Sears	23.09	42.09	39.24	11.91	20.99	31.99	26.40	22.62	64.08	39.70	1.41
10. U Tech	36.99	40.66	54.16	27.00	21.48	31.87	28.81	22.10	39.70	67.45	1.45

Let the range of risk aversion parameters be: $\lambda_k = 10((k - 1)/25 + 1)^{-12}$, $k = 1, \dots, 25$. Figure 1.2 shows the efficient frontier and Figure 1.3 shows the allocations of the optimal portfolios.

FIGURE 1.2: The efficient frontier.

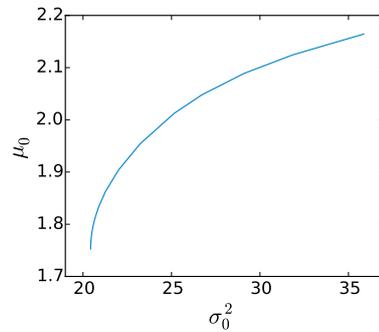
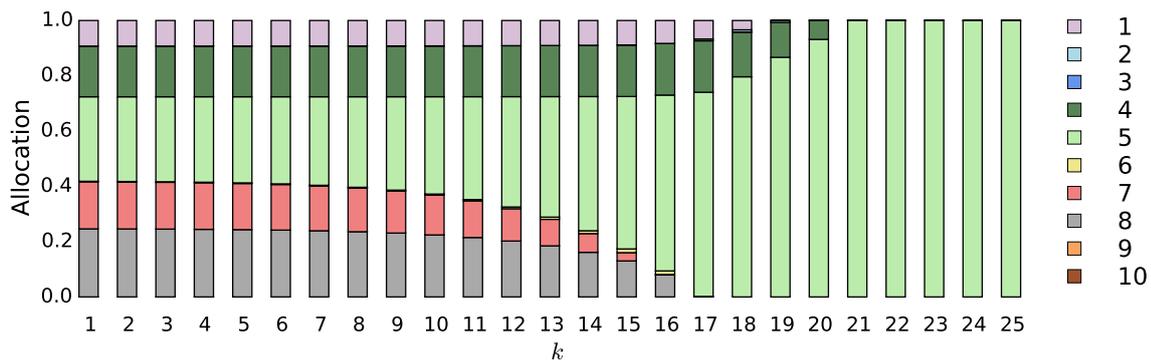


FIGURE 1.3: Allocation of the optimal portfolios corresponding to risk aversion parameter λ_k , $k = 1, \dots, 25$.



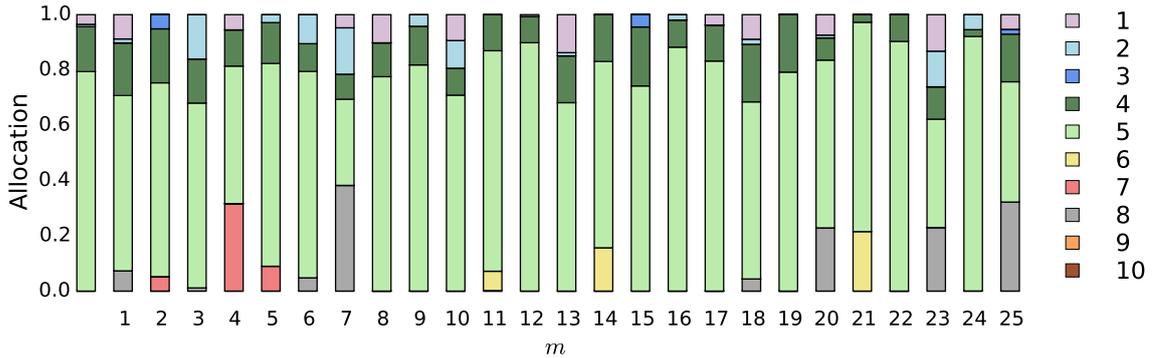
To demonstrate the robustness problem, we adjust the input parameters M times. First, let us investigate what effect adjustments in the expected returns have, later we investigate the effect of adjustments in the variances and adjustments in the correlation matrix. Let $\zeta_n^m \sim \mathcal{N}(0, 1)$, for $n = 1, \dots, N$, $m = 1, \dots, M$, and let $\phi = 0.1$. We adjust the return of security n as follows:

$$\mu_n^{\text{inp}m} = \mu_n(1 + \phi\zeta_n^m).$$

Denote the new return vector by $\boldsymbol{\mu}^{\text{inp}_m} = [\mu_1^{\text{inp}_m}, \dots, \mu_N^{\text{inp}_m}]$. Note that a typical deviation of $\mu_n^{\text{inp}_m}$ from μ_n is a deviation of order ϕ . This is because the standard deviation is 1 and we multiply by a factor of ϕ .

We execute the MV optimization method based on $\boldsymbol{\mu}^{\text{inp}_m}$ and $\boldsymbol{\Sigma}$. Figure 1.4 shows the results. The bar on the far left side represents the original solution, corresponding to $\lambda = 0.02$. Next to it are the 25 solutions, corresponding to the adjusted returns.

FIGURE 1.4: Allocation of the original solution, together with allocations of 25 solutions based on adjusted returns, for $\lambda = 0.02$ and $\phi = 0.1$.



We see that the 25 solutions with modified returns all differ from the original solution to a certain extent. Some are more diversified, like in the first variation, others are less diversified, like in the twelfth variation. In almost all variations, security 5 (Coca Cola Co.) dominates the portfolio. But, for example for variation 7, this is not the case. It seems that the allocations of the seventh variation deviate the most from the original solution.

We want to measure how much this and other portfolios deviate from the original portfolio. Therefore, define \mathbf{w}^{orig} to be the solution corresponding to original MV optimization, based on input parameters, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. Let $\mathbf{w}^{\text{inp}_m}$ be the solution corresponding to the m th variation of input parameters, corresponding to the same risk aversion parameter. Use the Euclidean distance to compare the difference in allocations between portfolios, \mathbf{w}^{orig} and $\mathbf{w}^{\text{inp}_m}$:

$$d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}_m}) = \sqrt{\sum_{n=1}^N (w_n^{\text{orig}} - w_n^{\text{inp}_m})^2}.$$

We go back to analyze Figure 1.4. We calculate the distance between the portfolio corresponding to variation $m = 7$ and the original solution. Rounded to 2 decimals, we obtain:

$$d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}_7}) = 0.64.$$

But what does a distance of 0.64 mean? We know that the maximum distance between two portfolios is $\sqrt{2}$, so we scale the distance to this maximum. We say that a distance of $\sqrt{2}$ is a difference of 100% and a distance of 0 is 0%. We get

$$d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}_7}) = 45\%.$$

This is a big difference. However, for other variations is the difference smaller. Hence, we want to calculate the average distance over all variations. Define

$$\bar{d}(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}}) = \frac{1}{M} \sum_{m=1}^M d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}_m}).$$

We also want to know the standard deviation of these distances:

$$\text{Sd}(d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}})) = \sqrt{\frac{1}{M} \sum_{m=1}^M (d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}_m}) - \bar{d}(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}}))^2}.$$

For these 25 variations we obtain

$$\begin{aligned} \bar{d}(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}}) &= 0.20 \\ \text{Sd}(d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}})) &= 0.15. \end{aligned}$$

Scaled to $\sqrt{2}$, this becomes

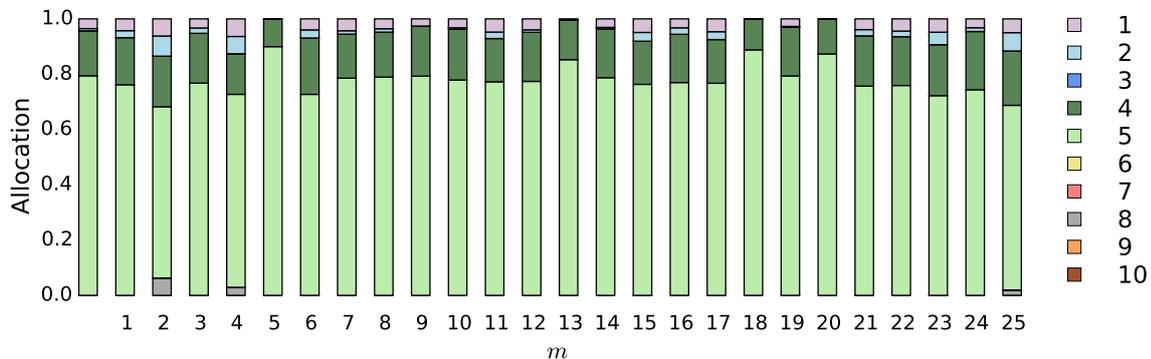
$$\begin{aligned} \bar{d}(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}}) &= 14\% \\ \text{Sd}(d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}})) &= 11\%. \end{aligned}$$

Secondly, let us see what influence adjusting the variances has. We have variance $\sigma_{n,n}^2$ of covariance matrix $\Sigma = (\sigma_{n,n}^2)$. This time, we leave the expected returns and the correlation matrix untouched and solve the MV optimization problem for adjusted variances. Let

$$(\sigma_{n,n}^{\text{inp}_m})^2 = \sigma_{n,n}^2 (1 + \phi \zeta_n^m)$$

for $n = 1, \dots, N$, and $m = 1, \dots, M$. Note that there is a possibility that these adjustments result in an adjusted covariance matrix that is not positive definite. If this is the case, we do not compute the solution, but we again adjust the variances like we do above. Figure 1.5 shows the results for $M = 25$ variations. Recall that $\lambda = 0.02$ and $\phi = 0.1$.

FIGURE 1.5: Allocation of the original solution, together with allocations of 25 solutions based on adjusted variances, for $\lambda = 0.02$ and $\phi = 0.1$.



We see that the allocations corresponding to the variations do not differ a lot from the original solution. Security 5 (Coca Cola Co.) dominates the portfolio in all cases and security 4 (Chevron Co.) is also represented in all variations. It is hard to tell which portfolio deviates the most from the original solution.

It seems that adjusting the variances has less influence than adjusting the returns. However, we can not conclude this right away, since the amount of deviation is influenced by the value of the random parameter, ζ_n^m . If, by coincidence, we have ζ_n^m close to 0 for all m , then Σ^{inp^m} deviates little from Σ . Therefore, it is likely that we obtain solutions that do not deviate much from the original solution. To reduce the influence of ζ_n^m , we should calculate $\bar{d}(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}})$ and $\text{Sd}(d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}}))$, based on a large number of variations. We do a more extensive analysis below, but first, we analyze these 25 variations. Scaled to $\sqrt{2}$, we obtain

$$\begin{aligned}\bar{d}(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}}) &= 4.1\% \\ \text{Sd}(d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}})) &= 3.4\%.\end{aligned}$$

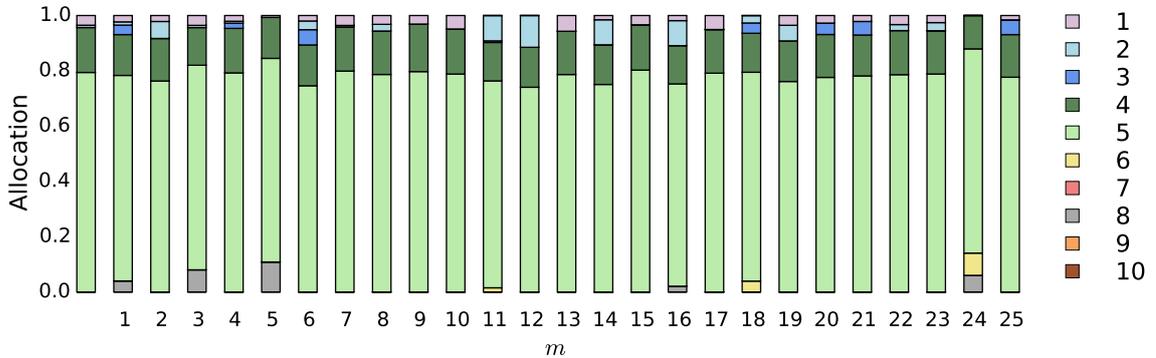
We see that these 25 portfolios, based on variations of the variances, are indeed closer to the original portfolio than the 25 portfolios based on variations of the return vector.

Lastly, let us investigate the influence of changing the correlation matrix. Denote the correlation matrix by $\mathbf{P} = (\rho_{n,n'})$. Let $\zeta_{n,n'}^m \sim \mathcal{N}(0, 1)$, for $n, n' = 1, \dots, N$, and $m = 1, \dots, M$. For $n \neq n'$, let

$$\rho_{n,n'}^{\text{inp}^m} = \rho_{n',n}^{\text{inp}^m} = \rho_{n,n'}(1 + \phi \zeta_{n,n'}^m).$$

Figure 1.6 shows the results. We still use $\lambda = 0.02$ and $\phi = 0.1$.

FIGURE 1.6: Allocation of the original solution, together with allocations of 25 solutions based on adjusted correlation matrices, for $\lambda = 0.02$ and $\phi = 0.1$.



Again, we see that the allocations corresponding to variations of the correlation matrix do not differ a lot from the original solution and that security 5 (Coca Cola Co.) dominates all the portfolios. It seems that these solutions perform similar to the solutions corresponding to adjustments in the variances. Scaled to $\sqrt{2}$, we obtain

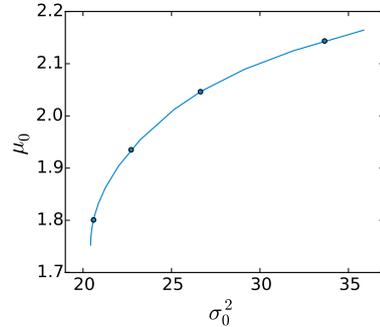
$$\begin{aligned}\bar{d}(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}}) &= 4.5\% \\ \text{Sd}(d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}})) &= 2.3\%.\end{aligned}$$

The difference with the case of variations in variances, is small. It is based on only 25 variations, so we can not draw any conclusions yet.

Now, let us do a more extensive analysis by taking $M = 500$. We compare the solutions based on the original input parameters, μ and Σ , to the solutions based on the modified input parameters. We adjust the input parameters like we do above, by either adjusting the returns, or adjusting the variances, or adjusting the correlation matrix. Let us use four

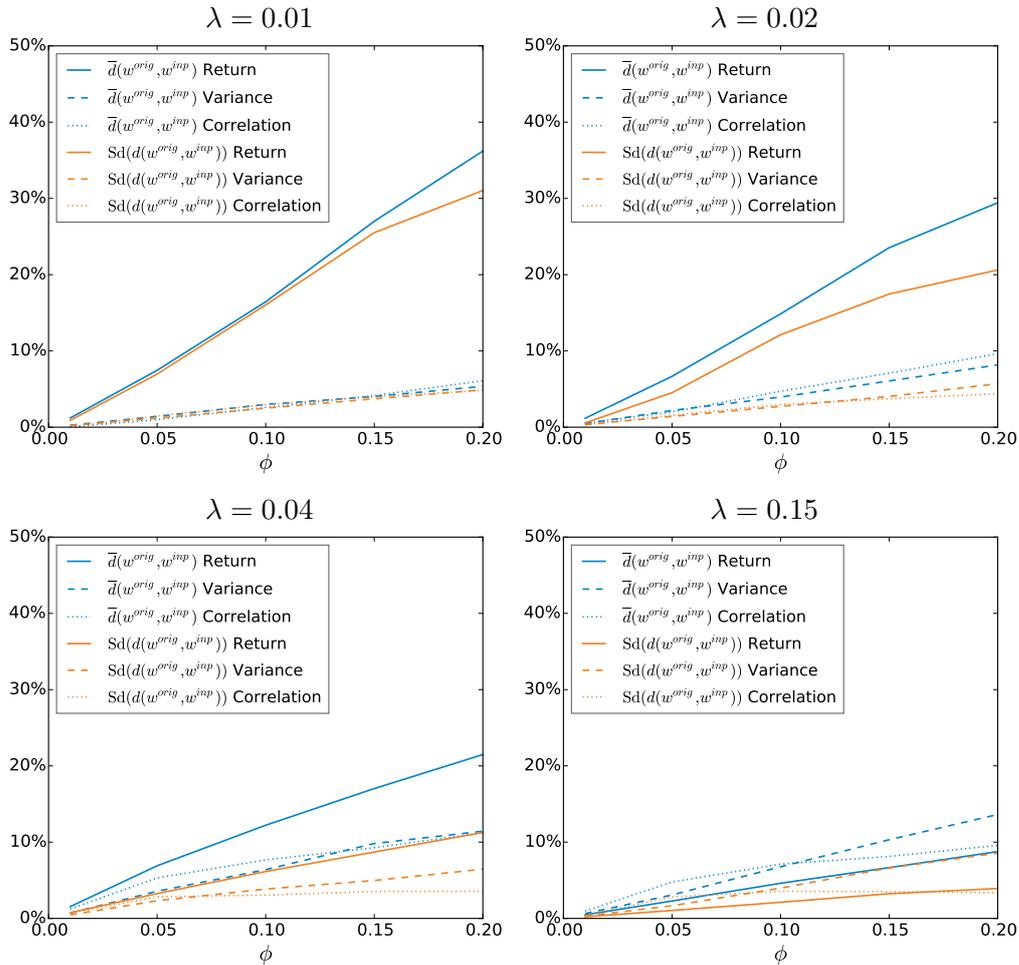
different risk aversion parameters: $\lambda = 0.01, 0.02, 0.04, 0.15$. Figure 1.7 shows that these risk aversion parameters are on different parts of the efficient frontier.

FIGURE 1.7: The efficient frontier, with $\lambda = 0.01, 0.02, 0.04, 0.15$ highlighted.



Not only do we vary the risk aversion parameter, we also want to investigate the influence of different orders of deviation of the input parameters: let $\phi = 0.01, 0.05, 0.1, 0.15, 0.2$. Figure 1.8 shows the results. Again, the average distances and standard deviations are scaled to $\sqrt{2}$.

FIGURE 1.8: The averages and standard deviations of the distances between the original solution and solutions with adjusted input.



We see for all three kind of adjustments that the average distance increases, as the value of ϕ increases. This is as expected, because when we have a large order of deviation of the input parameters, there is a high probability that some asset class suddenly attains favorable features, like a high return in combinations with a low risk. This asset class attains a lot of weight for this iteration and in each iteration this can be another asset class. Hence, we obtain portfolios that are not robust.

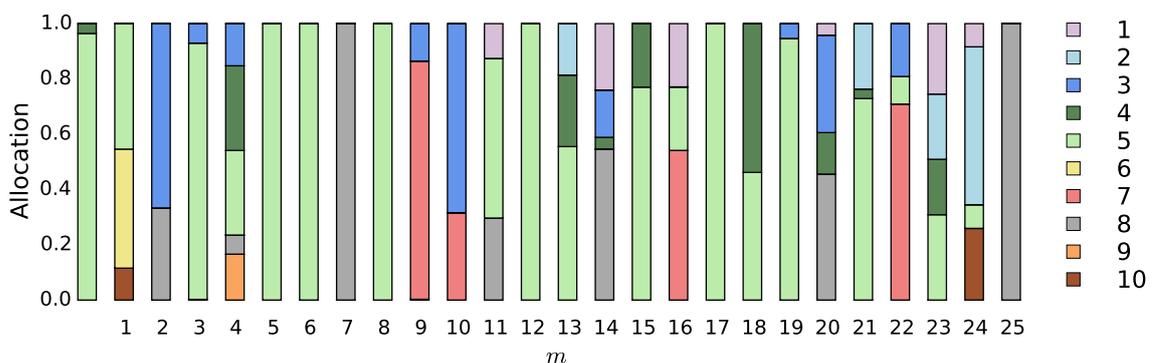
For $\lambda = 0.01$, $\lambda = 0.02$, and $\lambda = 0.04$, we see that $\bar{d}(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{imp}})$, corresponding to adjusted returns is biggest and increases faster than the others. This means that in this case, adjusting the returns has the biggest impact on the allocations. For $\lambda = 0.15$ and $\phi < 0.10$, we see that the average distance of adjusted correlations is slightly bigger than the rest, hence, adjusting the correlations and have the biggest effect here. Furthermore, for $\lambda = 0.15$ and $\phi > 0.11$, we see that the adjusting the variances has the biggest effect on the allocations.

We can make similar observations when we look at the standard deviations. As opposed to the average distance, we see that the standard deviations do not necessarily get smaller, as the value of λ increases.

An important note is that we do not know whether or not the relation between high risk aversion and low average distance, is a causal relation. We only want to point out that the effect can be different for different risk aversion parameters.

Of all three types of adjusting the input parameters we consider, all the risk aversion parameters, and all the values for ϕ , we see that the largest effect is attained for adjusting the returns with $\lambda = 0.01$ and $\phi = 0.2$: the average portfolio deviates about 36%. We demonstrate what kind of portfolios are attained for adjusting the returns with $\lambda = 0.01$ and $\phi = 0.2$. In Figure 1.9 we show the allocations of 25 variations.

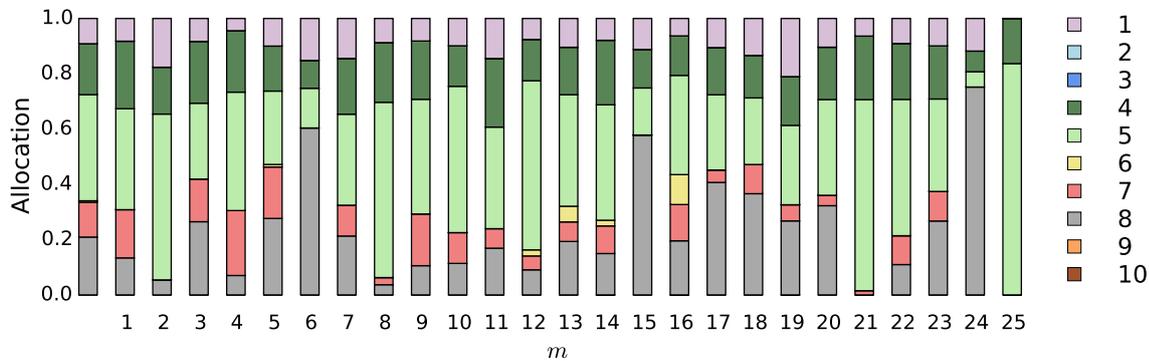
FIGURE 1.9: Allocation of the original solution, together with allocations of 25 solutions based on adjusted returns, for $\lambda = 0.01$ and $\phi = 0.2$.



We see that these portfolios can indeed differ a lot from the original portfolio. Some portfolios are completely different, while others are alike. Imagine that we currently have the original MV optimal portfolio and after some time period we do the MV optimization process again, this time with updated input parameters. Suppose that the portfolio corresponding to variation $m = 2$ is the new MV optimal portfolio. What should we do? This is a unpleasant situation, hence, we see why we want the MV optimization procedure to be more robust.

The effect of adjustments in the variances is biggest for $\lambda = 0.15$ and $\phi = 0.2$: we have an average difference of approximately 14%. Let us look at what this means in terms of allocations. Figure 1.10 shows the results of 25 variations.

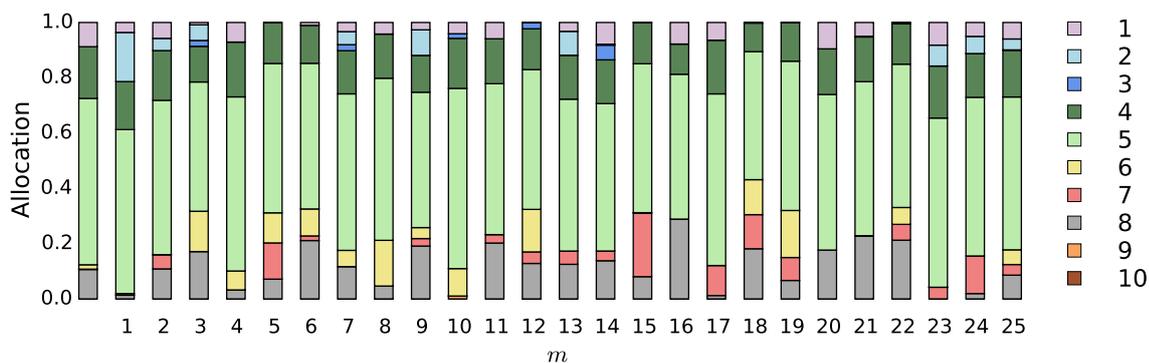
FIGURE 1.10: Allocation of the original solution, together with allocations of 25 solutions based on adjusted variances, for $\lambda = 0.15$ and $\phi = 0.2$.



We see that the overall difference is not big. However, for some variations we do have a portfolio of which the allocations highly differ from the original allocations, like in variation $m = 24$. So, again, we like to avoid this non robustness.

Adjusting the correlations has the most effect for $\lambda = 0.04$ and $\phi = 0.2$: we have an average difference of approximately 11%. Let us look at what this means in terms of the allocations. Figure 1.11 shows the results of 25 variations.

FIGURE 1.11: Allocation of the original solution, together with allocations of 25 solutions based on adjusted correlation matrices, for $\lambda = 0.04$ and $\phi = 0.2$.



We see that these allocations do not differ much. However, a more robust portfolio is still desirable.

Chopra and Ziemba [3] claim in their paper that adjustments in the return vector have much more influence than adjustments in the variances or correlation matrix. Although they use a different way of measuring the effect, we too see this phenomenon when we look at a low risk aversion rate. However, for a high risk aversion level, we see that the influence of adjusting the variances and correlation matrix also can have the most influence. Furthermore, for all three kind of adjustments, we observe different levels of robustness for different risk aversion levels.

Not only use Chopra and Ziemba another measurement tool, but they also look at only one risk aversion parameter: $\lambda = 0.04$. Hence, we can not properly compare our results to theirs. Nevertheless, it has become clear that using adjusted input parameters can result in MV optimal portfolios that highly differ in terms of allocations.

1.3.1.1 Conclusion

A small change in any of the input parameters can result in a big change in allocations of the MV optimal portfolios. For this reason, we search for more robust portfolios.

Although Chopra and Ziemba claim that the effect of adjustments in the returns have the most effect on the allocations, we know this is not always the case. It all depends on the order of deviation of the input parameters and on the value of the risk aversion parameter.

Chapter 2

Resampling

In Section 1.3, we see that MV optimization can produce not only non robust, but also little diversified portfolios. One method that tries to tackle both of these problems is called resampling. Resampling is a well-established method to increase robustness [10] and in the United States, there is a patent on it. The idea behind resampling is based on the error-maximization tendency of the MV procedure. If an estimation error results in favor of a certain asset class, MV optimal solutions tend to put a high weight in that specific asset class. Resampling, however, uses this error-maximization tendency to its advantage.

The resampling procedure works as follows. One repeatedly solves the MV optimization problem for a slight variation of the estimated expected returns and the estimated covariance matrix; such an iteration is called a resampling. Each resampling results in a different portfolio with new weights. We repeat this procedure several times and in the end we average all the weights. The portfolio of averaged weights is called the resampled portfolio. We see that this procedure makes the portfolios more diversified and more robust.

The magnitude of the robustness and diversification effect of resampling depends on our choice of certain parameters, which we see below. However, this choice can come at a prize. We have to choose a position in a tradeoff between robustness and diversification on the one side and expected portfolio performance on the other side. In our analysis, we extensively discuss what position on this tradeoff we should take.

The diversification effect of resampling follows from the error maximization tendency of MV optimization [10]. With each resampling, we do the MV optimization procedure based on a variation of input parameters. This variation of input parameters determines which asset class attains favorable features in terms of expected return, expected risk, and correlations with other asset classes. Because of this error maximization tendency, in each resampling there is a chance that one of the asset classes attains a lot of weight; a so called 'lucky draw' for this asset class. Each resampling has a different variation of input parameters, so each asset class repeatedly gets a chance to attain a lot of weight. Hence, by averaging the solutions of all the resamplings, it is likely that all asset classes appear in the resampled portfolio to a certain extent. Note that for a certain asset class, the amount of occurrences of a lucky draw depends on the original input parameters. If initially an asset class has favorable features (like for example, a high expected return in combination with a low expected risk), it is more likely that it also attains favorable features in a single variation. The reason for this lies in construction of the resampling procedure, which we see below.

The robustness effect of resampling follows from the diversification effect of resampling. Suppose we choose the robustness and diversification effect over the expected portfolio

performance, then the resampled portfolio that we attain is well diversified, due to reasons we explain above. Suppose it turns out that the original input parameters are not correct and we do the resampling procedure again, this time with the new, better input parameters, then, we have a high probability that the newly attained portfolio is also well diversified (provided that we remain to be on the robustness and diversification side of the tradeoff). We now have two well diversified portfolios: one based on the old input parameters and one based on the new input parameters. We see that the difference between the two portfolios is small and thus is the resampled portfolio, based on the initial input, robust.

The last important property of resampling is that the resampled frontier, i.e., the frontier obtained by doing the resampling procedure for different risk/return levels, can contain convex parts. In Theorem 2.2.2, we show that for a simplified version of the resampling procedure, the probability of convex parts in the resampled frontier converges to zero, as the number of resamplings goes to infinity.

2.1 Formulation

Unless stated otherwise, we assume that we are not allowed to go short. We define the resampling procedure for the risk aversion formulation; this means that we average the solutions that correspond to the same risk aversion parameter. One could also choose to do the resampling procedure based on the minimum risk formulation. That is, do the resampling procedure and average the solutions that correspond to the same required return. Or, one could choose to do the resampling procedure based on the maximum return formulation, i.e., do the resampling procedure and average the solutions that correspond to the same level of risk.

We argue that doing the resampling procedure based on the risk aversion formulation is the best choice. To see this, consider the case in which we average per required expected risk and our required expected risk, σ_0^2 , is as small as possible. Then, we have the possibility that in a resampling, the variation of input parameters is such that an expected portfolio risk of σ_0^2 can not be attained. When we average all solutions of the resamplings, we are forced to include portfolios with a higher expected risk level, which initially was not our intention.

Suppose we do the resampling procedure for some fixed expected required return (not necessarily the smallest possible) and suppose that for some resampling, the variation of input parameters is such that the obtained expected portfolio risk is large. Then, maybe we are not willing to take such a big risk but we want to adjust the expected return we require. This is not possible when we average per fixed expected required return. However, when we average per risk aversion parameter, the return and risk level automatically adjusts to the correct equilibrium. Hence, the portfolios we average are all portfolios that fit our risk aversion appetite.

For risk aversion parameter $\lambda \geq 0$, recall from Section 1.1 the risk aversion formulation of MV optimization:

$$\min_w \lambda w^\top \Sigma w - w^\top \mu, \text{ subject to } w^\top \mathbf{1} = 1, w \geq \mathbf{0}.$$

Definition 2.1.1. We define the *resampling procedure* as follows. We draw $T \in \mathbb{N}$ times from a multivariate normal distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, resulting in a return matrix with $n = 1, \dots, N$ asset classes and $t = 1, \dots, T$ data points. We repeat this $I \in \mathbb{N}$ times, obtaining return matrices: $\mathbf{X}_1 = (\chi_{n,t,1}), \dots, \mathbf{X}_I = (\chi_{n,t,I})$. In each iteration $i = 1, \dots, I$, we estimate our expected return vector $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}_i = (\sigma_{n,n',i}), n, n' = 1, \dots, N$, as follows:

$$\boldsymbol{\mu}_i^\top = [\mu_{1,i}, \dots, \mu_{N,i}] = \left[\frac{1}{T} \sum_{t=1}^T \chi_{1,t,i}, \dots, \frac{1}{T} \sum_{t=1}^T \chi_{N,t,i} \right]$$

$$\boldsymbol{\Sigma}_i = (\sigma_{n,n',i}), \text{ with } \sigma_{n,n',i} = \frac{1}{T} \sum_{t=1}^T (\chi_{n,t,i} - \mu_{n,i})(\chi_{n',t,i} - \mu_{n',i}).$$

For each variation of input parameters, $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$, we do the MV optimization procedure for risk aversion parameter $\lambda \geq 0$. This results in weights

$$\boldsymbol{w}_{i,\lambda}^\top = [w_{1,i,\lambda}, \dots, w_{N,i,\lambda}].$$

We call an iteration i in the resampling procedure, a *resampling*. So for each resampling i , we obtain weight $w_{n,i,\lambda}$ for asset class n and risk aversion parameter λ .

Definition 2.1.2. Consider the setting of Definition 2.1.1. For risk aversion parameter λ and asset class n , let $\bar{w}_{n,\lambda}$ denote the average of the weights that we obtain in the resamplings, i.e.,

$$\bar{w}_{n,\lambda} = \frac{1}{I} \sum_{i=1}^I w_{n,i,\lambda}.$$

The *resampled portfolio* or *resampled solution* for risk aversion parameter λ , is given by

$$\bar{\boldsymbol{w}}_\lambda^\top = [\bar{w}_{1,\lambda}, \dots, \bar{w}_{N,\lambda}].$$

Definition 2.1.3. Let \bar{W} be the set of resampled solutions, for different risk aversion parameters:

$$\bar{W} = \{\bar{\boldsymbol{w}}_\lambda : \bar{\boldsymbol{w}}_\lambda^\top = [\bar{w}_{1,\lambda}, \dots, \bar{w}_{N,\lambda}], \lambda \geq 0\}.$$

Then the *resampled frontier* is parameterized by:

$$\{(\sigma_0^2, \mu_0) : \sigma_0^2 = \bar{\boldsymbol{w}}_\lambda^\top \boldsymbol{\Sigma} \bar{\boldsymbol{w}}_\lambda, \mu_0 = \bar{\boldsymbol{w}}_\lambda^\top \boldsymbol{\mu}, \bar{\boldsymbol{w}}_\lambda \in \bar{W}, \lambda \geq 0\}.$$

Note that the weights of a resampled portfolio sum up to one:

$$\sum_{n=1}^N \bar{w}_{n,\lambda} = \sum_{n=1}^N \left(\frac{1}{I} \sum_{i=1}^I w_{n,i,\lambda} \right) = \frac{1}{I} \sum_{i=1}^I \left(\sum_{n=1}^N w_{n,i,\lambda} \right) = 1$$

for $\lambda \geq 0$.

To investigate the resampling procedure, we work out an example. The data we use is based on a paper of Scherer (2002) [10] on portfolio resampling.

2.2 Example

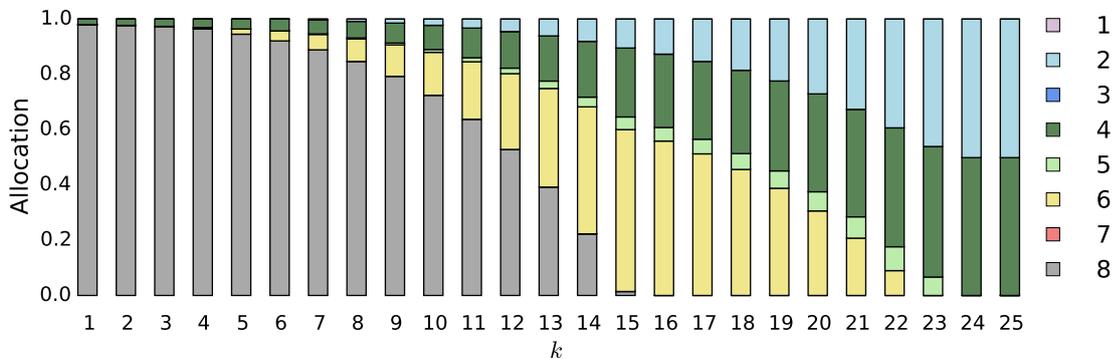
Suppose we have $N = 8$ asset classes. The estimated expected returns and covariance matrix of these asset classes are given in Table 2.1 [10].

TABLE 2.1: Overview of input parameters, μ and Σ .

Asset classes	Covariance matrix								Means
Equity									
1. Canadian	30.25	15.85	10.26	9.68	19.17	16.79	2.87	2.83	0.39
2. French	15.85	49.42	27.11	20.79	22.82	13.30	3.11	2.85	0.88
3. German	10.26	27.11	38.69	15.33	17.94	9.10	3.38	2.72	0.53
4. Japanese	9.68	20.79	15.33	49.56	16.92	6.66	1.98	1.76	0.88
5. U.K.	19.17	22.82	17.94	16.92	36.12	14.47	3.02	2.72	0.79
6. U.S.	16.79	13.30	9.10	6.66	14.47	18.49	3.11	2.82	0.71
Bonds									
7. U.S.	2.87	3.11	3.38	1.98	3.02	3.11	4.04	2.88	0.25
8. European	2.83	2.85	2.72	1.76	2.72	2.82	2.88	2.43	0.27

We use $K = 25$ risk aversion parameters: $\lambda_k = ((k-1)/25+1)^{-9}$, $k = 1, \dots, 25$. We choose these risk aversion parameters to capture the range from minimum risk to maximum return. Figure 2.1 shows the allocation of the MV optimal portfolios per risk aversion parameter. In the legend, '1' represents the first asset class, Canadian Equity, and so on.

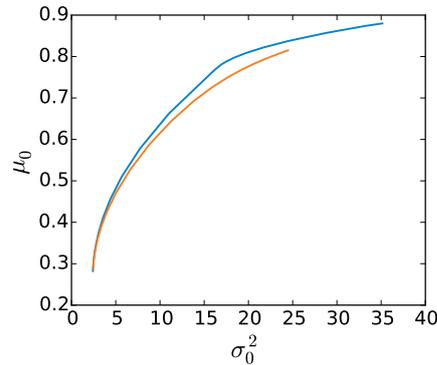
FIGURE 2.1: Asset allocation per risk aversion parameter numbered by $k = 1, \dots, 25$.



For a high risk aversion rate, the least risky asset class dominates the portfolio. For a low risk aversion rate, there are two asset classes that take the largest share, because they share the highest expected return. A total of five asset classes appear somewhere in the range of solutions, the other asset classes do not appear. Apparently, the latter asset classes have unfavorable features.

For the resampling procedure, we choose parameters, $I = 200$ and $T = 200$. Figure 2.2 shows the original efficient frontier (blue), together with the resampled frontier (orange). We see that the resampled frontier is close to the original efficient frontier. Note that the maximum required return corresponding to the resampling procedure is lower than the original maximum required return. This is caused by the diversification effect of resampling, as we see below in Section 2.2.1.

FIGURE 2.2: The original efficient frontier (blue) and the resampled frontier (orange), with $T = 200$ and $I = 200$.



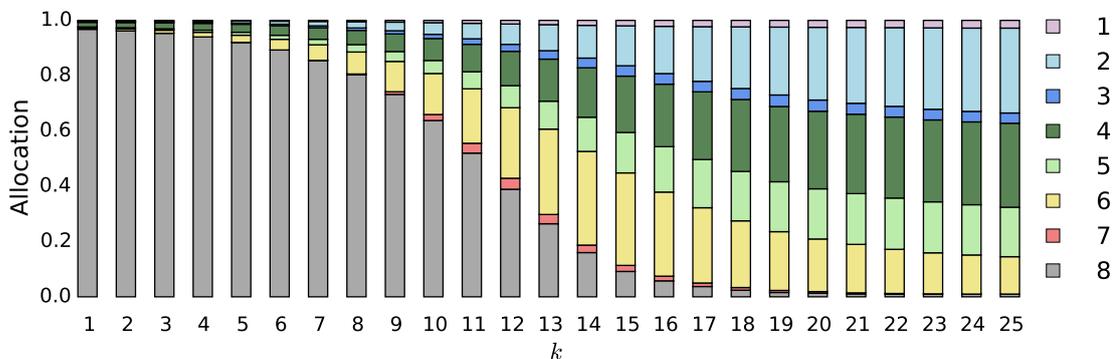
We investigate for this example two key questions.

- Is a resampled portfolio more diversified?
- Is a resampled portfolio more robust?

2.2.1 The diversification effect of resampling

To answer the first question, we plot the allocations of the resampled solutions in Figure 2.3. We use again risk aversion parameters $\lambda_k = ((k - 1)/25 + 1)^{-9}$, $k = 1, \dots, 25$.

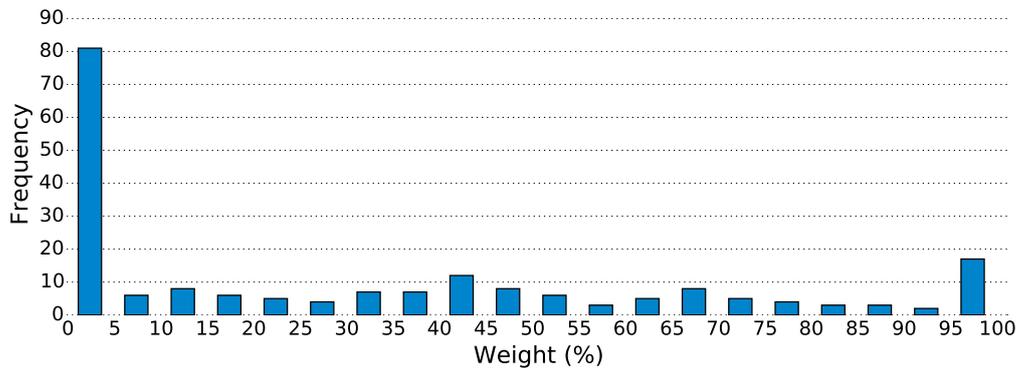
FIGURE 2.3: Asset allocation of the resampled portfolios per risk aversion parameter numbered by $k = 1, \dots, 25$, with $T = 200$ and $I = 200$.



We have two criteria to judge about the extent to which a portfolio is diversified. The first is the number of asset classes that appear in the portfolio; the more asset classes, the more diversified the portfolio. The second is the extent to which the weights of the asset classes that appear in the portfolio are equally divided; more equally divided means more diversified. Here, we see that for some risk aversion parameters, all asset classes appear in the solution, but the second criteria is never completely met.

The reason that all asset classes appear in the solution, lies in the foundation of the resampling procedure, as we describe above as 'lucky draws'. Let us investigate the effect of these lucky draws for this example. In Figure 2.4, we compute the distribution of the weights that U.S. Equity obtains in the resamplings, corresponding to risk aversion parameter λ_{17} .

FIGURE 2.4: Distribution of the weights of U.S. Equity, for $I = 200$ resamplings, λ_{17} , and $T = 200$.



We obtain $\bar{w}_{6,\lambda_{17}} = 0.3066$, rounded up to 4 digits, and we see that this average is attained by a wide variety of weights. In most of the resamplings, the weight ranges from 0% to 5%, but in some other resamplings we obtain a weight of 95% till 100%. Altogether, we can conclude that the average of 0.3066 is attained under influence of lucky draws. Note that we say we have a lucky draw when we have ‘a lot of weight’ of a certain asset class, hence, having a weight of 95% till 100% is not the only time a lucky draw occurs. We consider a weight of 50%, for example, also as a lucky draw.

It is not guaranteed that all asset classes appear somewhere in the range of resampled solutions. If a certain asset class has unfavorable features, the probability of that asset class gaining favorable features in a resampling might be so small, that it either does not happen at all, or it does not happen enough. In the latter case, the few times that it does happen gets averaged out by the great number of resamplings. In that case, it has such a small weight that we consider it to be negligible.

We reevaluate Figure 2.3. We see that for a high risk aversion rate, the only asset class in the race is European Bonds. The other asset classes have an expected risk that is too high, an expected return that is too low, and/or unfortunate correlations. In a resampling, the probability of another asset class gaining a lot of weight is therefore too small. So the diversification effect is low, because there are not enough asset classes with characteristics that are comparable to those of European Bonds.

For a low risk aversion rate, we see that the resampled portfolios are more diversified than the original MV portfolios, especially for risk aversion parameter λ_{12} and higher. At this risk aversion rate, more asset classes have comparable characteristics, apparently. Hence, asset classes that did not appear in the original MV portfolios, do appear in the resampled portfolios. We see that the effect of lucky draws result in more diversified portfolios.

The extent to which resampling has a diversification effect depends on the parameter T . If we choose $T = 20000$, we attain resampling parameters, μ_i and Σ_i , that are close to the original input parameters, μ and Σ . Hence, the resampled portfolios are close to the original portfolios. Figure 2.5 shows the results for $T = 20000$ and $I = 200$. The distribution of the obtained weights of 200 resamplings is given in Figure 2.6. We use the same risk aversion parameters as above.

FIGURE 2.5: Asset allocation of the resampled portfolios per risk aversion parameter numbered by $k = 1, \dots, 25$, with $T = 20000$ and $I = 200$.

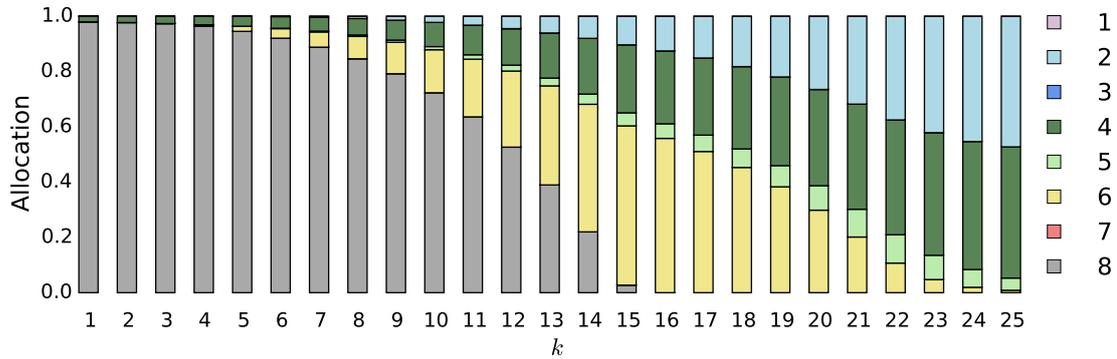
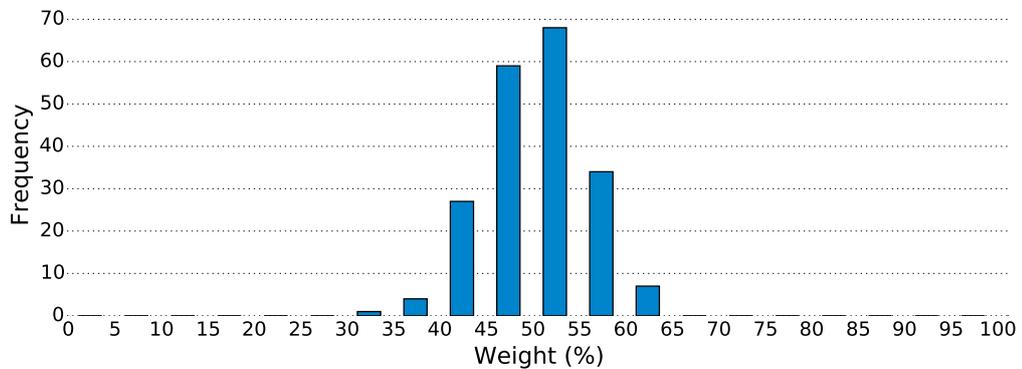


FIGURE 2.6: Distribution of the weights of U.S. Equity, for $I = 200$ resamplings, λ_{17} , and $T = 20000$.



When we compare Figure 2.5 to Figure 2.1, we see that these resampled solutions indeed do not differ much from the original MV solutions. Furthermore, when we compare Figure 2.6 to Figure 2.4, we see that the distribution corresponding to $T = 20000$ is more centered than in the case that $T = 200$. As we argue above, for $T = 20000$, the input parameters of the resamplings are close to the original input parameters. Hence, the probability that a lucky draw occurs for an asset class that originally does not appear, is small. This explains why the diversification effect is low for $T = 20000$.

Another extreme case is when we use $T = 2$. Figure 2.7 and Figure 2.8 show the results.

FIGURE 2.7: Asset allocation of the resampled portfolios per risk aversion parameter numbered by $k = 1, \dots, 25$, with $T = 2$ and $I = 200$.

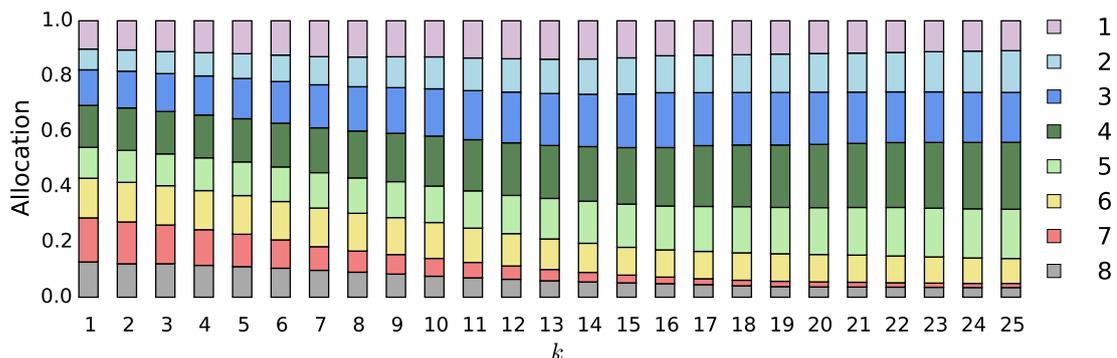
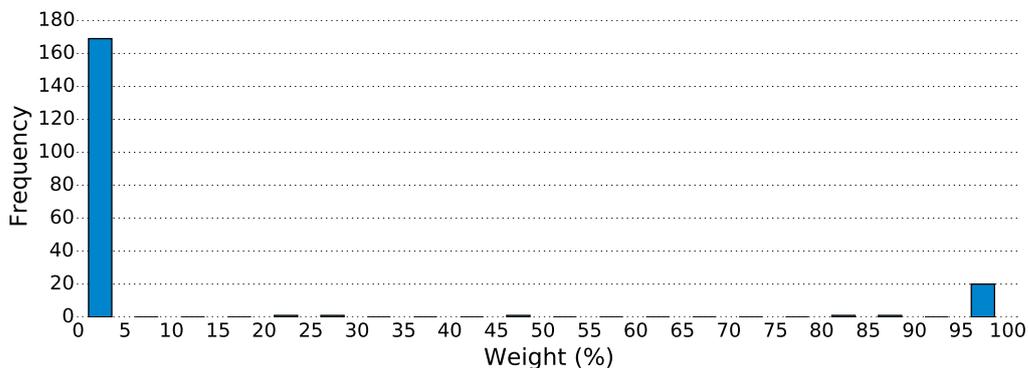


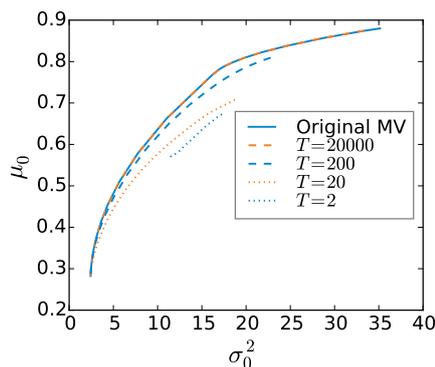
FIGURE 2.8: Distribution of the weights of U.S. Equity, for $I = 200$ resamplings, λ_{17} , and $T = 2$.



By drawing only two times from $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, we lose a lot of the initial structure. In each resampling, either only one or just a small number asset classes attains favorable characteristics and therefore will be chosen above the rest. Figure 2.8 shows this phenomenon. In almost each resampling, U.S. Equity attains an extreme weight, ultimately resulting in an average weight of 0.0989. Hence, resampling many times results in portfolio weights that are almost perfectly diversified, as we see in Figure 2.7.

When we use resampling, we have to make a decision about the value of T and I . We explain in Section 2.2.3 why we want I as big as possible. But what is a smart choice for T ? As we see above, a big value of T results in a small diversification effect and a small value of T results in a big diversification effect. Let us investigate the influence of parameter T on the expected portfolio risk and return. Figure 2.9 shows the efficient frontier of the original MV procedure and the resampled frontiers corresponding to $T = 20000$, $T = 200$, $T = 20$, and $T = 2$. All resampled frontiers use $I = 200$ and λ_k as above.

FIGURE 2.9: The original efficient frontier, together with resampled frontiers for different values of T and $I = 200$.



For $T = 20000$, we see that the original and the resampled frontier are almost identical. This is not surprising, since the obtained original and resampled portfolios are also almost identical. The resampled frontier for $T = 200$ is close to the original efficient frontier, as we already know from Figure 2.2. For $T = 20$ and $T = 2$, the difference between the resampled and original frontier is larger. Furthermore, for $T = 200$, $T = 20$, and $T = 2$ is the range of risk and return levels limited. We explain this by looking at the case of $T = 2$. From Figure 2.7 we see that the portfolios for $k = 1$ and $k = 25$, are alike, hence, the difference in expected portfolio risk and return between the portfolios is small.

2.2.1.1 Conclusion

Resampling has a positive effect on diversification, provided that there are asset classes with comparable characteristics. We have to choose a position in a tradeoff between diversification on the one side and expected portfolio performance on the other side. We choose a position by picking a value for T .

A small value of T means a larger diversification effect, but a lower expected portfolio return, and/or a higher expected portfolio risk. It also results in a limited range of possible expected portfolio risks and returns. Furthermore, by choosing small T , in each resampling one throws away a lot of information that can be important. We can not predict the future, so we can not say for certain that small T results in portfolios that perform poorly, but common sense tells us that we should not throw away too much information in exchange for a diversification effect.

A big value of T means less diversification effect, but a higher expected portfolio return and/or a lower expected portfolio risk. Here, the range of possible portfolio risks and returns is only a little bit more limited. Lastly, opposed to the case of small T , a lot of the structure of the initial input parameters remains intact.

2.2.2 The robustness effect of resampling

To investigate the robustness effect, we modify the input parameters and compare the influence on asset allocation of the original MV optimization procedure versus the influence on asset allocation of the resampling procedure.

In this section, we use a different, less arbitrary method of modifying the input parameters than we do in Section 1.3.1. Here, we use a procedure that we also use in resampling: we draw a number of times, say T_{inp} , from a multivariate normal distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and compute the sample means and the sample covariance matrix. These parameters are our new, modified input parameters on which we execute the MV optimization and the resampling procedure again. Using this method, we are able to decide about the extend to which we keep the structure of the initial input parameters, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. This extend depends on the value of T_{inp} : a large value for T_{inp} means that we keep a lot of the initial structure intact and small T_{inp} means that we lose a lot of the initial structure. Hence, this method gives us the power to determine the level of deviation of the initial input parameters. In the method of Section 1.3.1, we modify our parameters individually per entry, using a random factor. This results in less grip on the overall order of deviation of the parameters.

Let us formulate the procedure of obtaining modified input parameters. We draw T_{inp} times from $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Let M denote the total number of times that we obtain modified input parameters. For $m = 1, \dots, M$, $n, n' = 1, \dots, N$, $t = 1, \dots, T_{\text{inp}}$, we get return vector $\mathbf{X}^{\text{inp}_m} = (\chi_{n,t}^{\text{inp}_m})$ and modified input parameters:

$$\boldsymbol{\mu}^{\text{inp}_m \top} = [\mu_1^{\text{inp}_m}, \dots, \mu_N^{\text{inp}_m}] = \left[\frac{1}{T_{\text{inp}}} \sum_{t=1}^{T_{\text{inp}}} \chi_{1,t}^{\text{inp}_m}, \dots, \frac{1}{T_{\text{inp}}} \sum_{t=1}^{T_{\text{inp}}} \chi_{N,t}^{\text{inp}_m} \right]$$

$$\boldsymbol{\Sigma}^{\text{inp}_m} = (\sigma_{n,n'}^{\text{inp}_m}), \text{ with } \sigma_{n,n'}^{\text{inp}_m} = \frac{1}{T_{\text{inp}}} \sum_{t=1}^{T_{\text{inp}}} (\chi_{n,t}^{\text{inp}_m} - \mu_n^{\text{inp}_m})(\chi_{n',t}^{\text{inp}_m} - \mu_{n'}^{\text{inp}_m}).$$

Based on the original and modified input parameters, we execute the MV optimization method and the resampling method. Note that in a resampling $i = 1, \dots, I$, we again draw a number of times from a normal distribution: T times from $\mathcal{N}(\boldsymbol{\mu}^{\text{inp}_m}, \boldsymbol{\Sigma}^{\text{inp}_m})$. This results in a return vector $\mathbf{X}_i^{\text{inp}_m} = (\chi_{n,t,i}^m)$, $t = 1, \dots, T$. For variation m and resampling i , we obtain parameters:

$$\boldsymbol{\mu}_i^{\text{inp}_m \top} = [\mu_{1,i}^{\text{inp}_m}, \dots, \mu_{N,i}^{\text{inp}_m}] = \left[\frac{1}{T} \sum_{t=1}^T \chi_{1,t,i}^m, \dots, \frac{1}{T} \sum_{t=1}^T \chi_{N,t,i}^m \right]$$

$$\boldsymbol{\Sigma}_i^{\text{inp}_m} = (\sigma_{n,n',i}^{\text{inp}_m}), \text{ with } \sigma_{n,n',i}^{\text{inp}_m} = \frac{1}{T} \sum_{t=1}^T (\chi_{n,t,i}^m - \mu_{n,i}^{\text{inp}_m})(\chi_{n',t,i}^m - \mu_{n',i}^{\text{inp}_m}).$$

Define the solution corresponding to the original input parameters and the original MV method as the *original MV solution* or *original MV portfolio* and denote it by \mathbf{w}^{orig} . Define the solution corresponding to the original input parameters and the resampling method as the *original resampled solution* or *original resampled portfolio* and denote it by \mathbf{w}^{res} . Furthermore, suppose we look at variation $m = 1, \dots, M$. Let $\mathbf{w}^{\text{inp}_m \text{orig}}$ denote the solution of the original MV procedure, based on input parameters, $\boldsymbol{\mu}^{\text{inp}_m}$ and $\boldsymbol{\Sigma}^{\text{inp}_m}$. Let $\mathbf{w}^{\text{inp}_m \text{res}}$ denote the solution of the resampling procedure, also based on input parameters, $\boldsymbol{\mu}^{\text{inp}_m}$ and $\boldsymbol{\Sigma}^{\text{inp}_m}$. Table 2.2 gives a notational overview of the input parameters and corresponding solutions.

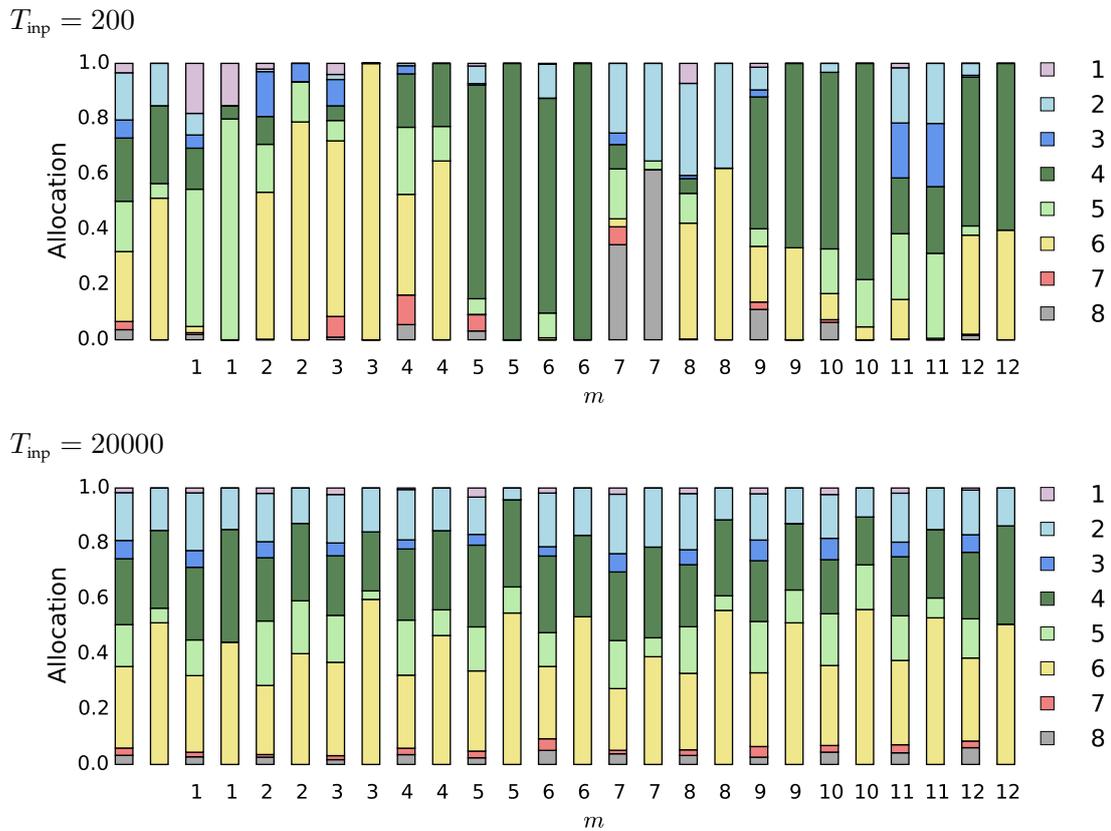
TABLE 2.2: Notational overview.

	MV optimization		Resampling	
	Input	Solution	Input	Solution
Original	$\boldsymbol{\mu}, \boldsymbol{\Sigma}$	\mathbf{w}^{orig}	$\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\mu}_I, \boldsymbol{\Sigma}_I$	\mathbf{w}^{res}
Modified	$\boldsymbol{\mu}^{\text{inp}_1}, \boldsymbol{\Sigma}^{\text{inp}_1}$	$\mathbf{w}^{\text{inp}_1 \text{orig}}$	$\boldsymbol{\mu}_1^{\text{inp}_1}, \boldsymbol{\Sigma}_1^{\text{inp}_1}, \dots, \boldsymbol{\mu}_I^{\text{inp}_1}, \boldsymbol{\Sigma}_I^{\text{inp}_1}$	$\mathbf{w}^{\text{inp}_1 \text{res}}$
	\vdots	\vdots	\vdots	\vdots
	$\boldsymbol{\mu}^{\text{inp}_M}, \boldsymbol{\Sigma}^{\text{inp}_M}$	$\mathbf{w}^{\text{inp}_M \text{orig}}$	$\boldsymbol{\mu}_1^{\text{inp}_M}, \boldsymbol{\Sigma}_1^{\text{inp}_M}, \dots, \boldsymbol{\mu}_I^{\text{inp}_M}, \boldsymbol{\Sigma}_I^{\text{inp}_M}$	$\mathbf{w}^{\text{inp}_M \text{res}}$

Note that we use a different notation than we do in Section 2.1 and Section 1.3.1. Here, we explicitly show which method we use and we do not display the risk aversion parameter. This is because we want to make a clear distinction between the original MV solutions and the resampled solutions.

Before we discuss the quantitative results, we first give an impression of the influence of the variations of the input parameters on the allocations. We show the original MV solution, the original resampled solution, and the effect of changing the input parameters on both these solutions. Use again $\lambda_k = ((k-1)/25+1)^{-9}$. We take λ_{17} , $I = 200$, $T = 200$, and do $M = 12$ variations. Figure 2.10 shows what happens in terms of the allocations. The far left bar represents the original resampled solution, the one next to it represents the original MV solution. For $m = 1, \dots, 12$, we obtain repeatedly two portfolios: one of the resampling procedure (the left one of the pair) and one of the original MV optimization procedure (the right one of the pair). We analyze two cases: $T_{\text{inp}} = 200$ and $T_{\text{inp}} = 20000$. In the case of $T_{\text{inp}} = 200$ we are dealing with a large order of input parameter deviation and in the case of $T_{\text{inp}} = 20000$ we are dealing with a small order of input parameter deviation.

FIGURE 2.10: Resampled (left one of a pair) and original MV (right one of a pair) asset allocation of variation $m = 1, \dots, 12$ of the input parameters, with $\lambda_{17}, T = 200$, and $I = 200$.

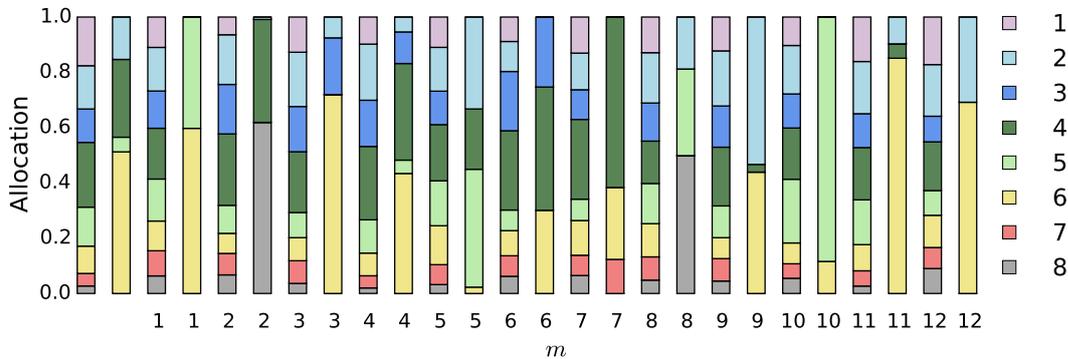


We see that a large deviation of input parameters results in large differences and that a small deviation of input parameters results in small differences, like we expect. Furthermore, we compare the MV solutions, based on the modified input parameters, to the original MV solution. We also compare the resampled solutions, based on the modified input parameters, to the original resampled solution. From Figure 2.10 it is hard to tell which method results in more robust portfolios. However, for $T_{\text{inp}} = 200$, we see that the original MV method results in extreme allocations, with alternately a large weight in a different asset class. Furthermore, we see that the resampling method results in more diversified portfolios. Hence, both the original resampled portfolio and the resampled portfolios based on the modified input parameters, are diversified. So we expect the resampling method to be more robust than the original MV optimization method.

To endorse this presumption, let us investigate the influence of parameter T once more. We use $T = 2, T_{\text{inp}} = 200, I = 200$, and λ_{17} . Figure 2.11 shows the allocations for $M = 12$ variations of input parameters.

We see that the resampled portfolios based on modified input parameters are all alike and close to the original resampled portfolio. Also, we see again that the MV method results in extreme portfolios, with most of them different than the original MV portfolio. This supports our presumption that in resampling, with T not too big, the diversification effect causes the resampled portfolios to be robust.

FIGURE 2.11: Resampled (left one of a pair) and original MV (right one of a pair) asset allocation of variation $m = 1, \dots, 12$ of the input parameters, with $\lambda_{17}, T = 2, T_{\text{inp}} = 200$, and $I = 200$.



We now investigate the extent to which resampling results in more robust portfolios. First, we set the construction to compare the robustness of the original MV optimization method to the robustness of the resampling method. Next, we investigate the influence of parameter T .

Again, we use the Euclidean norm to calculate the distance between two portfolios. For example, the distance between the original MV solution and the MV solution corresponding to variation m , is given by

$$d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}_m \text{orig}}) = \sqrt{\sum_{n=1}^N (w_n^{\text{orig}} - w_n^{\text{inp}_m \text{orig}})^2}.$$

We calculate the average and the standard deviation of the distances. For $\alpha = \mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{res}}, \beta_m = \mathbf{w}^{\text{inp}_m \text{orig}}, \mathbf{w}^{\text{inp}_m \text{res}}$:

$$\bar{d}(\alpha, \beta) = \frac{1}{M} \sum_{m=1}^M d(\alpha, \beta_m)$$

$$\text{Sd}(d(\alpha, \beta)) = \sqrt{\frac{1}{M} \sum_{m=1}^M (d(\alpha, \beta_m) - \bar{d}(\alpha, \beta))^2}.$$

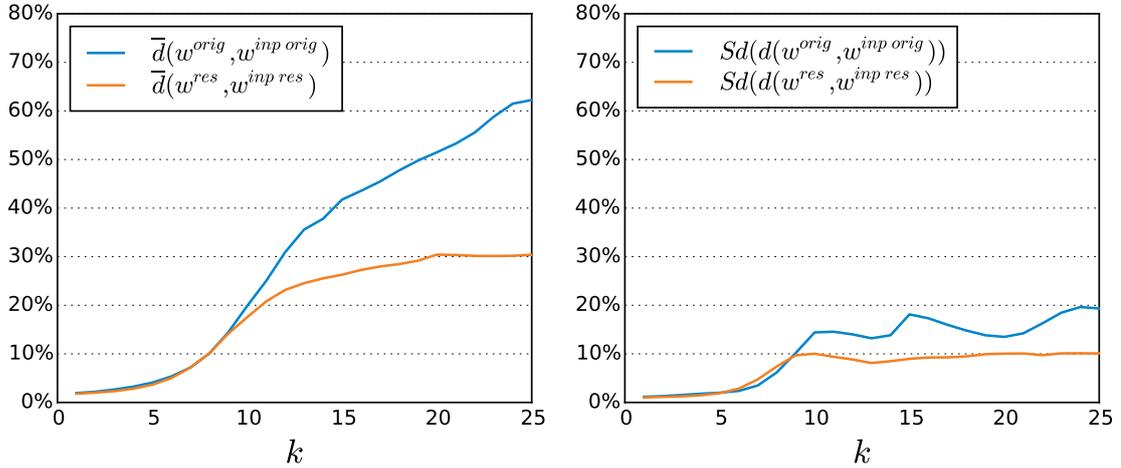
For risk aversion parameter λ_{17} , we investigate the robustness performance of MV optimization and resampling based on $M = 200$ variations of input parameters with $T_{\text{inp}} = 200$. We do the resampling method with parameters, $T = 200$ and $I = 200$. Rounded up to an integer and scaled to a maximum distance of $\sqrt{2}$, we obtain

$$\begin{aligned} \bar{d}(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}_m \text{orig}}) &= 37\% & \text{Sd}(d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}_m \text{orig}})) &= 24\% \\ \bar{d}(\mathbf{w}^{\text{res}}, \mathbf{w}^{\text{inp}_m \text{res}}) &= 16\% & \text{Sd}(d(\mathbf{w}^{\text{res}}, \mathbf{w}^{\text{inp}_m \text{res}})) &= 8\%. \end{aligned}$$

We see that $\bar{d}(\mathbf{w}^{\text{res}}, \mathbf{w}^{\text{inp}_m \text{res}})$ has a lower value than $\bar{d}(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}_m \text{orig}})$. This means that the average Euclidean distance between \mathbf{w}^{res} and the resampled solutions based on the variations of input parameters is smallest. Hence, in this case, resampling results in more robust portfolios than the original MV method does. Furthermore, $\text{Sd}(d(\mathbf{w}^{\text{res}}, \mathbf{w}^{\text{inp}_m \text{res}}))$ is also lower than $\text{Sd}(d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}_m \text{orig}}))$. This means that the spread of the values of $d(\mathbf{w}^{\text{res}}, \mathbf{w}^{\text{inp}_m \text{res}})$, $m = 1, \dots, M$, is smallest. We conclude that in this case, the resampling method performs best, that is, the resampling method is more robust.

We extend our analysis by looking at what happens for different risk aversion parameters. Figure 2.12 shows the results.

FIGURE 2.12: Robustness performances for different risk aversion parameters numbered by $k = 1, \dots, 25$, with $T_{\text{inp}} = 200$, $T = 200$, $M = 200$, and $I = 200$.



First, let us look at the standard deviation. We see that for both methods, we have a low spread for a high risk aversion level. We can explain this by observing that the average distances corresponding to both methods, are also low. As the risk aversion level decreases, in general, both the spread and the average distance increase. We explain below why this is the case. Furthermore, note the small decrease in spread around $k = 13$. Why this is, is not clear.

Next, we look at the average distances. We see that for both methods, we have robust portfolios for high risk aversion levels and we have non robust portfolios for low risk aversion levels. This is conform our presumption and we explain this phenomenon by looking at the extreme cases of minimum risk and maximum return in a single iteration m .

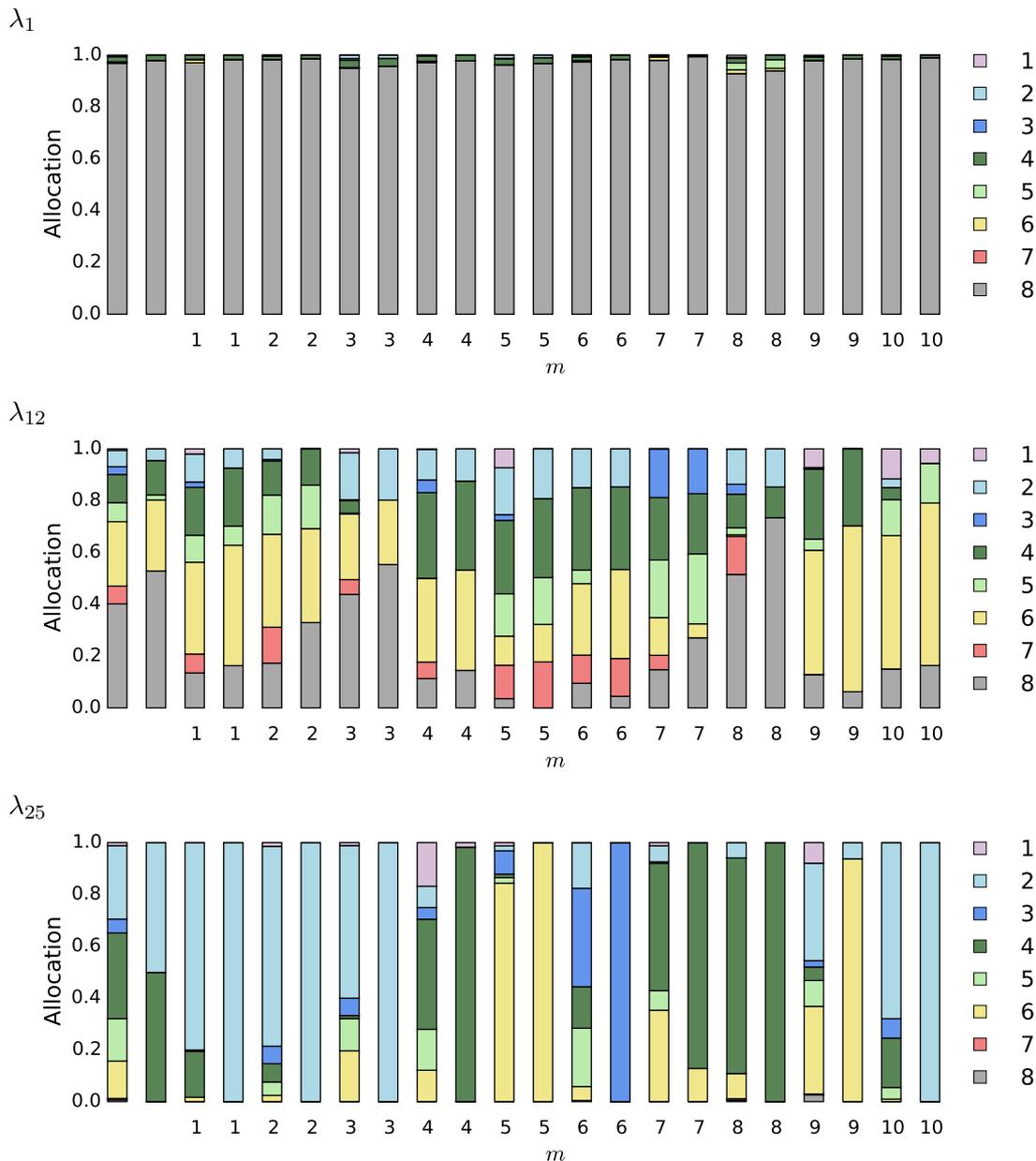
The only way to obtain maximum return in an iteration, is to fully invest in the asset class with the highest expected return. If there are more asset classes that share the maximum expected return (with comparable risks and correlations not equal to 1), we split the total weight between those asset classes, to obtain a lower expected risk. However, the probability that in an iteration m , two or more asset classes obtain the exact same highest return is tiny. So, there is a high probability that one asset class dominates the maximum return portfolio. In each variation of input parameters, that is, for each T_{inp} drawings from $\mathcal{N}(\mu, \Sigma)$, this can be another asset class. Therefore, we foresee low robustness for low risk aversion. Note that this is not necessarily the case in general but depends on the set of asset classes under consideration. To make the latter statement explicit, suppose that we consider a set of asset classes with low expected risk and low expected return, plus one asset class with high expected risk and high expected return. Then, in each iteration m , provided that T_{inp} is not too small, it is likely that the high return asset class again attains a high expected return and a high expected risk. In that case, we expect high robustness for low risk aversion portfolios.

If we want minimum risk, it is often possible, in general, to push the risk below the asset class with minimum risk, by diversifying between low risk asset classes. This is the case when we have multiple low risk asset classes with comparable characteristics. The

probability that we obtain a diversified portfolio for a high risk aversion parameter in an iteration m is big. So, in each iteration, it is likely that we obtain a well diversified portfolio for a high risk aversion level. However, note that in our example we have another reason for the low risk portfolios to be robust: here, European Bonds dominates each low risk portfolio, as we show next.

In Figure 2.13, we show what happens for different risk aversion parameters. We take λ_1 , λ_{12} , and λ_{25} , and compute the resampled and original MV allocations of 10 variations of input parameters. Recall that the far most left bar represents the original resampled portfolio, the bar next to it represents the original MV portfolio, the left bar of variation m represents the resampled solution, and the right bar represents the corresponding solution of the original MV method.

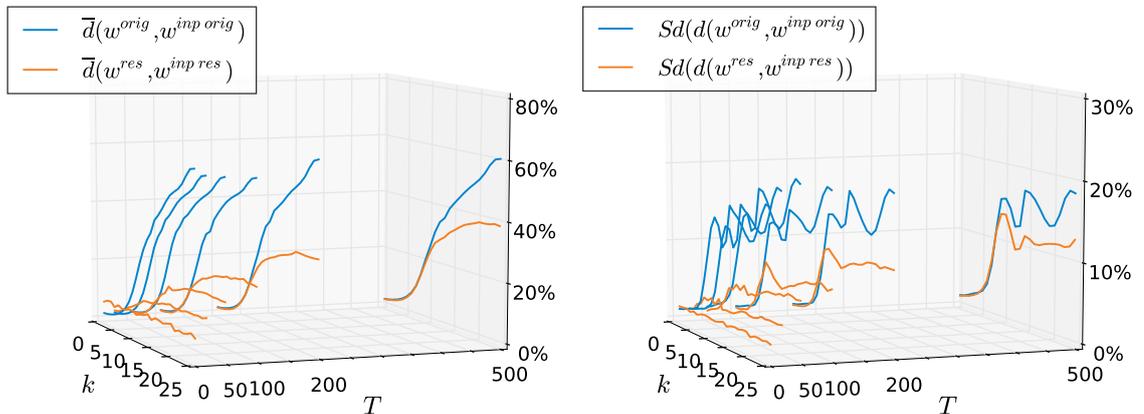
FIGURE 2.13: Resampled (left one of a pair) and original MV (right one of a pair) asset allocation of variation $m = 1, \dots, 10$ of the input parameters at three different risk aversion rates, with $T = 200$, $T_{\text{inp}} = 200$, and $I = 200$.



We see that in our case, European Bonds dominates each low risk portfolio. Apparently, European Bonds has such strong characteristics, that the probability that in a variation a more diversified portfolio can attain the same level of return but a lower risk, is tiny; in the 10 variations above, it does not occur. So, the initial strong characteristics of European Bonds and the lack of asset classes with comparable characteristics are the reason for the robustness in our case. We see that for λ_{12} , the portfolios are diversified; again with the resampled portfolios more diversified than the corresponding MV portfolios. Recall from Figure 2.12, that the resampling method is more robust than the original MV method for λ_{12} . So, here we see that the more diversified portfolios of the resampling method are also more robust than the less diversified portfolios of the original MV method. Furthermore, we see that for λ_{25} , almost all the MV solutions of the variations put all weight in one asset class. Which asset class this is, varies per variation m . The corresponding resampled portfolios are a bit more diversified, but still extreme in their allocations. We see again, when we combine the results of Figure 2.12 with the results of Figure 2.13, that the more diversified portfolios of the resampling method are also more robust than the less diversified portfolios of the original MV method.

We now know that, in our example, for a level of deviation of input parameters corresponding to $T_{\text{inp}} = 200$ and a resampling procedure with parameter $T = 200$, we have high robustness for high risk aversion levels and low robustness for low risk aversion levels. We want to know what happens if we vary the parameter T ; what influence does this have on the robustness? Let $T = 2, 20, 50, 100, 200, 500$, $T_{\text{inp}} = 200$, $M = 200$, and $I = 200$. We plot $\bar{d}(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp orig}})$, $\bar{d}(\mathbf{w}^{\text{res}}, \mathbf{w}^{\text{inp res}})$, $\text{Sd}(d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp orig}}))$, and $\text{Sd}(d(\mathbf{w}^{\text{res}}, \mathbf{w}^{\text{inp res}}))$, for the different values of T , in Figure 2.14. We use the same risk aversion parameters as above and scale again towards the maximum distance.

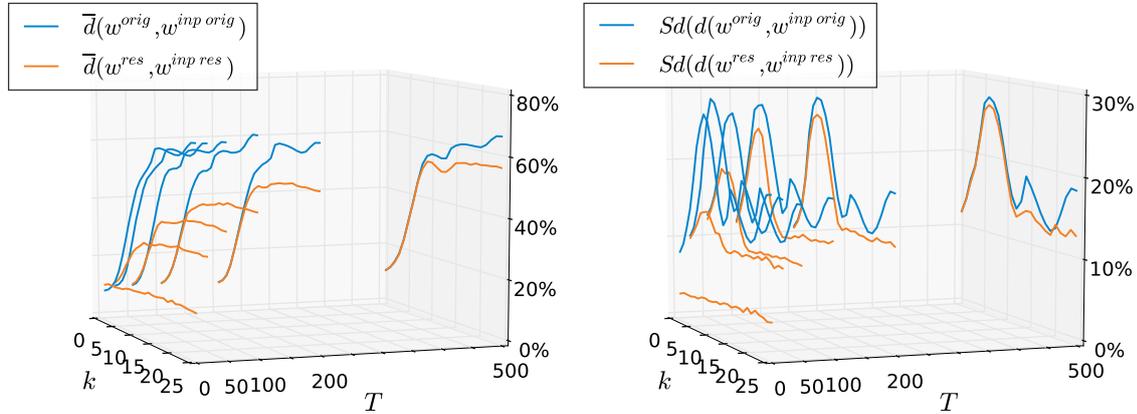
FIGURE 2.14: Robustness performances per risk aversion parameter numbered by $k = 1, \dots, 25$, and with $T_{\text{inp}} = 200$, $M = 200$, and $I = 200$.



First, note that the values for $\bar{d}(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp orig}})$ and $\text{Sd}(d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp orig}}))$ slightly differ per value of T . This is because each time we look at a different set of 200 variations. We see that for a larger value of T , the performance of the original MV method and the resampling method, are more alike. This is what we expect, since for large T , the parameters, $\boldsymbol{\mu}_i^{\text{inp } m}$ and $\boldsymbol{\Sigma}_i^{\text{inp } m}$, are close to $\boldsymbol{\mu}^{\text{inp } m}$ and $\boldsymbol{\Sigma}^{\text{inp } m}$, respectively. Furthermore, we see that the resampling method performs better for smaller T . However, for k up to 5, $T = 2$ and $T = 20$, the original MV method performs slightly better. Based on these observations, provided that we initially consider a somewhat diversified portfolio, it is best to choose a small value for T , from a robustness point of view.

Let us do the same analysis for an even higher level of deviation of input parameters: take $T_{\text{inp}} = 20$. Figure 2.15 shows the performances corresponding to the same values of T and λ_k .

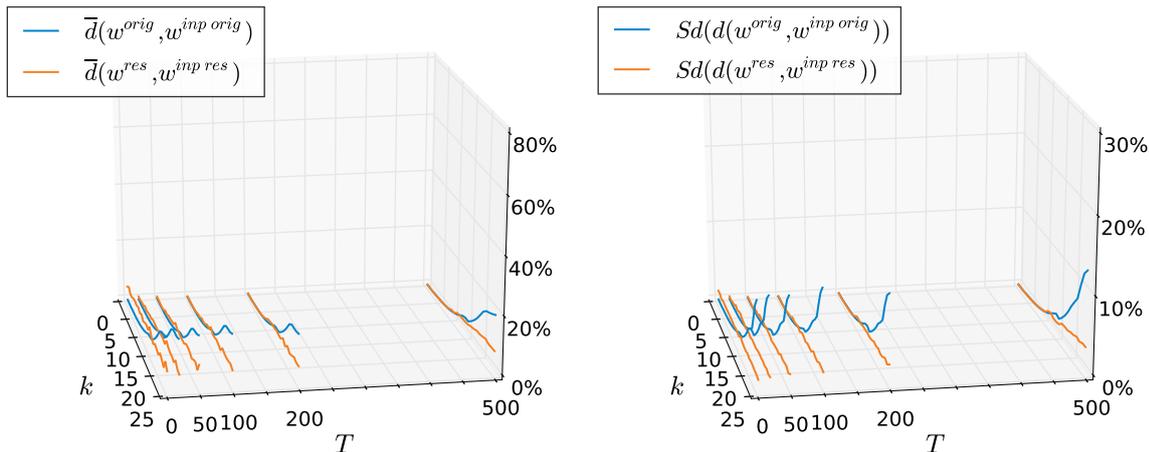
FIGURE 2.15: Robustness performances per risk aversion parameter numbered by $k = 1, \dots, 25$, and with $T_{\text{inp}} = 20$, $M = 200$, and $I = 200$.



All values of \bar{d} and Sd are higher than in the case that $T_{\text{inp}} = 200$. This is as expected since we consider a higher order of deviation of the input parameters. Furthermore, we see that the spread has a peak around $k = 7$, the reason for this is not clear. We also see that the resampling method with $T = 2$ again performs best, both in terms of the average and the standard deviations of the distances. Hence, from a robustness point of view, small T is the smartest choice in this case too.

We now investigate another extreme value of T_{inp} , let us take $T_{\text{inp}} = 20000$. Figure 2.16 shows the results for a low level of deviation of input parameters.

FIGURE 2.16: Robustness performances per risk aversion parameter numbered by $k = 1, \dots, 25$, and with $T_{\text{inp}} = 20000$, $M = 200$, and $I = 200$.



Again, it is logical that the observed values are low, since T_{inp} is big. We see that in this case, there is almost no difference between the results corresponding to small T and the results corresponding to big T . However, for $k < 16$, $T = 2$ and $T = 20$, we have that the original MV method outperforms the resampling method. This is not the case for large T . Hence, in this case large T is the best choice, if we want a robust portfolio.

In reality, we do not know the value of T_{inp} , that is, we do not know the exact order that the input parameters can deviate. Suppose that we also can not make an estimation about the order of deviation. Then, by comparing the cases of small T_{inp} versus big T_{inp} , we see that, from a robustness point of view, it is best to choose small T . If the input parameters deviate much from the original input parameters, a small value of T ensures robust portfolios. If the order of deviation of the input parameters is small, there is only a small difference between the robustness performances corresponding to a small value of T and the robustness performances corresponding to a big value of T .

Now, suppose we can make an estimation about the order of deviation of the input parameters and suppose we know for certain that our original input parameters are correct, that is, the expected portfolio risk and return are the real portfolio risk and return. Then, it would not be logical to look for a different, more robust portfolio. We know that the obtained portfolio is optimal, so there is no need to look for another portfolio. Hence, the robustness problem is inapplicable in this case.

Suppose we know that the real input parameters can deviate a lot from our original input parameters, then, one should choose a low value for T . The expected portfolio performance of the original MV portfolio, based on these uncertain input parameters, is also uncertain. So when we do resampling and we give in on the expected portfolio risk and return, this does not necessarily mean that our portfolio indeed performs poorly. Hence, it is better to choose a robust and diversified portfolio that has an uncertain performance, than to choose a non robust and non diversified portfolio that too has an uncertain performance.

Suppose we know that the real input parameters can not deviate much from our original input parameters, then, one should choose a high value for T . We only give in a little bit on the expected portfolio risk and return and find a portfolio that is more robust than our original MV portfolio. If we choose small T , we give in a lot on the expected portfolio risk and return, while we estimate them to be realistic. That does not make sense. Furthermore, the portfolio obtained by the original MV procedure is already fairly robust, since we know that the input parameters can not deviate much.

2.2.2.1 Conclusion

Input parameters are subject to estimation errors. If this would not be the case and we know that our input parameters are the real parameters, we do not need to look for more robust portfolios. However, we can not predict the future and past results are not a reliably indicator of future results. So, suppose we can estimate how much the input parameters can deviate from the original ones, then, we know which position to choose on the tradeoff between diversification and robustness effect on the one side and expected portfolio performance on the other side. If we estimate that the order of deviation is big, we decide to be on the diversification and robustness side, by choosing small T . If we estimate that the order of deviation is small, we decide to be on the expected portfolio performance side, by choosing a big value for T .

2.2.3 Convexity in the resampled frontier

We now have a good idea on how resampling performs in terms of robustness. However, there is an inconvenience that can occur in resampling: one can obtain convex parts in

the resampled frontier. In this section, we look at how resampling works in more detail and investigate this convexity.

One remark before we start our analysis: note that in resampling, we want parameter I as big as possible. The reason for this is that in resampling $i = 1, \dots, I$, the size of T determines the distribution of μ_i, Σ_i and the size of I determines to which extent μ_i, Σ_i converges to this distribution. How big I can be, depends on the computational power available.

We go back to the notation from Definition 2.1.1. Recall that w_{i,λ_k} denotes the obtained weights for resampling $i = 1, \dots, I$ and risk aversion parameter $\lambda_k = \lambda_1, \dots, \lambda_K$. We calculate the corresponding expected portfolio return, using the original expected return vector μ , i.e.,

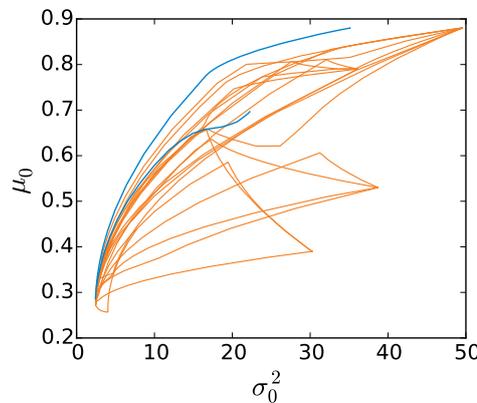
$$\mu_0 = w_{i,\lambda_k}^\top \mu.$$

We calculate the corresponding expected portfolio risk, using the original covariance matrix Σ , i.e.,

$$\sigma_0^2 = w_{i,\lambda_k}^\top \Sigma w_{i,\lambda_k}.$$

Hence, these portfolios always end up below the original efficient frontier. Figure 2.17 shows the original efficient frontier (upper blue line), together with the resampled frontier (lower blue line), and the frontiers of $I = 20$ resamplings (orange lines). We use $T = 50$ and $\lambda_k = ((k - 1)/25 + 1)^{-9}$, $k = 1, \dots, 25$, as above.

FIGURE 2.17: The original efficient frontier (upper blue line), the resampled frontier (lower blue line), and the frontiers of 20 resamplings (orange lines).



The first thing that comes to our attention is that the frontier of some resamplings has a negative slope. We explain how this can happen. Let λ_k be the risk aversion parameter. Suppose an asset class has a high expected risk and a low expected return and suppose that in some resampling i , this asset class attains a low expected risk and a high expected return, compared to the other asset classes. Then, the asset class under consideration obtains a lot of weight in this resampling. However, the portfolio return, $w_{i,\lambda_k}^\top \mu$, is low and the portfolio risk, $w_{i,\lambda_k}^\top \Sigma w_{i,\lambda_k}$, is high. Hence, higher risk does not necessarily result in higher return. Thus, in this case the frontier corresponding to resampling i , has a negative slope.

Furthermore, note that the resampled frontier contains a convex part. We show that we do not expect this to happen in Lemma 2.2.1 and we show in Theorem 2.2.2 that for a slight variation of the resampling procedure the probability of convex parts in the resampled frontier converges to zero, as the number of resamplings goes to infinity.

Lemma 2.2.1. *The efficient frontier of MV optimization is concave.*

Proof. We assume that the real portfolio returns are normally distributed: $\mathbf{r} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Let λ_1 and λ_2 be two risk aversion parameters and let \mathbf{w}_{λ_1} and \mathbf{w}_{λ_2} be their corresponding optimal portfolios, respectively. Their real portfolio returns have distributions

$$\begin{aligned} r_{0,\lambda_1} &\sim \mathcal{N}(\mathbf{w}_{\lambda_1}^\top \boldsymbol{\mu}, \mathbf{w}_{\lambda_1}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_1}) = \mathcal{N}(\mu_{0,\lambda_1}, \sigma_{0,\lambda_1}^2) \\ r_{0,\lambda_2} &\sim \mathcal{N}(\mathbf{w}_{\lambda_2}^\top \boldsymbol{\mu}, \mathbf{w}_{\lambda_2}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_2}) = \mathcal{N}(\mu_{0,\lambda_2}, \sigma_{0,\lambda_2}^2). \end{aligned}$$

We get

$$\begin{bmatrix} r_{0,\lambda_1} \\ r_{0,\lambda_2} \end{bmatrix} = \begin{bmatrix} \mathbf{w}_{\lambda_1}^\top \mathbf{r} \\ \mathbf{w}_{\lambda_2}^\top \mathbf{r} \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mathbf{w}_{\lambda_1}^\top \boldsymbol{\mu} \\ \mathbf{w}_{\lambda_2}^\top \boldsymbol{\mu} \end{bmatrix}, \begin{bmatrix} \mathbf{w}_{\lambda_1}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_1} & \mathbf{w}_{\lambda_1}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_2} \\ \mathbf{w}_{\lambda_2}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_1} & \mathbf{w}_{\lambda_2}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_2} \end{bmatrix} \right). \quad (2.1)$$

Let $\alpha \in [0, 1]$ and take the product with $[\alpha, 1 - \alpha]$:

$$\begin{aligned} \alpha \mathbf{w}_{\lambda_1}^\top \mathbf{r} + (1 - \alpha) \mathbf{w}_{\lambda_2}^\top \mathbf{r} &\sim \mathcal{N} \left([\alpha, 1 - \alpha] \begin{bmatrix} \mathbf{w}_{\lambda_1}^\top \boldsymbol{\mu} \\ \mathbf{w}_{\lambda_2}^\top \boldsymbol{\mu} \end{bmatrix}, \right. \\ &\quad \left. [\alpha, 1 - \alpha] \begin{bmatrix} \mathbf{w}_{\lambda_1}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_1} & \mathbf{w}_{\lambda_2}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_1} \\ \mathbf{w}_{\lambda_1}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_2} & \mathbf{w}_{\lambda_2}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_2} \end{bmatrix} [\alpha, 1 - \alpha] \right) \\ &\sim \mathcal{N} \left(\alpha \mu_{0,\lambda_1} + (1 - \alpha) \mu_{0,\lambda_2}, \right. \\ &\quad \left. \alpha^2 \sigma_{0,\lambda_1}^2 + 2\alpha(1 - \alpha) \mathbf{w}_{\lambda_1}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_2} + (1 - \alpha)^2 \sigma_{0,\lambda_2}^2 \right). \end{aligned}$$

Let $\mathbf{w}_{\lambda_3} = \alpha \mathbf{w}_{\lambda_1} + (1 - \alpha) \mathbf{w}_{\lambda_2}$. We know that \mathbf{w}_{λ_3} is a feasible solution, but it is not necessarily the optimal solution corresponding to risk aversion parameter λ_3 : here, is it a portfolio obtained by a convex combination of \mathbf{w}_{λ_1} and \mathbf{w}_{λ_2} . We obtain

$$\begin{aligned} \mu_{0,\lambda_3} &= \alpha \mu_{0,\lambda_1} + (1 - \alpha) \mu_{0,\lambda_2} \\ \sigma_{0,\lambda_3}^2 &= \alpha^2 \sigma_{0,\lambda_1}^2 + 2\alpha(1 - \alpha) \mathbf{w}_{\lambda_1}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_2} + (1 - \alpha)^2 \sigma_{0,\lambda_2}^2 \\ &\leq \alpha \sigma_{0,\lambda_1}^2 + (1 - \alpha) \sigma_{0,\lambda_2}^2. \end{aligned}$$

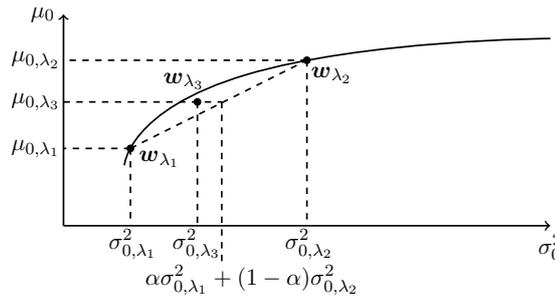
The latter inequality holds, because

$$\mathbf{w}_{\lambda_1}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_2} = \langle \mathbf{w}_{\lambda_1}, \mathbf{w}_{\lambda_2} \rangle_{\boldsymbol{\Sigma}} \leq \sqrt{\mathbf{w}_{\lambda_1}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_1}} \sqrt{\mathbf{w}_{\lambda_2}^\top \boldsymbol{\Sigma} \mathbf{w}_{\lambda_2}} = \sigma_{0,\lambda_1} \sigma_{0,\lambda_2}$$

and because $(\alpha \sigma_{0,\lambda_1} + (1 - \alpha) \sigma_{0,\lambda_2})^2 \leq \alpha \sigma_{0,\lambda_1}^2 + (1 - \alpha) \sigma_{0,\lambda_2}^2$.

Hence, the efficient frontier is concave. Figure 2.18 gives a graphical representation.

FIGURE 2.18: Graphical representation.



□

In Theorem 2.2.2 below, we use a variation of the resampling method that we define in Section 2.1. Here, in resampling $i = 1, \dots, I$, we only use μ_i as resampling parameter. So, we obtain a solution to resampling i , by using parameters, μ_i and Σ . Secondly, we average per value of μ_0 and not per risk aversion parameter, like we do above. For this variation of the resampling method we are able to proof the following theorem.

Theorem 2.2.2. *Assume we are allowed to go short. Do a variation of the resampling method: use for resampling i , input parameters, μ_i and Σ , and average per fixed required return. The probability of convex parts in the resampled frontier converges to zero, as I goes to infinity.*

Proof. Let $\mu_0 \in \mathbb{R}_{>\mu_{GMV}}$ be the fixed required return. Let $I, N \in \mathbb{N}$ and suppose we have resampling parameters μ_1, \dots, μ_I , and corresponding weights w_1, \dots, w_I . Assume, without loss of generality, that for all $i = 1, \dots, I$, there is a $n \in \{1, \dots, N\}$, such that $\mu_{n,i} \geq \mu_0$. Recall from Lemma 1.2.1 that for resampling $i = 1, \dots, I$, solution $w_i(\mu_0) = g_i + h_i\mu_0$ is such that $\mu^\top [g_i + h_i\mu_0] = \mu_0$, with

$$\begin{aligned} g_i &= \frac{1}{ac_{\mu_i} - b_{\mu_i}^2} \Sigma^{-1} [c\mathbf{1} - b\mu_i] \\ h_i &= \frac{1}{ac_{\mu_i} - b_{\mu_i}^2} \Sigma^{-1} [a\mu_i - b\mathbf{1}] \\ a &= \mathbf{1}^\top \Sigma^{-1} \mathbf{1} \\ b_{\mu_i} &= \mathbf{1}^\top \Sigma^{-1} \mu_i \\ c_{\mu_i} &= \mu_i^\top \Sigma^{-1} \mu_i. \end{aligned}$$

Let

$$\bar{w}(\mu_0) = \frac{1}{I} \sum_{i=1}^I w_i(\mu_0) = \frac{1}{I} \sum_{i=1}^I [g_i + h_i\mu_0] = \frac{1}{I} \sum_{i=1}^I g_i + \frac{1}{I} \sum_{i=1}^I h_i\mu_0 = \bar{g} + \bar{h}\mu_0.$$

From Corollary 1.2.2 we know that the following function represents the frontier corresponding to a resampling $i = 1, \dots, I$:

$$\mu_0(\sigma_0^2) = \frac{-g_i^\top \Sigma h_i + \sqrt{(g_i^\top \Sigma h_i)^2 - (h_i^\top \Sigma h_i)(g_i^\top \Sigma g_i - \sigma_0^2)}}{h_i^\top \Sigma h_i}$$

We want to know how the resampling method performs in terms of the original input parameters, so we compute $\mu_0 = \mu^\top [g_i + h_i\mu_0]$:

$$\begin{aligned} \mu_0(\sigma_0^2) &= w_i^\top(\mu_0(\sigma_0^2)) \mu \\ &= g_i^\top \mu + \frac{-g_i^\top \Sigma h_i + \sqrt{(g_i^\top \Sigma h_i)^2 - (h_i^\top \Sigma h_i)(g_i^\top \Sigma g_i - \sigma_0^2)}}{h_i^\top \Sigma h_i} h_i^\top \mu. \end{aligned}$$

This is a concave function if and only if $h_i^\top \mu \geq 0$, i.e., if and only if

$$\frac{1}{ac_{\mu_i} - b_{\mu_i}^2} \mu^\top \Sigma^{-1} [a\mu_i - b_{\mu_i} \mathbf{1}] \geq 0.$$

From Lemma 1.2.4 we know that the inequality above is equivalent to

$$\mu^\top \Sigma^{-1} [a\mu_i - b_{\mu_i} \mathbf{1}] \geq 0.$$

We have $\boldsymbol{\mu}_i \sim \mathcal{N}(\boldsymbol{\mu}, \frac{1}{T}\boldsymbol{\Sigma})$, so

$$\begin{aligned} a\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_i &\sim \mathcal{N}\left(a\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}, \frac{a^2}{T}\boldsymbol{\Sigma}^{-1}\right) \\ b_{\boldsymbol{\mu}_i} = \mathbf{1}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_i &\sim \mathcal{N}\left(b_{\boldsymbol{\mu}}, \frac{a}{T}\right) \\ \boldsymbol{\Sigma}^{-1}\mathbf{1}b_{\boldsymbol{\mu}_i} &\sim \mathcal{N}\left(\boldsymbol{\Sigma}^{-1}\mathbf{1}b_{\boldsymbol{\mu}}, \frac{a}{T}[\boldsymbol{\Sigma}^{-1}\mathbf{1}][\boldsymbol{\Sigma}^{-1}\mathbf{1}]^\top\right). \end{aligned}$$

Hence,

$$\begin{aligned} \boldsymbol{\mu}^\top \boldsymbol{\Sigma}^{-1} [a\boldsymbol{\mu}_i - b_{\boldsymbol{\mu}_i}\mathbf{1}] &\sim \mathcal{N}\left(\boldsymbol{\mu}^\top [a\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} - \boldsymbol{\Sigma}^{-1}\mathbf{1}b_{\boldsymbol{\mu}}], \boldsymbol{\mu}^\top \left[\frac{a^2}{T}\boldsymbol{\Sigma}^{-1} - \frac{a}{T}[\boldsymbol{\Sigma}^{-1}\mathbf{1}][\boldsymbol{\Sigma}^{-1}\mathbf{1}]^\top\right] \boldsymbol{\mu}\right) \\ &\sim \mathcal{N}\left(ac_{\boldsymbol{\mu}} - b_{\boldsymbol{\mu}}^2, \frac{a}{T}(ac_{\boldsymbol{\mu}} - b_{\boldsymbol{\mu}}^2)\right). \end{aligned} \quad (2.2)$$

Define $R = \max_{i=1, \dots, I} \{(ac_{\boldsymbol{\mu}_i} - b_{\boldsymbol{\mu}_i}^2)^{-1}\} > 0$, then

$$\begin{aligned} \mathbb{P}(\boldsymbol{\mu}^\top \bar{\mathbf{h}} \geq 0) &= \mathbb{P}\left(\boldsymbol{\mu}^\top \left[\frac{1}{I} \sum_{i=1}^I \mathbf{h}_i\right] \geq 0\right) \\ &\geq \mathbb{P}\left(\boldsymbol{\mu}^\top \left[\sum_{i=1}^I R\boldsymbol{\Sigma}^{-1}[a\boldsymbol{\mu}_i - b_{\boldsymbol{\mu}_i}\mathbf{1}]\right] \geq 0\right). \end{aligned}$$

Use (2.2) to see that

$$\boldsymbol{\mu}^\top \left[\sum_{i=1}^I R\boldsymbol{\Sigma}^{-1}[a\boldsymbol{\mu}_i - b_{\boldsymbol{\mu}_i}\mathbf{1}]\right] \sim \mathcal{N}(RI(ac_{\boldsymbol{\mu}} - b_{\boldsymbol{\mu}}^2), R^2I(ac_{\boldsymbol{\mu}} - b_{\boldsymbol{\mu}}^2)).$$

Rewrite this to a standard normal distribution:

$$\begin{aligned} &\mathbb{P}\left(\boldsymbol{\mu}^\top \left[\sum_{i=1}^I R\boldsymbol{\Sigma}^{-1}[a\boldsymbol{\mu}_i - b_{\boldsymbol{\mu}_i}\mathbf{1}]\right] \geq 0\right) \\ &= \mathbb{P}\left(\frac{\boldsymbol{\mu}^\top \left[\sum_{i=1}^I R\boldsymbol{\Sigma}^{-1}[a\boldsymbol{\mu}_i - b_{\boldsymbol{\mu}_i}\mathbf{1}]\right] - RI(ac_{\boldsymbol{\mu}} - b_{\boldsymbol{\mu}}^2)}{\sqrt{R^2I(ac_{\boldsymbol{\mu}} - b_{\boldsymbol{\mu}}^2)}} \geq -\frac{RI(ac_{\boldsymbol{\mu}} - b_{\boldsymbol{\mu}}^2)}{\sqrt{R^2I(ac_{\boldsymbol{\mu}} - b_{\boldsymbol{\mu}}^2)}}\right) \\ &= \mathbb{P}\left(Y \geq -\sqrt{I(ac_{\boldsymbol{\mu}} - b_{\boldsymbol{\mu}}^2)}\right), \text{ for } Y \sim \mathcal{N}(0, 1). \end{aligned}$$

Hence, $\mathbb{P}(\boldsymbol{\mu}^\top \bar{\mathbf{h}} \geq 0) \rightarrow 1$, as $I \rightarrow \infty$. □

2.2.3.1 Conclusion

In Theorem 2.2.2, we show that we probably do not have to worry about convexity in the resampled frontier, when we take parameter I big enough. Note that we can not state this with certainty, since we prove it only for a simplified variation of the resampling procedure. However, the fact that we are able to prove it for this simplified version is reassuring.

Chapter 3

Shrinkage

In the previous chapter, we discuss the resampling method; a procedure that makes the MV optimization method more robust, given the estimated input parameters. In this chapter, we look at a procedure that tries to make the parameter estimation more robust: shrinkage. In MV optimization, we use the sample means and the sample covariance matrix. In the shrinkage method, we still use the sample means, but another covariance matrix: a shrunken covariance matrix. We investigate what effect a shrunken covariance matrix has on the robustness of the MV optimization method. We use the shrinkage method introduced by Ledoit and Wolf [6]. They look at a variation of the shrinkage method, which can be used when there many stocks, but only a few historical data points available. They explain

“(...) when there are many stocks under consideration, especially compared to the number of historical return observations available (as is the usual case), the sample covariance matrix is estimated with a lot of error. This means the most extreme coefficients in such an estimated matrix tend to take on extreme values, not because they are correct but because they are subject to an extreme amount of error. Invariably the mean-variance optimization software will latch onto the extremes, and place its biggest bets on the coefficients that are the most unreliable.” [6].

Ortec Finance uses a Monte Carlo method to generate economic scenarios, in which all kind of future states of global economies and capital markets are simulated. The number of scenarios that is generated is large. Hence, in our case we have many data points, compared to the number of stocks under consideration. We investigate whether or not the shrinkage method is still useful.

3.1 Formulation

We define the shrinkage procedure for the covariance matrix. Let $\Gamma \in \mathbb{R}^N \times \mathbb{R}^N$ denote the *shrinkage target*; sometimes we abbreviate it to *target*. Let $\delta \in [0, 1]$ be the *shrinkage parameter*.

Definition 3.1.1. The *shrinkage procedure* is as follows. We have the sample covariance matrix Σ and we have a target Γ . The amount of shrinkage towards the target, is determined by the value of δ . The covariance matrix we obtain, by shrinking Σ towards Γ , is $\delta\Sigma + (1 - \delta)\Gamma$.

We call $\delta\Sigma + (1 - \delta)\Gamma$ the *shrunken covariance matrix*.

The following two questions arise.

- Towards which target should we shrink?
- How much should we shrink towards this target?

There are many different shrinkage methods. These targets can be split into two groups: linear and nonlinear shrinkage methods. Here, we only look at linear shrinkage and we investigate this linear shrinkage method by means of a real world data set.

3.2 Data

We use US securities from the S&P 500 list; a list of the 500 biggest companies in the U.S.A., based on market capitalization. We choose 30 companies among the 100 largest of which there is data available of the historical yearly returns from 1980 onward. The selected companies are listed in Table 3.1, Table 3.2 gives the expected returns, and Table 3.3 gives the covariance matrix, divided by 100.

TABLE 3.1: List of securities.

1. 3M Co.	11. JPMorgan Chase & Co.	21. Walt Disney Co.
2. TJX Cos.	12. Pfizer Inc.	22. Occidental Petroleum Corp.
3. Exxon Mobil Corp.	13. Coca-Cola Co.	23. United Technologies Corp.
4. IBM Corp.	14. Merck & Co. Inc.	24. Caterpillar Inc.
5. General Electric Co.	15. PepsiCo Inc.	25. American Express Co.
6. Chevron Corp.	16. Bank of America Corp.	26. Altria Group Inc.
7. Wal-Mart Stores Inc.	17. McDonald's Corp.	27. CVS Health Corp.
8. Procter & Gamble Co.	18. ConocoPhillips	28. Bristol-Myers Squibb Co.
9. Johnson & Johnson	19. Abbott Laboratories	29. Boeing Co.
10. Wells Fargo & Co.	20. Schlumberger Ltd.	30. Union Pacific Corp.

TABLE 3.2: Expected yearly returns of the securities, rounded to 2 decimals.

1.	10.58	11.	12.50	21.	17.24
2.	26.97	12.	13.59	22.	6.80
3.	9.21	13.	14.65	23.	11.99
4.	10.01	14.	13.48	24.	9.82
5.	12.85	15.	14.71	25.	14.98
6.	7.24	16.	13.45	26.	18.76
7.	22.26	17.	16.42	27.	15.92
8.	12.15	18.	7.60	28.	12.41
9.	13.33	19.	14.32	29.	14.32
10.	13.99	20.	8.67	30.	9.94

Note that we only look at one asset class: equity. We do not include fixed income or cash equivalents assets. Furthermore, we know that this data is biased, since we only include companies that are biggest right now. However, this does not interfere with our analysis, since we only use the data for investigating the robustness performances of the shrinkage method.

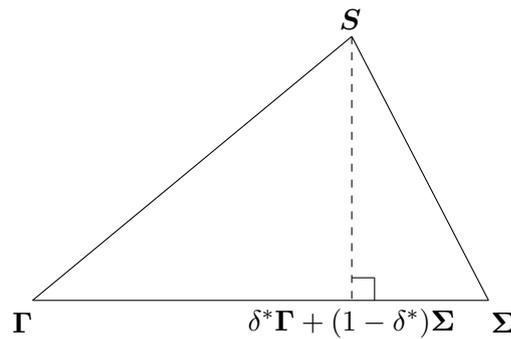
TABLE 3.3: Covariance matrix of the securities, divided by 100, rounded to 1 decimal.

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.	16.	17.	18.	19.	20.	21.	22.	23.	24.	25.	26.	27.	28.	29.	30.
1.	3.3	2.0	0.5	0.9	2.4	0.8	1.0	1.2	1.1	1.3	1.6	0.8	1.9	1.9	0.8	2.0	0.1	0.9	1.0	2.9	2.6	1.4	2.5	1.2	4.0	1.6	1.6	1.4	3.4	2.1
2.	2.0	23.0	0.6	7.4	4.6	-0.2	6.7	1.4	1.9	2.8	9.0	5.6	3.2	1.1	1.1	6.7	2.2	-0.7	3.5	0.7	2.1	-0.7	4.3	0.1	6.7	1.9	9.0	3.6	4.1	3.0
3.	0.5	0.6	2.3	0.9	1.8	1.8	1.0	0.8	1.0	1.6	1.4	1.9	1.3	2.5	1.1	1.3	1.5	1.7	1.4	2.1	1.5	1.4	1.8	0.9	1.1	1.3	1.5	2.0	2.2	1.3
4.	0.9	7.4	0.9	8.8	3.1	1.1	4.0	1.5	1.3	-0.6	0.7	2.5	0.9	0.4	-0.7	0.2	1.4	0.4	2.3	1.0	-0.3	0.4	3.2	1.6	3.8	-0.5	2.8	2.8	1.3	-0.1
5.	2.4	4.6	1.8	3.1	6.6	1.7	5.4	2.8	2.5	3.1	4.3	3.8	3.2	2.8	1.8	4.1	3.3	1.0	2.5	3.0	3.4	0.8	3.8	1.4	4.8	1.4	3.7	3.9	3.8	1.1
6.	0.8	-0.2	1.8	1.1	1.7	3.0	-0.5	1.2	0.7	1.3	1.1	0.8	1.4	1.3	1.3	1.1	1.6	2.3	1.2	3.3	2.1	2.1	2.1	2.3	1.8	1.2	0.8	1.4	2.2	1.9
7.	1.0	6.7	1.0	4.0	5.4	-0.5	14.7	3.4	3.9	2.9	2.9	5.6	3.3	3.7	1.4	2.7	3.4	-2.7	4.6	-1.7	1.4	-2.8	2.0	-2.5	2.4	0.3	4.4	4.2	-0.7	-0.8
8.	1.2	1.4	0.8	1.5	2.8	1.2	3.4	3.1	1.9	1.4	1.0	1.5	2.4	1.4	1.5	1.9	1.5	0.4	1.4	1.8	2.3	0.2	1.4	0.4	2.4	0.5	1.8	1.8	1.1	0.6
9.	1.1	1.9	1.0	1.3	2.5	0.7	3.9	1.9	3.7	2.4	2.0	3.9	2.8	4.0	2.3	2.5	1.6	-0.4	3.1	1.1	2.4	-0.6	1.4	-0.7	1.7	2.4	1.7	2.7	1.8	1.2
10.	1.3	2.8	1.6	-0.6	3.1	1.3	2.9	1.4	2.4	6.1	4.3	4.8	2.9	3.8	2.4	4.7	0.9	0.6	2.5	3.3	2.9	0.8	2.4	1.4	2.3	2.8	2.6	3.0	2.8	2.9
11.	1.6	9.0	1.4	0.7	4.3	1.1	2.9	1.0	2.0	4.3	11.7	5.3	3.2	2.9	2.0	8.0	3.7	0.5	1.5	2.2	3.8	1.2	3.7	1.4	5.8	1.5	3.8	2.8	3.0	3.1
12.	0.8	5.6	1.9	2.5	3.8	0.8	5.6	1.5	3.9	4.8	5.3	9.2	4.5	6.4	2.7	4.3	2.2	-0.3	4.7	1.6	1.8	-1.2	3.0	-1.1	2.4	4.6	4.2	5.7	2.5	1.6
13.	1.9	3.2	1.3	0.9	3.2	1.4	3.3	2.4	2.8	2.9	3.2	4.5	5.3	5.0	2.6	3.6	2.2	0.5	3.1	2.8	3.5	0.4	2.4	0.0	2.8	3.7	2.6	3.1	3.3	2.4
14.	1.9	1.1	2.5	0.4	2.8	1.3	3.7	1.4	4.0	3.8	2.9	6.4	5.0	9.2	3.7	3.1	1.9	0.2	4.3	2.9	3.7	-0.2	2.5	-0.4	2.3	5.6	2.8	4.5	2.9	2.9
15.	0.8	1.1	1.1	-0.7	1.8	1.3	1.4	1.5	2.3	2.4	2.0	2.7	2.6	3.7	4.3	2.7	2.1	0.6	2.5	2.5	3.4	0.4	1.1	0.8	1.5	3.4	2.1	2.4	2.2	1.7
16.	2.0	6.7	1.3	0.2	4.1	1.1	2.7	1.9	2.5	4.7	8.0	4.3	3.6	3.1	2.7	13.4	0.8	1.5	2.2	2.3	3.9	1.0	3.3	0.1	4.4	2.8	2.5	1.5	3.8	3.2
17.	0.1	2.2	1.5	1.4	3.3	1.6	3.4	1.5	1.6	0.9	3.7	2.2	2.2	1.9	2.1	0.8	5.5	-0.1	2.2	0.2	2.7	0.3	2.2	1.1	1.8	1.2	2.7	2.7	1.2	0.2
18.	0.9	-0.7	1.7	0.4	1.0	2.3	-2.7	0.4	-0.4	0.6	0.5	-0.3	0.5	0.2	0.6	1.5	-0.1	4.4	-0.5	4.1	1.0	3.1	1.8	2.8	1.5	1.3	1.0	0.4	3.4	1.9
19.	1.0	3.5	1.4	2.3	2.5	1.2	4.6	1.4	3.1	2.5	1.5	4.7	3.1	4.3	2.5	2.2	2.2	-0.5	4.8	0.5	2.5	-1.0	2.0	-0.5	0.6	3.5	3.2	3.5	1.5	1.2
20.	2.9	0.7	2.1	1.0	3.0	3.3	-1.7	1.8	1.1	3.3	2.2	1.6	2.8	2.9	2.5	2.3	0.2	4.1	0.5	9.5	2.8	5.1	3.6	3.3	5.2	3.0	2.6	2.4	5.9	4.2
21.	2.6	2.1	1.5	-0.3	3.4	2.1	1.4	2.3	2.4	2.9	3.8	1.8	3.5	3.7	3.4	3.9	2.7	1.0	2.5	2.8	7.8	1.1	1.9	2.1	5.1	2.6	3.3	2.9	3.3	2.3
22.	1.4	-0.7	1.4	0.4	0.8	2.1	-2.8	0.2	-0.6	0.8	1.2	-1.2	0.4	-0.2	0.4	1.0	0.3	3.1	-1.0	5.1	1.1	5.6	2.1	3.1	3.1	0.3	0.1	0.0	3.3	2.5
23.	2.5	4.3	1.8	3.2	3.8	2.1	2.0	1.4	1.4	2.4	3.7	3.0	2.4	2.5	1.1	3.3	2.2	1.8	2.0	3.6	1.9	2.1	5.1	1.8	4.6	1.9	3.6	2.6	4.8	2.5
24.	1.2	0.1	0.9	1.6	1.4	2.3	-2.5	0.4	-0.7	1.4	1.4	-1.1	0.0	4.0	0.8	0.1	1.1	2.8	-0.5	3.3	2.1	3.1	1.8	7.6	2.6	-0.1	0.2	-0.3	0.5	2.2
25.	4.0	6.7	1.1	3.8	4.8	1.8	2.4	2.4	1.7	2.3	5.8	2.4	2.8	2.3	1.5	4.4	1.8	1.5	0.6	5.2	5.1	3.1	4.6	2.6	10.9	0.7	4.0	3.0	5.3	2.6
26.	1.6	1.9	1.3	-0.5	1.4	1.2	0.3	0.5	2.4	2.8	1.5	4.6	3.7	5.6	3.4	2.8	1.2	1.3	3.5	3.0	2.6	0.3	1.9	-0.1	0.7	7.6	3.6	3.4	3.5	2.5
27.	1.6	9.0	1.5	2.8	3.7	0.8	4.4	1.8	1.7	2.6	3.8	4.2	2.6	2.8	2.1	2.5	2.7	1.0	3.2	2.6	3.3	0.1	3.6	0.2	4.0	3.6	8.9	4.2	3.8	1.3
28.	1.4	3.6	2.0	2.8	3.9	1.4	4.2	1.8	2.7	3.0	2.8	5.7	3.1	4.5	2.4	1.5	2.7	0.4	3.5	2.4	2.9	0.0	2.6	-0.3	3.0	3.4	4.2	6.4	3.4	1.0
29.	3.4	4.1	2.2	1.3	3.8	2.2	-0.7	1.1	1.8	2.8	3.0	2.5	3.3	2.9	2.2	3.8	1.2	3.4	1.5	5.9	3.3	3.3	4.8	0.5	5.3	3.5	3.8	3.4	10.5	3.4
30.	2.1	3.0	1.3	-0.1	1.1	1.9	-0.8	0.6	1.2	2.9	3.1	1.6	2.4	2.9	1.7	3.2	0.2	1.9	1.2	4.2	2.3	2.5	2.5	2.2	2.6	2.5	1.3	1.0	3.4	5.5

3.3 Linear shrinkage

We have $N = 30$ stocks and sample covariance matrix $\Sigma = (\sigma_{n,n'})$, $n, n' = 1, \dots, N$. Let $S = (s_{n,n'})$ denote the ‘real’ covariance matrix, which is unknown. We have Σ as estimator of S , but we search for a better, more robust, estimator, so we shrink Σ towards a target: $\Gamma = (\gamma_{n,n'})$. Let δ^* be the optimal shrinkage parameter, that is, δ^* minimizes the expected distance between the shrunken covariance matrix and S . Figure 3.1 shows the geometric interpretation of linear shrinkage.

FIGURE 3.1: Geometric interpretation of the shrinkage method.



3.3.1 Constant correlation target

Let T denote our total number of data points. Let $X = (\chi_{n,t})$ be the matrix of returns, with $n = 1, \dots, N$ and $t = 1, \dots, T$. So $\chi_{n,t}$ denotes the return of security n on time period t . The sample means are given by $\mu^\top = [\mu_1, \dots, \mu_N]$, with

$$\mu_n = \bar{\chi}_n = \frac{1}{T} \sum_{t=1}^T \chi_{n,t}.$$

We look at a target introduced by Ledoit and Wolf: the covariance matrix with constant correlation. Let $\mathbf{G} = (g_{n,n'})$, $n, n' = 1, \dots, N$, be the constant correlation matrix, based on variances and covariances from the real covariance matrix \mathbf{S} . We do not know \mathbf{S} , hence our shrinkage target, $\mathbf{\Gamma}$, will be an estimation of \mathbf{G} . In order to estimate \mathbf{G} , recall the sample correlation matrix $\mathbf{P} = (\rho_{n,n'})$. Let $\mathbf{R} = (r_{n,n'})$ denote the 'real' correlation matrix, which is unknown. We have $r_{n,n'} = s_{n,n'} / \sqrt{s_{n,n} s_{n',n'}}$ and equivalently $\rho_{n,n'} = \sigma_{n,n'} / \sqrt{\sigma_{n,n} \sigma_{n',n'}}$. The average of the correlations is given by

$$\bar{r} = \frac{2}{(N-1)N} \sum_{n=1}^{N-1} \sum_{n'=n+1}^N r_{n,n'}$$

and

$$\bar{\rho} = \frac{2}{(N-1)N} \sum_{n=1}^{N-1} \sum_{n'=n+1}^N \rho_{n,n'}.$$

Define the constant correlation matrix \mathbf{G} , based on the real variances and real correlations, by

$$g_{n,n} = s_{n,n}, \quad g_{n,n'} = \bar{r} \sqrt{s_{n,n} s_{n',n'}}, \quad \text{for } n \neq n'.$$

Then, the estimator of \mathbf{G} , the sample constant correlation matrix $\mathbf{\Gamma} = (\gamma_{n,n'})$, is given by

$$\gamma_{n,n} = \sigma_{n,n}, \quad \gamma_{n,n'} = \bar{\rho} \sqrt{\sigma_{n,n} \sigma_{n',n'}}, \quad \text{for } n \neq n'.$$

We consider the Frobenius norm of the difference between the shrunken covariance matrix and the real covariance matrix. We want to minimize the quadratic loss function:

$$\|\delta^* \mathbf{G} + (1 - \delta^*) \mathbf{\Sigma} - \mathbf{S}\|^2.$$

Any value of $\delta^* \in [0, 1]$ is a trade off between the shrinkage target and the sample covariance matrix. Ledoit and Wolf use an estimation of δ^* . Ledoit and Wolf explain their approach, referring to another paper: 'Improved estimation of the covariance matrix of stock returns with an application to portfolio selection' [7]. We summarize some of the results of [6] and [7] in Theorem 3.3.1 below.

Before we formulate the theorem, note that for a parameter ψ , an estimator $\tilde{\psi}_T$, based on T data points, is said to be *consistent* if

$$\lim_{T \rightarrow \infty} \mathbb{P}(|\tilde{\psi}_T - \psi| \geq \epsilon) = 0, \quad \text{for all } \epsilon > 0.$$

Also, AsyVar denotes the asymptotic variance and AsyCov denotes the asymptotic covariance. For further detail of the definitions, we refer to [6] and [7].

Theorem 3.3.1 (Ledoit-Wolf, [6], [7]). *Define*

$$\begin{aligned} \pi &= \sum_{n=1}^N \sum_{n'=1}^N \text{AsyVar}(\sqrt{T} \sigma_{n,n'}) \\ \eta &= \sum_{n=1}^N \sum_{n'=1}^N \text{AsyCov}(\sqrt{T} \gamma_{n,n'}, \sqrt{T} \sigma_{n,n'}) \\ \tau &= \sum_{n=1}^N \sum_{n'=1}^N (g_{n,n'} - s_{n,n'})^2. \end{aligned}$$

Then, a consistent estimator for π is

$$\hat{\pi} = \sum_{n=1}^N \sum_{n'=1}^N \hat{\pi}_{n,n'}$$

with

$$\hat{\pi}_{n,n'} = \frac{1}{T} \sum_{t=1}^T ((\chi_{n,t} - \bar{\chi}_n)(\chi_{n',t} - \bar{\chi}_{n'}) - \sigma_{n,n'})^2.$$

A consistent estimator for $\text{AsyCov}(\sqrt{T}\sigma_{n,n}, \sqrt{T}\sigma_{n,n'})$ is given by

$$\hat{\theta}_{n,n,n,n'} = \frac{1}{T} \sum_{t=1}^T ((\chi_{n,t} - \bar{\chi}_n)^2 - \sigma_{n,n})(\chi_{n,t} - \bar{\chi}_n)(\chi_{n',t} - \bar{\chi}_{n'}) - \sigma_{n,n'}).$$

Analogously, a consistent estimator for $\text{AsyCov}(\sqrt{T}\sigma_{n',n'}, \sqrt{T}\sigma_{n,n'})$ is given by

$$\hat{\theta}_{n',n',n,n'} = \frac{1}{T} \sum_{t=1}^T ((\chi_{n',t} - \bar{\chi}_{n'})^2 - \sigma_{n',n'})(\chi_{n,t} - \bar{\chi}_n)(\chi_{n',t} - \bar{\chi}_{n'}) - \sigma_{n,n'}).$$

A consistent estimator for η is

$$\hat{\eta} = \sum_{n=1}^N \hat{\pi}_{n,n'} + \sum_{n=1}^N \sum_{n'=1, n' \neq n}^N \frac{\bar{\rho}}{2} \left(\sqrt{\frac{\sigma_{n',n'}}{\sigma_{n,n}}} \hat{\theta}_{n,n,n,n'} + \sqrt{\frac{\sigma_{n,n}}{\sigma_{n',n'}}} \hat{\theta}_{n',n',n,n'} \right)$$

and a consistent estimator for τ is

$$\hat{\tau} = \sum_{n=1}^N \sum_{n'=1}^N (\gamma_{n,n'} - \sigma_{n,n'})^2.$$

Furthermore, the shrinkage constant, defined by

$$\hat{\delta}^* = \max\{0, \min\{\hat{\kappa}/T, 1\}\}$$

with

$$\hat{\kappa} = \frac{\hat{\pi} - \hat{\eta}}{\hat{\tau}}$$

is estimated optimal.

We use the theorem above in our investigation on the robustness effect of shrinkage on the MV optimization procedure. We have $T = 36$ data points. We calculate

$$\begin{aligned} \bar{\rho} &= 0.3183 \\ \hat{\eta} &= 142746689 \\ \hat{\pi} &= 577540815 \\ \hat{\tau} &= 19523029 \\ \hat{\kappa} &= 22.2708 \\ \hat{\delta}^* &= 0.6363. \end{aligned}$$

This results in the shrunken covariance matrix given in Table 3.4. We see that we have only positive entries, since $\bar{\rho} > 0$.

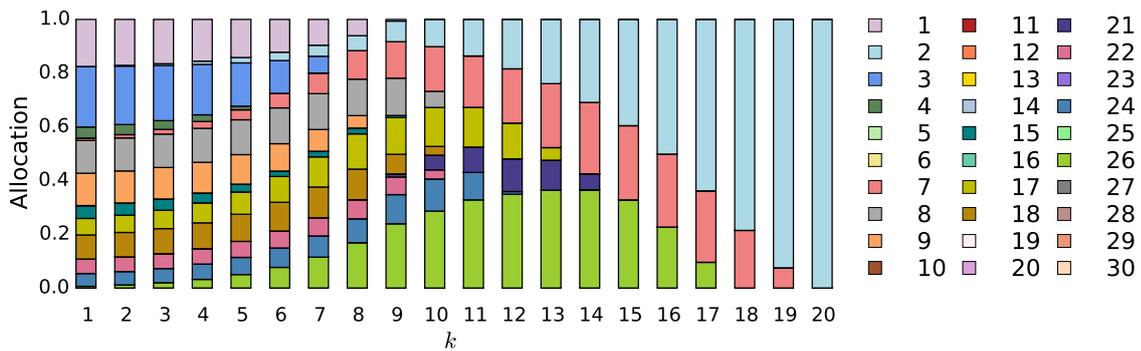
TABLE 3.4: Shrunk covariance matrix of the securities, divided by a factor of 100, rounded to 1 decimal.

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.	16.	17.	18.	19.	20.	21.	22.	23.	24.	25.	26.	27.	28.	29.	30.
1.	3.3	2.5	0.7	1.4	1.8	0.9	1.8	1.1	1.1	1.4	1.8	1.4	1.5	1.8	1.1	2.1	0.9	1.1	1.2	2.2	2.0	1.4	1.7	1.4	2.7	1.6	1.7	1.4	2.4	1.6
2.	2.5	23.0	1.7	5.6	4.2	1.6	6.2	2.2	2.6	3.4	6.6	5.0	3.4	3.4	2.4	6.0	3.1	1.8	3.4	3.2	3.5	2.1	3.7	2.7	5.7	3.4	6.2	3.7	4.6	3.4
3.	0.7	1.7	2.3	1.2	1.5	1.2	1.6	0.8	0.9	1.4	1.6	1.6	1.2	1.8	1.0	1.6	1.2	1.3	1.2	1.7	1.4	1.2	1.4	1.2	1.4	1.3	1.4	1.5	1.8	1.2
4.	1.4	5.6	1.2	8.8	2.7	1.4	3.8	1.6	1.6	1.3	2.3	2.7	1.7	2.0	1.0	2.3	1.9	1.4	2.1	2.2	1.6	1.5	2.5	2.2	3.4	1.5	2.8	2.5	2.4	1.4
5.	1.8	4.2	1.5	2.7	6.6	1.5	4.0	1.9	1.9	2.4	3.3	3.0	2.4	2.6	1.7	3.4	2.4	1.5	2.1	2.7	2.7	1.5	2.5	2.0	3.5	1.9	2.9	2.7	3.1	1.6
6.	0.9	1.6	1.2	1.4	1.5	3.0	1.2	1.0	0.9	1.3	1.6	1.4	1.3	1.6	1.2	1.7	1.4	1.6	1.2	2.3	1.7	1.6	1.6	1.8	1.8	1.4	1.4	1.4	1.9	1.5
7.	1.8	6.2	1.6	3.8	4.0	1.2	14.7	2.6	2.9	3.0	3.7	4.4	3.0	3.7	2.1	3.8	3.1	0.6	3.4	1.8	2.7	0.8	2.5	1.2	3.4	2.2	3.9	3.5	2.3	1.5
8.	1.1	2.2	0.8	1.6	1.9	1.0	2.6	3.1	1.4	1.4	1.6	1.6	1.7	1.6	1.3	2.0	1.4	0.9	1.3	1.7	1.8	0.9	1.3	1.1	2.0	1.1	1.7	1.5	1.6	1.0
9.	1.1	2.6	0.9	1.6	1.9	0.9	2.9	1.4	3.7	1.8	2.0	2.6	1.9	2.6	1.7	2.3	1.5	0.7	2.0	1.6	2.0	0.7	1.4	0.8	1.9	2.0	1.8	2.0	1.9	1.4
10.	1.4	3.4	1.4	1.3	2.4	1.3	3.0	1.4	1.8	6.1	3.3	3.3	2.2	2.9	1.9	3.6	1.5	1.3	2.0	2.7	2.5	1.5	2.0	1.9	2.5	2.4	2.4	2.4	2.6	2.2
11.	1.8	6.6	1.6	2.3	3.3	1.6	3.7	1.6	2.0	3.3	11.7	4.0	2.8	3.1	2.2	5.5	3.0	1.6	2.0	2.9	3.3	2.1	2.9	2.4	4.4	2.5	3.4	2.8	3.3	2.8
12.	1.4	5.0	1.6	2.7	3.0	1.4	4.4	1.6	2.6	3.3	4.0	9.2	3.0	4.2	2.2	3.8	2.2	1.2	3.0	2.5	2.4	1.0	2.5	1.3	2.9	3.4	3.4	3.6	2.9	2.0
13.	1.5	3.4	1.2	1.7	2.4	1.3	3.0	1.7	1.9	2.2	2.8	3.0	5.3	3.2	1.9	3.0	1.9	1.2	2.2	2.5	2.6	1.2	1.9	1.3	2.6	2.6	2.3	2.3	2.7	2.0
14.	1.8	3.4	1.8	2.0	2.6	1.6	3.7	1.6	2.6	2.9	3.1	4.2	3.2	9.2	2.6	3.4	2.1	1.4	2.9	3.0	3.0	1.4	2.3	1.5	2.9	3.7	2.8	3.2	3.0	2.5
15.	1.1	2.4	1.0	1.0	1.7	1.2	2.1	1.3	1.7	1.9	2.2	2.2	1.9	2.6	4.3	2.5	1.7	1.1	1.8	2.2	2.4	1.1	1.3	1.4	1.9	2.4	2.0	1.9	2.2	1.6
16.	2.1	6.0	1.6	2.3	3.4	1.7	3.8	2.0	2.3	3.6	5.5	3.8	3.0	3.4	2.5	13.4	2.0	2.1	2.4	3.1	3.5	2.1	2.9	2.1	4.0	3.1	3.1	2.4	3.8	2.9
17.	0.9	3.1	1.2	1.9	2.4	1.4	3.1	1.4	1.5	1.5	3.0	2.2	1.9	2.1	1.7	2.0	5.5	0.9	1.8	1.5	2.3	1.2	1.9	1.7	2.2	1.7	2.4	2.2	2.0	1.2
18.	1.1	1.8	1.3	1.4	1.5	1.6	0.6	0.9	0.7	1.3	1.6	1.2	1.2	1.4	1.1	2.1	0.9	4.4	0.7	2.8	1.6	2.1	1.6	2.2	2.0	1.6	1.6	1.2	2.6	1.7
19.	1.2	3.4	1.2	2.1	2.1	1.2	3.4	1.3	2.0	2.0	2.0	3.0	2.2	2.9	1.8	2.4	1.8	0.7	4.8	1.6	2.2	0.7	1.7	1.0	1.7	2.5	2.5	2.4	2.0	1.5
20.	2.2	3.2	1.7	2.2	2.7	2.3	1.8	1.7	1.6	2.7	2.9	2.5	2.5	3.0	2.2	3.1	1.5	2.8	1.6	9.5	2.8	3.3	2.7	2.9	4.0	2.8	2.8	2.4	4.2	3.0
21.	2.0	3.5	1.4	1.6	2.7	1.7	2.7	1.8	2.0	2.5	3.3	2.4	2.6	3.0	2.4	3.5	2.3	1.6	2.2	2.8	7.8	1.7	2.0	2.3	3.7	2.5	2.9	2.5	3.0	2.1
22.	1.4	2.1	1.2	1.5	1.5	1.6	0.8	0.9	0.7	1.5	2.1	1.0	1.2	1.4	1.1	2.1	1.2	2.1	0.7	3.3	1.7	5.6	1.9	2.4	2.7	1.4	1.5	1.2	2.8	2.0
23.	1.7	3.7	1.4	2.5	2.5	1.6	2.5	1.3	1.4	2.0	2.9	2.5	1.9	2.3	1.3	2.9	1.9	1.6	1.7	2.7	2.0	1.9	5.1	1.9	3.2	2.0	2.7	2.1	3.2	2.0
24.	1.4	2.7	1.2	2.2	2.0	1.8	1.2	1.1	0.8	1.9	2.4	1.3	1.3	1.5	1.4	2.1	1.7	2.2	1.0	2.9	2.3	2.4	1.9	7.6	2.8	1.5	1.7	1.3	2.0	2.1
25.	2.7	5.7	1.4	3.4	3.5	1.8	3.4	2.0	1.9	2.5	4.4	2.9	2.6	2.9	1.9	4.0	2.2	2.0	1.7	4.0	3.7	2.7	3.2	2.8	10.9	2.1	3.4	2.8	4.1	2.5
26.	1.6	3.4	1.3	1.5	1.9	1.4	2.2	1.1	2.0	2.4	2.5	3.4	2.6	3.7	2.4	3.1	1.7	1.6	2.5	2.8	2.5	1.4	2.0	1.5	2.1	7.6	3.0	2.6	3.1	2.2
27.	1.7	6.2	1.4	2.8	2.9	1.4	3.9	1.7	1.8	2.4	3.4	3.4	2.3	2.8	2.0	3.1	2.4	1.6	2.5	2.8	2.9	1.5	2.7	1.7	3.4	3.0	8.9	3.1	3.3	1.9
28.	1.4	3.7	1.5	2.5	2.7	1.4	3.5	1.5	2.0	2.4	2.8	3.6	2.3	3.2	1.9	2.4	2.2	1.2	2.4	2.4	2.5	1.2	2.1	1.3	2.8	2.6	3.1	6.4	2.9	1.5
29.	2.4	4.6	1.8	2.4	3.1	1.9	2.3	1.6	1.9	2.6	3.3	2.9	2.7	3.0	2.2	3.8	2.0	2.6	2.0	4.2	3.0	2.8	3.2	2.0	4.1	3.1	3.3	2.9	10.5	2.8
30.	1.6	3.4	1.2	1.4	1.6	1.5	1.5	1.0	1.4	2.2	2.8	2.0	2.0	2.5	1.6	2.9	1.2	1.7	1.5	3.0	2.1	2.0	2.0	2.1	2.5	2.2	1.9	1.5	2.8	5.5

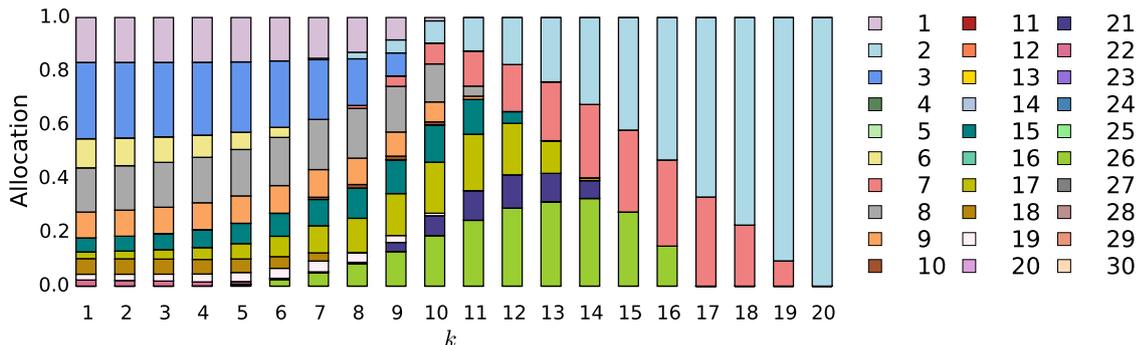
We compute the optimal portfolios, based on the sample covariance matrix, and we compute the optimal portfolios, using the shrunk covariance matrix; abbreviate the latter to *shrunk MV solutions* or *shrunk MV portfolios*. We use risk aversion parameters: $\lambda_k = ((k - 1)/20 + 1)^{-10}$ for $k = 1, \dots, 20$. The allocations of the two methods are given in Figure 3.2.

FIGURE 3.2: Allocations of the original MV solutions and the shrunk MV solutions, for risk aversion parameters numbered by $k = 1, \dots, 20$.

Original MV solutions



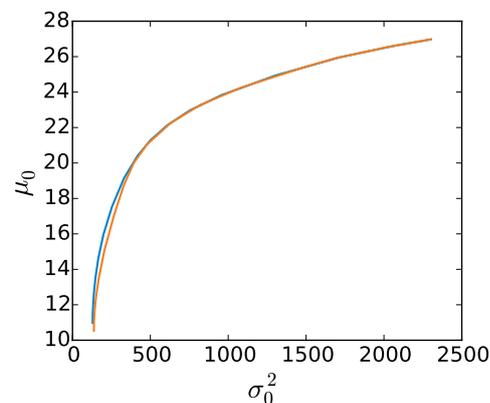
Shrunk MV solutions



We see that the allocations do not differ much; generally, the same securities appear in the solutions. Also, in both methods, we see that security 2 (TJX Cos) dominates the portfolios of low risk aversion. However, we see that there are a few differences. Some securities do appear in the original MV solutions but not in the shrunken MV solutions and vice versa. Furthermore, for high risk aversion levels, the original MV portfolios seem to be more a bit more diversified than the shrunken MV portfolios.

Let us see how the portfolios differ in terms of expected risk and return levels. The frontiers of the two methods are plotted in Figure 3.3. The blue line represents the efficient frontier of original MV optimization and the orange line represents the efficient frontier of MV optimization method, based on the shrunken covariance matrix; abbreviate the latter to *shrunken MV optimization*. Note that we plot the risk and return levels based on the original input parameters, μ and Σ , so the efficient frontier of shrunken MV optimization can not surpass the original efficient frontier. We see that the difference between the two is small.

FIGURE 3.3: The efficient frontier of original MV optimization (blue) and the efficient frontier of shrunken MV optimization (orange).



Summarizing the results, we see that shrunken and original MV solutions do not differ much in allocations. The shrunken MV solutions are a bit less diversified and for high risk aversion levels they perform slightly worse on the expected risk and return levels. In the next section, we investigate whether or not shrinkage results in more robust portfolios.

3.3.2 The robustness effect of shrinkage

We want to investigate the influence of the constant correlation matrix on the robustness of the MV optimal portfolios. Therefore, we suppose we know the ‘real’ returns μ , the ‘real’ covariance matrix Σ , and the ‘real’ optimal portfolios. We vary the input parameters by generating a large amount of data points. Now, we can calculate the solutions, based on the variation of input parameters and compare them to the ‘real’ solutions. If we have robust portfolios, the solutions based on the variation of input parameters should be close to the ‘real’ solutions.

We like to compare the robustness of two different methods: original MV optimization and shrunken MV optimization. We compute the difference between the ‘real’ original MV solutions and the MV solutions based on the variation of input parameters and we compute the difference between the ‘real’ original MV solutions and the shrunken MV solutions, based on the same variation of input parameters. Now, shrinkage is more

robust if the latter difference is smaller than the difference corresponding to original MV optimization.

Let us make the procedure we describe above more explicit. We have the $T = 36$ data points from $N = 30$ securities. We assume that parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, based on these data, are the ‘real’ parameters. We want to obtain M variations of input parameters. For each variation $m = 1, \dots, M$, we generate data by drawing T_{inp} times from the ‘real’ distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. We obtain a return matrix $\mathbf{X} = (\chi_{n,t}^m)$ and calculate the sample means $\boldsymbol{\mu}^{\text{inp}_m}$ and the sample covariance matrix $\boldsymbol{\Sigma}^{\text{inp}_m}$, for $n, n' = 1, \dots, N$, as follows:

$$\boldsymbol{\mu}^{\text{inp}_m \top} = [\mu_1^{\text{inp}_m}, \dots, \mu_N^{\text{inp}_m}] = \left[\frac{1}{T_{\text{inp}}} \sum_{t=1}^{T_{\text{inp}}} \chi_{1,t}^m, \dots, \frac{1}{T_{\text{inp}}} \sum_{t=1}^{T_{\text{inp}}} \chi_{N,t}^m \right]$$

$$\boldsymbol{\Sigma}^{\text{inp}_m} = (\sigma_{n,n'}^{\text{inp}_m}), \text{ with } \sigma_{n,n'}^{\text{inp}_m} = \frac{1}{T_{\text{inp}}} \sum_{t=1}^{T_{\text{inp}}} (\chi_{n,t}^m - \mu_n^{\text{inp}_m})(\chi_{n',t}^m - \mu_{n'}^{\text{inp}_m}).$$

Following the shrinkage procedure of Ledoit and Wolf, we also obtain the shrunken covariance matrix: $\hat{\delta}^{*\text{inp}_m} \boldsymbol{\Gamma}^{\text{inp}_m} + (1 - \hat{\delta}^{*\text{inp}_m}) \boldsymbol{\Sigma}^{\text{inp}_m}$. Let \mathbf{w}^{orig} denote the solution of original MV optimization procedure, with the sample means and covariance matrix that are based on the original data. Hence, \mathbf{w}^{orig} is the ‘real’ optimal portfolio, which we are looking for. Let \mathbf{w}^{shr} denote the solution of the shrunken MV optimization procedure. Let $\mathbf{w}^{\text{inp}_m \text{orig}}$ denote the solution of the original MV optimization procedure, based on the input parameters corresponding to variation m . Lastly, let $\mathbf{w}^{\text{inp}_m \text{shr}}$ denote the corresponding solution of the shrunken MV optimization procedure. Table 3.5 gives an overview of the notation.

TABLE 3.5: Notational overview.

	MV optimization		Shrunken MV optimization	
	Input	Solution	Input	Solution
Original	$\boldsymbol{\mu}, \boldsymbol{\Sigma}$	\mathbf{w}^{orig}	$\boldsymbol{\mu}, \hat{\delta}^* \boldsymbol{\Gamma} + (1 - \hat{\delta}^*) \boldsymbol{\Sigma}$	\mathbf{w}^{shr}
Modified	$\boldsymbol{\mu}^{\text{inp}_1}, \boldsymbol{\Sigma}^{\text{inp}_1}$	$\mathbf{w}^{\text{inp}_1 \text{orig}}$	$\boldsymbol{\mu}^{\text{inp}_1}, \hat{\delta}^{*\text{inp}_1} \boldsymbol{\Gamma}^{\text{inp}_1} + (1 - \hat{\delta}^{*\text{inp}_1}) \boldsymbol{\Sigma}^{\text{inp}_1}$	$\mathbf{w}^{\text{inp}_1 \text{shr}}$
	\vdots	\vdots	\vdots	\vdots
	$\boldsymbol{\mu}^{\text{inp}_M}, \boldsymbol{\Sigma}^{\text{inp}_M}$	$\mathbf{w}^{\text{inp}_M \text{orig}}$	$\boldsymbol{\mu}^{\text{inp}_M}, \hat{\delta}^{*\text{inp}_M} \boldsymbol{\Gamma}^{\text{inp}_M} + (1 - \hat{\delta}^{*\text{inp}_M}) \boldsymbol{\Sigma}^{\text{inp}_M}$	$\mathbf{w}^{\text{inp}_M \text{shr}}$

To investigate whether the shrunken or the original MV optimization procedure is more robust, we calculate $\mathbf{w}^{\text{inp}_1 \text{orig}}, \dots, \mathbf{w}^{\text{inp}_M \text{orig}}$ and $\mathbf{w}^{\text{inp}_1 \text{shr}}, \dots, \mathbf{w}^{\text{inp}_M \text{shr}}$ and we calculate their distances to the ‘real’ solution \mathbf{w}^{orig} . We let the number of data points vary: take $T_{\text{inp}} = 20, 50, 100, 200, 500$. Use again the Euclidean distance

$$d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}_m \text{orig}}) = \sqrt{\sum_{n=1}^N (w_n^{\text{orig}} - w_n^{\text{inp}_m \text{orig}})^2}$$

$$d(\mathbf{w}^{\text{orig}}, \mathbf{w}^{\text{inp}_m \text{shr}}) = \sqrt{\sum_{n=1}^N (w_n^{\text{orig}} - w_n^{\text{inp}_m \text{shr}})^2}.$$

For $\boldsymbol{\beta}_m = \mathbf{w}^{\text{inp}_m \text{orig}}, \mathbf{w}^{\text{inp}_m \text{shr}}$, the average is given by

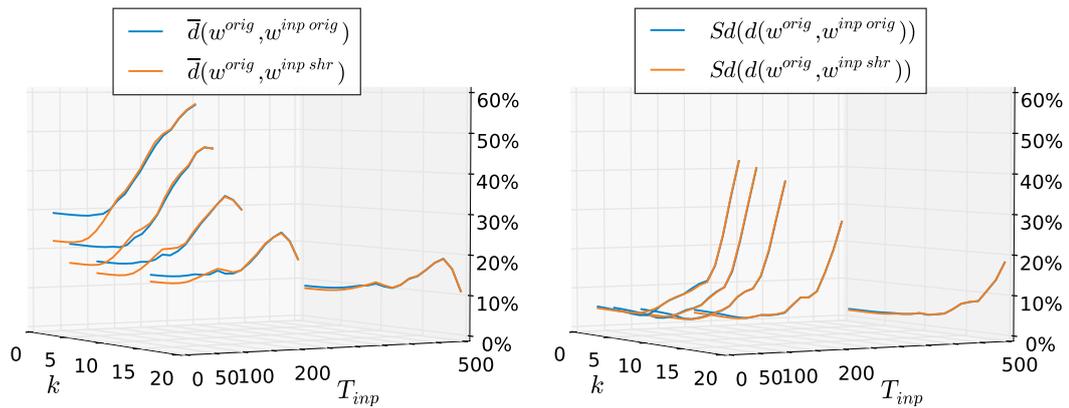
$$\bar{d}(\mathbf{w}^{\text{orig}}, \boldsymbol{\beta}) = \frac{1}{M} \sum_{m=1}^M d(\mathbf{w}^{\text{orig}}, \boldsymbol{\beta}_m).$$

We measure the spread by the standard deviation of the distances:

$$\text{Sd}(d(\mathbf{w}^{\text{orig}}, \boldsymbol{\beta})) = \sqrt{\frac{1}{M} \sum_{m=1}^M (d(\mathbf{w}^{\text{orig}}, \boldsymbol{\beta}_m) - \bar{d}(\mathbf{w}^{\text{orig}}, \boldsymbol{\beta}))^2}.$$

Figure 3.4 gives an overview of the results. Note that we scale the results to a maximum distance of $\sqrt{2}$, corresponding to 100%, just like we do in Chapter 1 and Chapter 2.

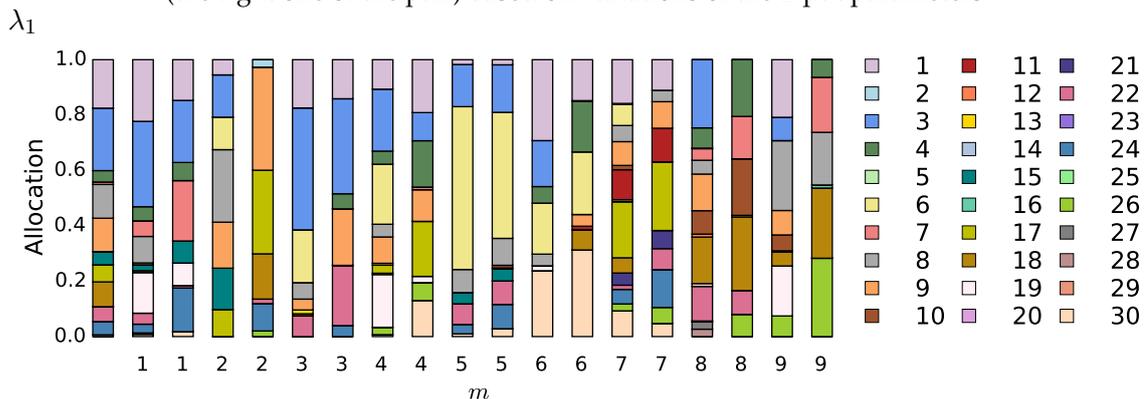
FIGURE 3.4: Robustness performances per risk aversion parameter numbered by $k = 1, \dots, 20$ and with $M = 200$.

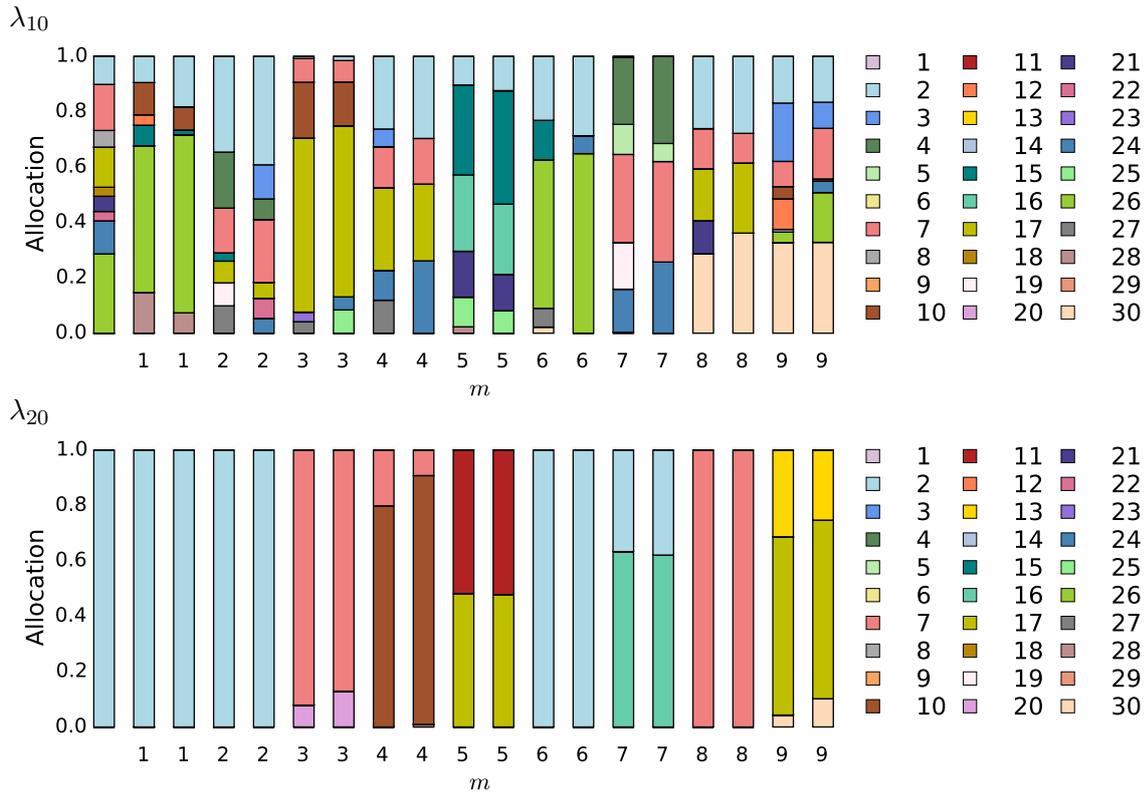


We see that for T_{inp} up till 200 and k approximately up till 7, shrinkage is more robust. The results are comparable for larger risk aversion parameters and for larger values of T_{inp} . The standard deviations of the two methods are comparable for each k and T_{inp} .

We also see that in both methods we have more robust portfolios for high risk aversion levels and we have less robust portfolios for low risk aversion levels. We explain in Chapter 2 that high risk aversion portfolios are usually diversified and more robust. For low risk aversion, we require the highest possible return and therefore we will put all the weight in the asset class with the highest expected return. In each variation m this could be another asset class, so the low risk aversion portfolios will differ a lot per variation. We show this in Figure 3.5, by computing $M = 9$ runs, for λ_1 , λ_{10} , and λ_{20} . The far left portfolio is ‘real’ optimal portfolio, based on the original input parameters. For each variation, we show the solution based on the shrunken covariance matrix (the left one of the pair) and we show the original MV solution (the right one of the pair).

FIGURE 3.5: Allocations of the real solutions (the far left bar) and of shrunken MV solutions (the left one of the pair) and original MV solutions (the right one of the pair) based on variations of the input parameters.





We see indeed that the high risk aversion portfolios are diversified and low risk aversion portfolios have extreme allocations, with alternately high weight in different assets. So therefore, we have high robustness for high risk aversion levels and low robustness for low risk aversion levels. However, from Figure 3.5 it is not clear why the shrinkage method performs better than MV optimization, for high risk aversion levels.

We go back to analyze Figure 3.4. We see that the difference between the shrunken and the original MV procedure decreases as T_{inp} increases. Hence, our expectation was correct: linear shrinkage is not helpful when we have a lot of data points. We decide that this method is not useful to us. However, we do complete our analysis below.

We expect that the robustness of the portfolios is due to the robustness of the constant correlation covariance matrix. We investigate the robustness of the covariance matrices as follows. We compare both the sample covariance matrix Σ^{inp_m} and the shrunken covariance matrix $\hat{\delta}^{\text{inp}_m} \mathbf{\Gamma}^{\text{inp}_m} + (1 - \hat{\delta}^{\text{inp}_m}) \Sigma^{\text{inp}_m}$ to the original sample covariance matrix Σ . We calculate the distance between two matrices using the Frobenius norm. We compute the distance between two matrices, the average distance over all variations, and standard deviation of the distances as follows. For $\tilde{\Sigma}^{\text{inp}_m} = \Sigma^{\text{inp}_m}, \hat{\delta}^{\text{inp}_m} \mathbf{\Gamma}^{\text{inp}_m} + (1 - \hat{\delta}^{\text{inp}_m}) \Sigma^{\text{inp}_m}$:

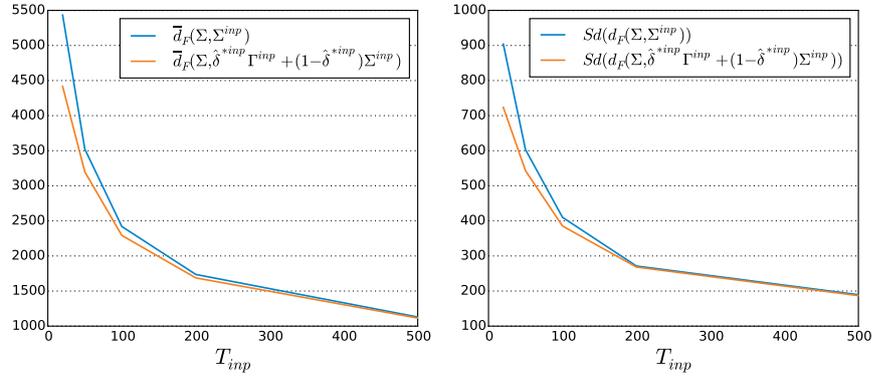
$$d_{\text{F}}(\Sigma, \tilde{\Sigma}^{\text{inp}_m}) = \sqrt{\sum_{n=1}^N \sum_{n'=1}^N (\sigma_{n,n'} - \tilde{\sigma}_{n,n'}^{\text{inp}_m})^2}$$

$$\bar{d}_{\text{F}}(\Sigma, \tilde{\Sigma}^{\text{inp}}) = \frac{1}{M} \sum_{m=1}^M d_{\text{F}}(\Sigma, \tilde{\Sigma}^{\text{inp}_m})$$

$$\text{Sd}(d_{\text{F}}(\Sigma, \tilde{\Sigma}^{\text{inp}})) = \sqrt{\frac{1}{M} \sum_{m=1}^M (d_{\text{F}}(\Sigma, \tilde{\Sigma}^{\text{inp}_m}) - \bar{d}_{\text{F}}(\Sigma, \tilde{\Sigma}^{\text{inp}}))^2}$$

Figure 3.6 shows the results of $M = 200$ iterations. Note that we do not scale the distance to a maximum here.

FIGURE 3.6: Average and standard deviation of the distances between the shrinkage parameters based on $M = 200$ variations and the real covariance matrix Σ .



The difference between the two is small, but we see that the constant correlation matrix is indeed more robust.

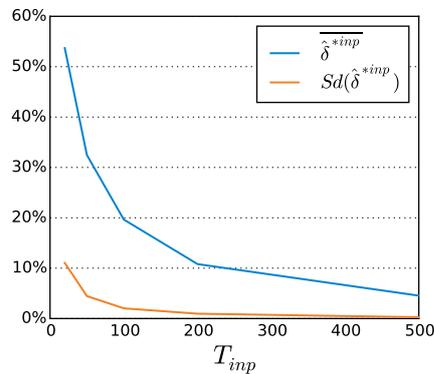
Lastly, let us investigate how the shrinkage parameter behaves. Figure 3.7 shows the average and the standard deviation of the shrinkage parameters $\hat{\delta}^{*inp_1}, \dots, \hat{\delta}^{*inp_M}$:

$$\overline{\hat{\delta}^{*inp}} = \frac{1}{M} \sum_{m=1}^M \hat{\delta}^{*inp_m}$$

$$Sd(\hat{\delta}^{*inp}) = \sqrt{\frac{1}{M} \sum_{m=1}^M (\hat{\delta}^{*inp_m} - \overline{\hat{\delta}^{*inp}})^2}.$$

We scale to a maximum distance of 1, corresponding to 100%, since $\hat{\delta}^{*inp_m} \in [0, 1]$ for all $m = 1, \dots, M$.

FIGURE 3.7: The average and standard deviation of the shrinkage parameters of $M = 200$ variations.



We see that for T_{inp} big, the shrinkage parameter becomes small. That explains why the difference in robustness between shrunken and original MV optimization decreases, as T_{inp} increases. We argue that for larger values of T_{inp} , the sample covariance matrix is a better estimator of the original 'real' covariance matrix, hence, we shrink less towards the target.

3.3.2.1 Conclusion

Shrinkage can be useful to make the MV optimization method more robust, but only when there are many assets (or asset classes) under consideration and only a few historical data points available. We can see this in Figure 3.4, when we compare the results corresponding to a few data points to the results corresponding to a lot of data points. If we consider the case of only a few data points, we see that the robustness effect can differ per risk aversion level. Furthermore, we see in Figure 3.3 that using the constant correlation covariance matrix results in portfolios that are close to the efficient frontier. Hence, we barely give in on the expected portfolio return and expected portfolio risk.

In our case, we have many data points available. For this reason, we do not look into the shrinkage method any further.

Chapter 4

Maximum distance optimization

In this section, we develop a method to obtain a set of portfolios that are all near optimal and call this maximum distance optimization. Mean-Variance optimization provides us with a range of portfolios that define the efficient frontier and now we want to know what kind of portfolios lie closely beneath that efficient frontier. Can we invest in totally different asset classes than the MV optimal portfolios do and still have a return and risk level that are close to optimal?

We start with an MV optimal portfolio that corresponds to our risk aversion level and search for another portfolio, of which we require that it has approximately the same risk and return level, but is totally different in terms of allocations. Next, we search for another portfolio that also has approximately the same risk and return level, but again has a different set of weights than both the MV optimal portfolio and the firstly obtained portfolio. We continue this process and each time we obtain a different portfolio that is near optimal.

Given such a set of different portfolios that all meet the risk and return level requirements, we can construct yet another portfolio, by taking a convex combination of the obtained portfolios. We prove that this convex combination portfolio also meets the risk and return level requirements. This means that this method provides us with uncountable infinitely many solutions, which all have an acceptable risk and return level.

The idea is now that we can select a preferred portfolio from the near optimal portfolios using qualitative arguments that are potentially difficult to incorporate in the optimization process. For example, huge transaction costs might be involved by taking a position in a certain asset class, or we have less confidence in the data used to model one of the asset classes. This practice suits well with how models are used: they are not assumed to represent the absolute truth, but used as one source of input when making decisions.

Manually choosing any mix we like is an advantage, which we did not encounter in other methods and as far as we know, it has not been published before. In resampling, for example, we do have some influence on the outcome by choosing values for the parameters, but the outcome for a certain risk aversion level is merely one portfolio. The method that we define in this chapter provides us with a whole scope of portfolios.

In this chapter, we first define the optimization program we like to solve, then we explain what Support Vector Machines are and show how Support Vector Machines enable us to implement the required optimization program. For this implementation, we use a heuristic method called Basin hopping and analyze the results.

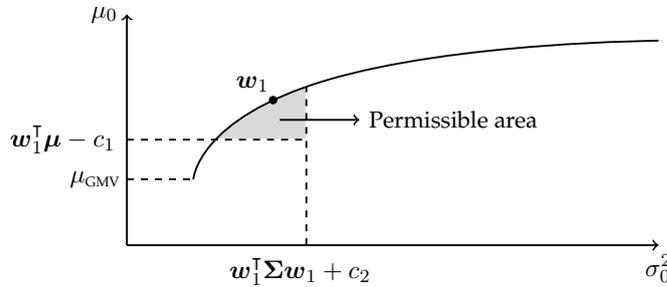
4.1 Maximum distance towards convex hull

In this section, we define the optimization program we like to solve. Unless stated otherwise, we assume we are not allowed to go short. Suppose that λ_k is our risk aversion parameter. Let \mathbf{w}_1 be the MV optimal solution corresponding to λ_k . We want to find a new portfolio that is far from \mathbf{w}_1 in terms of Euclidean distance between the two vectors, but it should also be close in terms of the expected risk and expected return level. Let $c_1, c_2 \in \mathbb{R}_{>0}$. We consider the following optimization program:

$$\begin{aligned} & \max_{\mathbf{w}_2 \in \mathbb{R}^N} \|\mathbf{w}_2 - \mathbf{w}_1\| \\ & \text{subject to } \mathbf{w}_2 \geq \mathbf{0} \\ & \mathbf{w}_2^\top \mathbf{1} = 1 \\ & \mathbf{w}_2^\top \boldsymbol{\mu} \geq \mathbf{w}_1^\top \boldsymbol{\mu} - c_1 \\ & \mathbf{w}_2^\top \boldsymbol{\Sigma} \mathbf{w}_2 \leq \mathbf{w}_1^\top \boldsymbol{\Sigma} \mathbf{w}_1 + c_2. \end{aligned}$$

Recall that μ_{GMV} denotes the global minimum expected return. Parameter c_1 determines to which extent the new return level may deviate from the return of \mathbf{w}_1 . We need c_1 to be less than or equal to $\mathbf{w}_1^\top \boldsymbol{\mu} - \mu_{\text{GMV}}$. Parameter c_2 determines to which extent the new risk level may deviate from the risk of \mathbf{w}_1 . Here, we need $\mathbf{w}_1^\top \boldsymbol{\Sigma} \mathbf{w}_1 + c_2$ to not exceed the maximum possible risk. Define the *permissible area* to be the area that is determined by the last two constraints of the optimization program above. Figure 4.1 gives a graphical representation.

FIGURE 4.1: Graphical representation of the permissible area.



By executing the optimization program above, we obtain \mathbf{w}_2 . Finding one other portfolio is insightful, but we want to repeat this procedure more often to make sure we explore the permissible area thoroughly. We want to do $\mathcal{I} \in \mathbb{N}$ iterations, each time searching for a portfolio that is at maximum distance towards the convex hull of the already obtained portfolios.

Let $i = 2, \dots, \mathcal{I}$, then the convex hull of portfolios $\mathbf{w}_1, \dots, \mathbf{w}_{i-1}$ is defined as

$$\begin{aligned} \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1}) = \{ \mathbf{w} \in \mathbb{R}^N : \mathbf{w} = \alpha_1 \mathbf{w}_1 + \dots + \alpha_{i-1} \mathbf{w}_{i-1}, \\ \alpha_1, \dots, \alpha_{i-1} \in \mathbb{R}_{\geq 0}, \\ \alpha_1 + \dots + \alpha_{i-1} = 1 \}. \end{aligned}$$

For $A \subset \mathbb{R}^N$ and $\mathbf{x} \in \mathbb{R}^N$ we denote the distance between \mathbf{x} and A by

$$\|\mathbf{x} - A\| = \inf \{ \|\mathbf{x} - \mathbf{a}\| : \mathbf{a} \in A \}.$$

Recall that if A is closed and non-empty, then

$$\|\mathbf{x} - A\| = \min\{\|\mathbf{x} - \mathbf{a}\| : \mathbf{a} \in A\}.$$

For $i = 2, \dots, \mathcal{I}$, we want to solve the following optimization program:

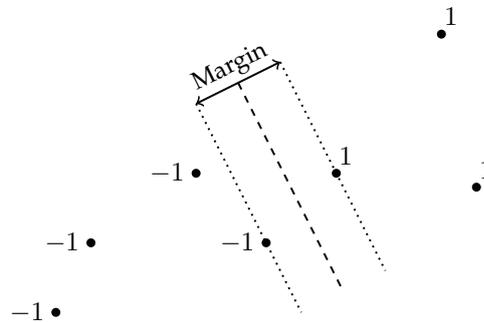
$$\begin{aligned} & \max_{\mathbf{w}_i \in \mathbb{R}^N} \|\mathbf{w}_i - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\| \\ & \text{subject to } \mathbf{w}_i \geq \mathbf{0} \\ & \mathbf{w}_i^\top \mathbf{1} = 1 \\ & \mathbf{w}_i^\top \boldsymbol{\mu} \geq \mathbf{w}_1^\top \boldsymbol{\mu} - c_1 \\ & \mathbf{w}_i^\top \boldsymbol{\Sigma} \mathbf{w}_i \leq \mathbf{w}_1^\top \boldsymbol{\Sigma} \mathbf{w}_1 + c_2. \end{aligned} \tag{4.1}$$

Unfortunately, we are not able to implement optimization program (4.1) in this form, because there is no straightforward way to calculate the distance towards the convex hull of the obtained solutions. However, Support Vector Machines enable us to implement this optimization program in another way. In the next section, we discuss what Support Vector Machines are and in the section thereafter, we show how we can use Support Vector Machines to implement optimization program (4.1).

4.2 Support Vector Machines

Support Vector Machines (SVM) is a machine learning algorithm which is used for classification, among other purposes. Let us briefly explain what it does. Suppose we have a set of \mathcal{I} points in \mathbb{R}^N , $N \in \mathbb{N}$: $\mathbf{w}_1, \dots, \mathbf{w}_{\mathcal{I}}$. We assume that the set is divided in two classes. One part of the set is associated to class '1', the other part is associated to class '-1'. We keep track of the classes by defining $y_i \in \{-1, 1\}$ to be the class of point \mathbf{w}_i , with $i = 1, \dots, \mathcal{I}$. We want to find a hyperplane that separates the points associated with '1' from the points associated with '-1'. Furthermore, we want the hyperplane to maximize the margin between these two classes. We give a formal definition of the margin below. Figure 4.2 shows a graphical representation of SVM.

FIGURE 4.2: Graphical representation of SVM.



We define the following optimization program:

$$\begin{aligned} & \min_{\mathbf{x} \in \mathbb{R}^N, z \in \mathbb{R}} \|\mathbf{x}\| \\ & \text{subject to } y_i(\mathbf{w}_i^\top \mathbf{x} + z) \geq 1, \text{ for all } i = 1, \dots, \mathcal{I}. \end{aligned} \tag{4.2}$$

Suppose it is possible to separate the two classes by a hyperplane $\{\mathbf{w} \in \mathbb{R}^N : \mathbf{w}^\top \mathbf{v} = a\}$ for some $\mathbf{v} \in \mathbb{R}^N$ and $a \in \mathbb{R}$. Scale \mathbf{v} and a to $\mathbf{x} \in \mathbb{R}^N$ and $z \in \mathbb{R}$ such that $y_i(\mathbf{w}_i^\top \mathbf{x} + z) \geq 1$, for all $i = 1, \dots, \mathcal{I}$. A vector, \mathbf{w}_i , for which either $\mathbf{w}_i^\top \mathbf{x} + z = 1$, or $\mathbf{w}_i^\top \mathbf{x} + z = -1$, is called a *support vector*. The *margin* associated to \mathbf{x} and z is defined by $2/\|\mathbf{x}\|$. Why this is the case, becomes clear in the following lemma.

Lemma 4.2.1. Let $\mathbf{x} \in \mathbb{R}^N$ and $z \in \mathbb{R}$. The distance between the hyperplanes, $\mathbf{w}^\top \mathbf{x} + z = 1$ and $\mathbf{w}^\top \mathbf{x} + z = -1$, equals $2/\|\mathbf{x}\|$.

Proof. The distance from $\mathbf{w}^\top \mathbf{x} + z = 1$ to the origin is $(z - 1)/\|\mathbf{x}\|$. The distance from $\mathbf{w}^\top \mathbf{x} + z = -1$ to the origin is $(z + 1)/\|\mathbf{x}\|$. So, the margin is

$$\frac{z + 1}{\|\mathbf{x}\|} - \frac{z - 1}{\|\mathbf{x}\|} = \frac{2}{\|\mathbf{x}\|}.$$

□

We explain how optimization program (4.2) separates the two classes in the next lemma.

Lemma 4.2.2. Let $\mathbf{w}_1, \dots, \mathbf{w}_{\mathcal{I}} \in \mathbb{R}^N$, with $y_i \in \{-1, 1\}$ for all $i = 1, \dots, \mathcal{I}$. Suppose it is possible to separate the classes by a hyperplane. Let $\mathbf{x} \in \mathbb{R}^N$ and $z \in \mathbb{R}$ form a solution to optimization program (4.2). Then $\mathbf{w}^\top \mathbf{x} + z = 0$ is the separating hyperplane that maximizes the margin.

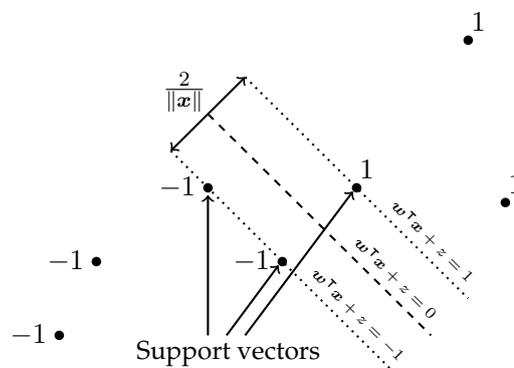
Proof. We first show that $\mathbf{w}^\top \mathbf{x} + z = 0$ separates the classes and then we show that $\mathbf{w}^\top \mathbf{x} + z = 0$ maximizes the margin. We check whether all points associated with '1' are on the one side of the hyperplane and all points associated with '-1' are on the other side, i.e.:

$$\begin{aligned} \mathbf{w}_i^\top \mathbf{x} + z &\geq 0, \text{ for all } i \text{ with } y_i = 1 \\ \mathbf{w}_i^\top \mathbf{x} + z &\leq 0, \text{ for all } i \text{ with } y_i = -1. \end{aligned}$$

These inequalities hold since we have $y_i(\mathbf{w}_i^\top \mathbf{x} + z) \geq 1$ for all i . To show that $\mathbf{w}^\top \mathbf{x} + z = 0$ maximizes the margin, use the previous lemma and note that maximizing $2/\|\mathbf{x}\|$ means minimizing $\|\mathbf{x}\|$. □

Figure 4.3 gives a graphical representation of the definitions above.

FIGURE 4.3: Graphical representation of SVM.



4.3 Formulation of maximum distance optimization

We make the following connection of optimization program (4.1) to SVM. Again, we have w_1 as the MV optimal portfolio corresponding to our risk aversion parameter. We want to find a new portfolio w_2 that is far from w_1 in terms of the Euclidean distance between the two vectors but still in the permissible area determined by c_1 and c_2 . Let $y_1 = 1$ and $y_2 = -1$. Then we search for w_2 such that

$$\begin{aligned} & \min_{\substack{x_2, w_2 \in \mathbb{R}^N \\ z_2 \in \mathbb{R}}} \|x_2\| \\ & \text{subject to } w_2 \geq \mathbf{0} \\ & \quad w_2^\top \mathbf{1} = 1 \\ & \quad w_2^\top \mu \geq w_1^\top \mu - c_1 \\ & \quad w_2^\top \Sigma w_2 \leq w_1^\top \Sigma w_1 + c_2 \\ & \quad y_1(w_1^\top x_2 + z_2) \geq 1 \\ & \quad y_2(w_2^\top x_2 + z_2) \geq 1. \end{aligned}$$

Again, let $\mathcal{I} \in \mathbb{N}$ be the total number of portfolios that we want to obtain. Although w_1 is obtained by executing the MV optimization program, we consider w_1 as the result of the ‘first iteration’, for notational purposes. For iteration $i = 2$, we obtain w_2 from the optimization program above and for iteration $i = 2, \dots, \mathcal{I}$, we search for solution w_i such that it is at maximum distance from the already obtained portfolios, w_1, \dots, w_{i-1} . We generalize the optimization program above for iteration $i = 2, \dots, \mathcal{I}$.

Definition 4.3.1. Let w_1 be the MV optimal solution corresponding to our risk aversion parameter. For iteration $i = 2, \dots, \mathcal{I}$, we have already obtained solutions w_1, \dots, w_{i-1} . Let $y_{i'} = 1$ for all $i' = 1, \dots, i-1$ and let $y_i = -1$. We define the following optimization program and call it *maximum distance optimization* (MDO):

$$\begin{aligned} & \min_{\substack{x_i, w_i \in \mathbb{R}^N \\ z_i \in \mathbb{R}}} \|x_i\| \\ & \text{subject to } w_i \geq \mathbf{0} \\ & \quad w_i^\top \mathbf{1} = 1 \\ & \quad w_i^\top \mu \geq w_1^\top \mu - c_1 \\ & \quad w_i^\top \Sigma w_i \leq w_1^\top \Sigma w_1 + c_2 \\ & \quad y_{i'}(w_{i'}^\top x_i + z_i) \geq 1, \text{ for all } i' = 1, \dots, i. \end{aligned} \tag{4.3}$$

We define solutions of MDO to be *maximum distance portfolios*. Note that since we consider w_1 to be the solution of the first iteration, w_1 is also a maximum distance portfolio. Let $\mathcal{I} \in \mathbb{N}$ and $w_1, \dots, w_{\mathcal{I}}$ be maximum distance portfolios. Also, let $\alpha_1, \dots, \alpha_{\mathcal{I}} \in \mathbb{R}_{\geq 0}$, with $\alpha_1 + \dots + \alpha_{\mathcal{I}} = 1$. Then we define the portfolio $\alpha_1 w_1 + \dots + \alpha_{\mathcal{I}} w_{\mathcal{I}}$ to be a *convex combination portfolio*.

In Section 4.4, we show that we can implement optimization program (4.3) using a heuristic method, however, does it solve our initial optimization program (4.1)? We prove two theorems. Theorem 4.3.2 shows that (4.1) and (4.3) are equivalent and in Theorem 4.3.3, we show that any convex combination portfolio lies in the permissible area.

Theorem 4.3.2. *Optimization programs, (4.1) and (4.3), are equivalent. That is, if $\mathbf{x}_i, \mathbf{w}_i, z_i$ are optimal for (4.3), then \mathbf{w}_i is optimal for (4.1) and if \mathbf{w}_i is optimal for (4.1) and $\|\mathbf{w}_i - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\| > 0$, then there exists \mathbf{x}_i and z_i such that $\mathbf{x}_i, \mathbf{w}_i, z_i$ are optimal for (4.3).*

Proof. Let λ_k be our risk aversion parameter and let $\mathbf{w}_1, \dots, \mathbf{w}_{i-1}$ be the already obtained portfolios. Suppose that for iteration i , \mathbf{w}_{CH} is optimal for (4.1), with $\|\mathbf{w}_{\text{CH}} - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\| > 0$. We show that \mathbf{w}_{CH} is also optimal for (4.3). Suppose that $\mathbf{w}_{\text{MDO}}, \mathbf{x}_{\text{MDO}}, z_{\text{MDO}}$ form a (not necessarily optimal) solution to (4.3), that is, they satisfy the constraints of (4.3). Let \mathbf{H}_{MDO} be the corresponding hyperplane: $\mathbf{H}_{\text{MDO}} = \{\mathbf{w} : \mathbf{w}^\top \mathbf{x}_{\text{MDO}} + z_{\text{MDO}} = 1\}$, and $\mathbf{H}_{\text{MDO}}^-$ the corresponding halfspace: $\mathbf{H}_{\text{MDO}}^- = \{\mathbf{w} : \mathbf{w}^\top \mathbf{x}_{\text{MDO}} + z_{\text{MDO}} \geq 1\}$. Note that $2/\|\mathbf{x}_{\text{MDO}}\| = \|\mathbf{w}_{\text{MDO}} - \mathbf{H}_{\text{MDO}}\|$ by Lemma 4.2.1. Suppose that

$$\|\mathbf{w}_{\text{MDO}} - \mathbf{H}_{\text{MDO}}\| > \|\mathbf{w}_{\text{CH}} - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\|.$$

Then,

$$\|\mathbf{w}_{\text{MDO}} - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\| \geq \|\mathbf{w}_{\text{MDO}} - \mathbf{H}_{\text{MDO}}\| > \|\mathbf{w}_{\text{CH}} - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\|$$

since $\text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1}) \subseteq \mathbf{H}_{\text{MDO}}^-$. But this contradicts with \mathbf{w}_{CH} being optimal for (4.1). Hence, $\|\mathbf{w}_{\text{MDO}} - \mathbf{H}_{\text{MDO}}\| \leq \|\mathbf{w}_{\text{CH}} - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\|$ and therefore any optimal solution for (4.1) is an optimal solution to (4.3).

For the second part, we give a graphical representation in Figure 4.4, to help understand the arguments. Suppose that for iteration i , $\mathbf{w}_{\text{MDO}}, \mathbf{x}_{\text{MDO}}, z_{\text{MDO}}$ form an optimal solution to (4.3), with hyperplane $\mathbf{H}_{\text{MDO}} = \{\mathbf{w} : \mathbf{w}^\top \mathbf{x}_{\text{MDO}} + z_{\text{MDO}} = 1\}$ and halfspace $\mathbf{H}_{\text{MDO}}^- = \{\mathbf{w} : \mathbf{w}^\top \mathbf{x}_{\text{MDO}} + z_{\text{MDO}} \geq 1\}$. We show that \mathbf{w}_{MDO} is also optimal for (4.1). Let \mathbf{w}_{CH} a (not necessarily optimal) solution to (4.1). Suppose that

$$\|\mathbf{w}_{\text{CH}} - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\| > \|\mathbf{w}_{\text{MDO}} - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\|.$$

Let $\mathbf{w}'_{\text{CH}} \in \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})$ be such that $\|\mathbf{w}_{\text{CH}} - \mathbf{w}'_{\text{CH}}\| = \|\mathbf{w}_{\text{CH}} - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\|$. Define a hyperplane \mathbf{H}_{CH} such that $\mathbf{w}_{\text{CH}} - \mathbf{w}'_{\text{CH}} \perp \mathbf{H}_{\text{CH}}$ and $\mathbf{w}'_{\text{CH}} \in \mathbf{H}_{\text{CH}}$. We prove that the corresponding halfspace \mathbf{H}_{CH}^- satisfies $\text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1}) \subseteq \mathbf{H}_{\text{CH}}^-$. Suppose that there is a point $\mathbf{v} \in \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})$, such that $\mathbf{v} \notin \mathbf{H}_{\text{CH}}^-$. We have $\alpha \mathbf{v} + (1 - \alpha) \mathbf{w}'_{\text{CH}} \in \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})$, for all $\alpha \in [0, 1]$, so

$$\|\mathbf{w}_{\text{CH}} - (\alpha \mathbf{v} + (1 - \alpha) \mathbf{w}'_{\text{CH}})\| \geq \|\mathbf{w}_{\text{CH}} - \mathbf{w}'_{\text{CH}}\|.$$

Rewrite this for $\alpha > 0$ as

$$2\langle \mathbf{w}_{\text{CH}}, \mathbf{w}'_{\text{CH}} \rangle - 2\langle \mathbf{w}_{\text{CH}}, \mathbf{v} \rangle + \alpha \langle \mathbf{v}, \mathbf{v} \rangle + 2(1 - \alpha) \langle \mathbf{v}, \mathbf{w}'_{\text{CH}} \rangle + (\alpha - 2) \langle \mathbf{w}'_{\text{CH}}, \mathbf{w}'_{\text{CH}} \rangle \geq 0.$$

Let $\alpha \rightarrow 0$, then we obtain:

$$\langle \mathbf{w}_{\text{CH}} - \mathbf{w}'_{\text{CH}}, \mathbf{w}'_{\text{CH}} - \mathbf{v} \rangle \geq 0.$$

On the other hand, since $\mathbf{w}_{\text{CH}} - \mathbf{w}'_{\text{CH}} \perp \mathbf{H}_{\text{CH}}$ and $\mathbf{v} \notin \mathbf{H}_{\text{CH}}^-$, we have

$$\langle \mathbf{w}_{\text{CH}} - \mathbf{w}'_{\text{CH}}, \mathbf{v} - \mathbf{w}'_{\text{CH}} \rangle > 0.$$

This gives a contradiction, hence such a \mathbf{v} cannot exist. We find a hyperplane such that $\mathbf{w}_{\text{CH}} - \mathbf{w}'_{\text{CH}} \perp \mathbf{H}_{\text{CH}}$, $\mathbf{w}'_{\text{CH}} \in \mathbf{H}_{\text{CH}}$, and $\text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1}) \subseteq \mathbf{H}_{\text{CH}}^-$. We get

$$\|\mathbf{w}_{\text{CH}} - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\| = \|\mathbf{w}_{\text{CH}} - \mathbf{w}'_{\text{CH}}\| = \|\mathbf{w}_{\text{CH}} - \mathbf{H}_{\text{CH}}\|.$$

4.4.2 Number of iterations

Let J denote the number of iterations of the BH algorithm. The greater the value of J , the higher the probability of finding a global minimum. However, large J means more computation time. In Section 4.5, we work out an example and show the influence of the number of BH iterations on the computation time.

Note that if we obtain a local optimal solution instead of a global optimum, the solution still contributes to our search. We want to explore the region below the efficient frontier and we can explore it as thoroughly as we want, since we can choose \mathcal{I} as large as we want. Hence, if we obtain a large set of local optima, we can still thoroughly explore the region below the efficient frontier.

4.4.3 Step size

The step size determines the size of the jumps. The step size should be comparable to the difference between local minima of the function that is being optimized [4]. BH adjusts the step size if necessary. A step size that is too small means that we may never jump beyond a barrier. A step size that is too big means that we may constantly jump over a global minimum. We use a step size of 1. The maximum distance between two portfolios is $\sqrt{2}$ and we argue that we rather make jumps that are too big than jumps that are too small.

4.4.4 Temperature

The probability that a worse temporary minimum is accepted, is determined by the objective function and by a parameter called the temperature, τ . The probability of acceptance is:

$$e^{-(\text{Objective Function}(\text{new minimum}) - \text{Objective Function}(\text{old temporary minimum}))/\tau}.$$

If the temperature is too big compared to the typical difference in objective function values between local minima, we will almost always accept a worse minimum, perhaps at the expense of not converging to a global minimum. If the temperature is too small, we may never accept a worse minimum and therefore never reach beyond a barrier. Hence, the temperature should be comparable to the typical difference of the objective function values between local minima.

We argue that in our case an appropriate value for τ depends on the number of asset classes, N . Let $n = 1, \dots, N$ and let $\mathbf{w}_i, \mathbf{x}_i, z_i$ form a local minimum. Let $w_{n,i}$ denote the weight of asset class n of portfolio \mathbf{w}_i . We get $w_{n,i} = \mathcal{O}(1/N)$. Let $\mathbf{w}_i^{\text{CH}} \in \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})$ such that $\|\mathbf{w}_i - \mathbf{w}_i^{\text{CH}}\| = \|\mathbf{w}_i - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\|$. Using Theorem 4.3.2, we get

$$2\|\mathbf{x}_i\| = \|\mathbf{w}_i - \mathbf{w}_i^{\text{CH}}\| = \sqrt{\sum_{n=1}^N (w_{n,i} - w_{n,i}^{\text{CH}})^2} = \sqrt{\sum_{n=1}^N \mathcal{O}(1/N)^2} = \mathcal{O}(1/\sqrt{N}).$$

So $\|\mathbf{x}_i\| = \mathcal{O}(1/\sqrt{N})$. Let $\mathbf{w}'_i, \mathbf{x}'_i, z'_i$ also form a local minimum. A typical difference of the objective function values between these local minima is:

$$\|\mathbf{x}_i\| - \|\mathbf{x}'_i\| = \mathcal{O}(1/\sqrt{N}).$$

Hence, for N not too big, we can choose $\tau = \sqrt{N}$. However, for large N , it is better to rescale our optimization problem as follows:

$$\begin{aligned} & \min_{\mathbf{x}_i, \mathbf{w}_i, z_i} \|\mathbf{x}_i\| \\ & \text{subject to } \mathbf{w}_i \geq \mathbf{0} \\ & \mathbf{w}_i^\top \mathbf{1} = N \\ & \mathbf{w}_i^\top \boldsymbol{\mu} \geq \mathbf{w}_1^\top \boldsymbol{\mu} - c_1 \\ & \mathbf{w}_i^\top \boldsymbol{\Sigma} \mathbf{w}_i \leq \mathbf{w}_1^\top \boldsymbol{\Sigma} \mathbf{w}_1 + c_2. \\ & y_{i'}(\mathbf{w}_i^\top \mathbf{x}_i + z_i) \geq 1, \text{ for all } i' = 1, \dots, i. \end{aligned}$$

We get $w_{n,i} = \mathcal{O}(1)$ and

$$2/\|\mathbf{x}_i\| = \sqrt{\sum_{n=1}^N (w_{n,i} - w_{n,i}^{\text{CH}})^2} = \sqrt{\sum_{n=1}^N \mathcal{O}(1)^2} = \mathcal{O}(\sqrt{N}).$$

So a typical difference of the objective function values between two local minima, \mathbf{w}_i and \mathbf{w}'_i , is:

$$\|\mathbf{x}_i\| - \|\mathbf{x}'_i\| = \mathcal{O}(1/\sqrt{N}).$$

We can choose $\tau = 1/\sqrt{N}$.

4.4.5 Local minimization algorithm

We use a local minimization algorithm called Sequential Least Squares Programming. We do not go in to too much detail about how this algorithm works. We only note that it has a stopping criterion of a maximum of 100 iterations and it tries to find an optimal function value up to a precision of 10^{-6} .

4.4.6 Basin hopping pseudocode

We give the basin hopping algorithm in pseudocode.

- (1) $j = 0$
- (2) $X_j = \text{Starting point}$
- (3) $Y_{\text{temp opt}} = \text{Find local minimum around } X_j$
- (4) **while** $j < J$ **do**
- (5) $X_{j+1} = \text{Jump from } Y_{\text{temp opt}} \text{ to another point}$
- (6) $Y_{j+1} = \text{Find local minimum around } X_{j+1}$
- (7) **if** $\text{Objective Function}(Y_{j+1}) < \text{Objective Function}(Y_{\text{temp opt}})$ **then**
 $Y_{\text{temp opt}} = Y_{j+1}$
 $Y_{\text{global opt}} = Y_{j+1}$
- (8) **if** $\text{Objective Function}(Y_{j+1}) \not\prec \text{Objective Function}(Y_{\text{temp opt}})$ **then**
with probability $e^{-(\text{Objective Function}(Y_{j+1}) - \text{Objective Function}(Y_{\text{temp opt}}))/\tau}$
 $Y_{\text{temp opt}} = Y_{j+1}$
- (9) $j = j + 1$
- (10) **return** $Y_{\text{global opt}}$

4.5 Example

We use the example from Section 2.2 to show the results of MDO. We use also the same risk aversion parameters of Section 2.2: $\lambda_k = ((k-1)/25 + 1)^{-9}$, $k = 1, \dots, 25$. Table 4.1 gives an overview of asset classes and input parameters. Before we discuss the results, we like to build in an extra stop criterion. In the next section, we explain why.

TABLE 4.1: Overview of input parameters, μ and Σ .

Asset classes	Covariance matrix								Means
Equity									
1. Canadian	30.25	15.85	10.26	9.68	19.17	16.79	2.87	2.83	0.39
2. French	15.85	49.42	27.11	20.79	22.82	13.30	3.11	2.85	0.88
3. German	10.26	27.11	38.69	15.33	17.94	9.10	3.38	2.72	0.53
4. Japanese	9.68	20.79	15.33	49.56	16.92	6.66	1.98	1.76	0.88
5. U.K.	19.17	22.82	17.94	16.92	36.12	14.47	3.02	2.72	0.79
6. U.S.	16.79	13.30	9.10	6.66	14.47	18.49	3.11	2.82	0.71
Bonds									
7. U.S.	2.87	3.11	3.38	1.98	3.02	3.11	4.04	2.88	0.25
8. European	2.83	2.85	2.72	1.76	2.72	2.82	2.88	2.43	0.27

4.5.1 Extra stop criterion

We can explore the region below the efficient frontier as thoroughly as we like, but we like to make sure that any obtained portfolio actually contributes to our search. Note that for $i = 2, \dots, \mathcal{I}$, $\text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})$ defines a convex polytope. When we add vector \mathbf{w}_i , the polytope expands, but if \mathbf{w}_i is too close to $\text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})$, it hardly contributes.

We implement an extra stop criterion. Let \mathcal{I} be the number of portfolios we like to obtain. Then, for each iteration $i = 2, \dots, \mathcal{I}$, we calculate the distance towards the convex hull of the already obtained portfolios. Use Theorem 4.3.2 to define

$$d_i^{\text{CH}} = \|\mathbf{w}_i - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\| = 2/\|\mathbf{x}_i\|.$$

We expect that the distance of \mathbf{w}_2 towards \mathbf{w}_1 is big and with each next iteration the distance between \mathbf{w}_i and $\text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})$ becomes smaller (provided that our heuristic method works properly). If we obtain a solution of which the distance is close to zero, we like the optimization program to stop and not bluntly continue up to iteration \mathcal{I} .

We decide that we want the optimization program to stop if the distance to the convex hull of the already obtained portfolios becomes less than 0.1. If we scale this to the maximum distance of $\sqrt{2}$, this limit corresponds to a distance of approximately 7.1%. A simple three dimensional example gives an indication of what this 7.1% limit means. Suppose we have obtained portfolios, $\mathbf{w}_1 = [1, 0, 0]^\top$ and $\mathbf{w}_2 = [0, 1, 0]^\top$. And suppose that in the third iteration we obtain $\mathbf{w}_3 = [0.45, 0.45, 0.0707]^\top$, approximately. We have $d_3^{\text{CH}} = 7.1\%$. Now, by taking convex combination $1/2 \cdot \mathbf{w}_1 + 1/2 \cdot \mathbf{w}_2$, the portfolio $[0.5, 0.5, 0]^\top$ is already included in $\text{CH}(\mathbf{w}_1, \mathbf{w}_2)$ and the difference between this portfolio and \mathbf{w}_3 is small. Hence, we interrupt our optimization program as soon as \mathbf{w}_i is at closer distance than 0.1 and in that case, we do not add \mathbf{w}_i to the set of maximum distance portfolios.

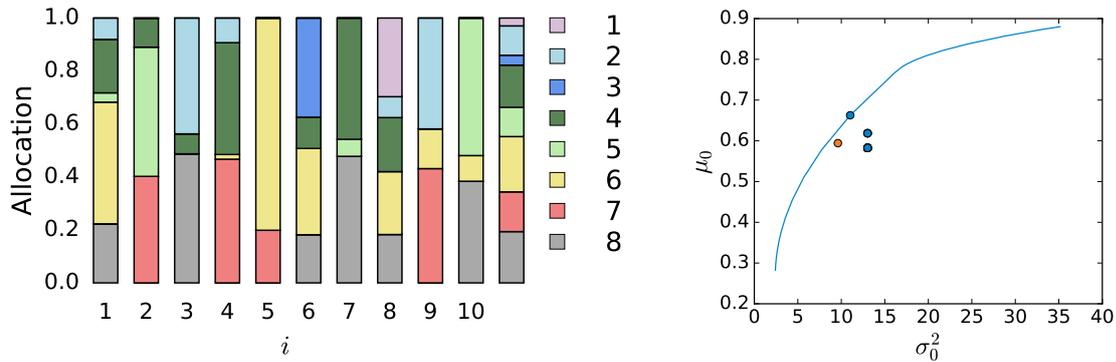
Let $I \leq \mathcal{I}$ denote the maximum number of portfolios that we find.

4.5.2 Results

We show the results of MDO for three different risk aversion levels. First, we discuss the results for medium risk aversion, then, we discuss the results at a high risk aversion level, and lastly, we discuss the results for a low risk aversion level.

For the medium risk aversion case, take $\mathcal{I} = 10$ and w_1 corresponding to λ_{14} . Let $c_1 = 0.08$, $c_2 = 2$, and $J = 200$. We find $I = 10$ portfolios. Figure 4.6 shows the results in terms of allocations in the figure on the left and the results in terms of risk and return levels in the figure on the right. The bar on the left represents the starting solution w_1 , corresponding to the blue dot on the efficient frontier. The bars next to w_1 represent solutions, w_2, \dots, w_{10} , respectively. These allocations correspond to the other blue dots in the figure on the right. The bar on the far right side represents a convex combination portfolio that is an average of the obtained portfolios: $\frac{1}{J}w_1 + \dots + \frac{1}{J}w_I$. This convex combination portfolio is represented by the orange dot in the figure on the right.

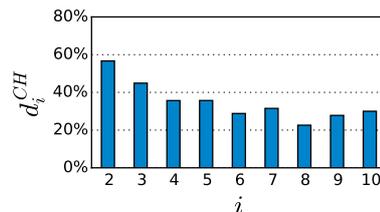
FIGURE 4.6: Allocations and risk and return levels of maximum distance portfolios (blue dots) plus a convex combination portfolio (orange dot), for λ_{14} , $J = 200$, $c_1 = 0.08$, and $c_2 = 2$.



We see that the solutions indeed all differ a lot from one another. We also see that the convex combination portfolio also lies in the permissible area. We discuss the possibilities of the convex combination portfolio more extensively in the section hereafter; in this section we focus on the maximum distance portfolios.

To get an indication of how good the BH algorithm works, we show d_i^{CH} for $i = 2, \dots, I$. Figure 4.7 shows the results. We scale these distances towards the maximum distance of $\sqrt{2}$, corresponding to 100%.

FIGURE 4.7: Distance to convex hull of already obtained portfolios, for λ_{14} , $J = 200$, $c_1 = 0.08$, and $c_2 = 2$.



We see that the distances do not necessarily decrease as i increases. This means that we sometimes find a local optimal portfolio instead of a global optimal portfolio. Let us see what happens if we increase the number of BH iterations: take $J = 1000$. Figure 4.8 and Figure 4.9 show the results.

FIGURE 4.8: Allocations and risk and return levels of maximum distance portfolios (blue dots) plus a convex combination portfolio (orange dot), for λ_{14} , $J = 1000$, $c_1 = 0.08$, and $c_2 = 2$.

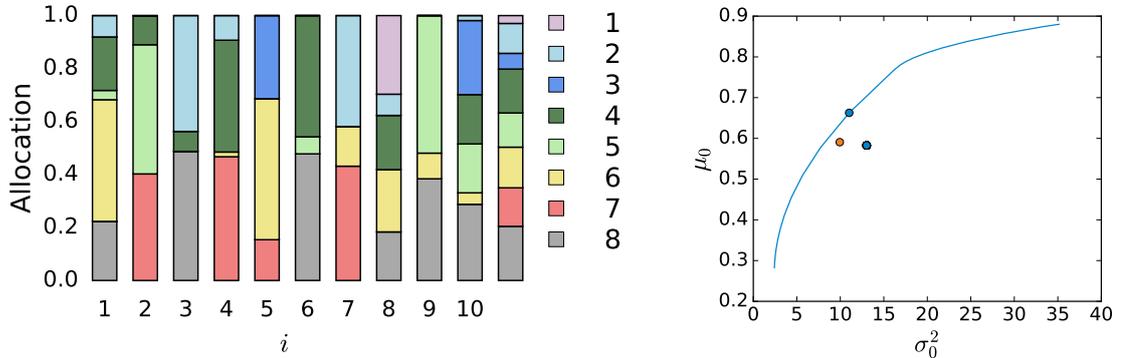
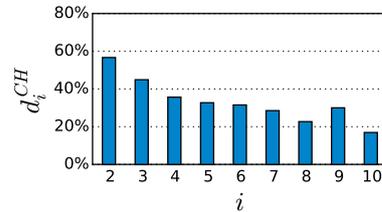


FIGURE 4.9: Distance to convex hull of already obtained portfolios, for λ_{14} , $J = 1000$, $c_1 = 0.08$, and $c_2 = 2$.



We see that d_i^{CH} decreases as i increases, except for $i = 9$. So, for such a large value of J , we still can obtain local optimal portfolios. Furthermore, compare the allocations of the solutions of $J = 1000$ to the allocations of the solutions of $J = 200$ (Figure 4.8 to Figure 4.6). We see that the first four portfolios are equal and that the sixth, seventh, eighth, and ninth portfolio corresponding to $J = 1000$ are also obtained for $J = 200$, only in a different order. Portfolios, five and ten, do not occur for $J = 200$. We calculate $d_5^{CH} = 32\%$ for $J = 1000$ and $d_5^{CH} = 29\%$ for $J = 200$. This means that w_5 corresponding to $J = 1000$, is a better solution than w_5 corresponding to $J = 200$. Note that we need that the solutions, w_1, \dots, w_4 , are equal for both $J = 1000$ and $J = 200$. This way, we calculate the distance of w_5 towards the same convex hull. We can, for example, not judge w_{10} the same way, because we have another convex hull of solutions, w_1, \dots, w_9 , for $J = 1000$ than we do for $J = 200$. Summarizing the observations, we see that a larger value of J does improve the results but since we use a heuristic method, we still do not have any certainty on finding global optimal solutions.

Note that in the conclusions above, when we say, for example, that the second portfolio w_2 corresponding to $J = 200$ (Figure 4.6) is the same as the second portfolio w_2 corresponding to $J = 1000$ (Figure 4.8), this is loosely spoken. There might be tiny differences in the allocations that we can not see in the figures.

As we discuss above, the higher the value of J , the higher the probability that we find global optimal solutions. Unfortunately, it does come at a cost: the computation time increases considerably. For the case of $J = 200$, we have a computation time of approximately 40 seconds. For $J = 1000$, we have a computation time of approximately three minutes. This is a significant increase.

In our analysis, especially when we analyze the robustness, we have to run the optimization program numerous times, so we like to have a low computation time. We argue that

the order in which we find the maximum distance portfolios does not matter. Furthermore, as we explain above, finding local optimal portfolios is also acceptable, because we can choose \mathcal{I} as big as we like and the stopping criterion as (non) strict as we like. The value of \mathcal{I} , in combination with the stopping criterion, has also an influence on the computation time. However, if we choose for example that we want to obtain $\mathcal{I} = 30$ portfolios, stopping criterion of still 7.1%, and $J = 200$, we have a computation time of two minutes. Recall that we have three minutes of computation time for $\mathcal{I} = 10$, $J = 1000$. A reduction of one-third of the computation time is significant. For that reason, we stick with $J = 200$ as the number of iterations of the BH algorithm.

A last disadvantage that we like to point out is that for lower J , the probability of finding a local optimal portfolio that exceeds the 7.1% stopping criterion while there are still other local or global solutions that do not exceed this limit, increases. In such a situation, we stop our search too early. This is a consequence we need to accept.

Let us go back to the analysis of MDO at a medium risk aversion level and let us narrow the permissible area. Take $c_1 = 0.04$ and $c_2 = 1$. We expect that the maximum distance portfolios do not differ much from the starting portfolio, as the restrictions on the risk and return level are more severe. Figure 4.10 and Figure 4.11 show the results.

FIGURE 4.10: Allocations and risk and return levels of maximum distance portfolios (blue dots) plus a convex combination portfolio (orange dot), for λ_{14} , $J = 200$, $c_1 = 0.04$, and $c_2 = 1$.

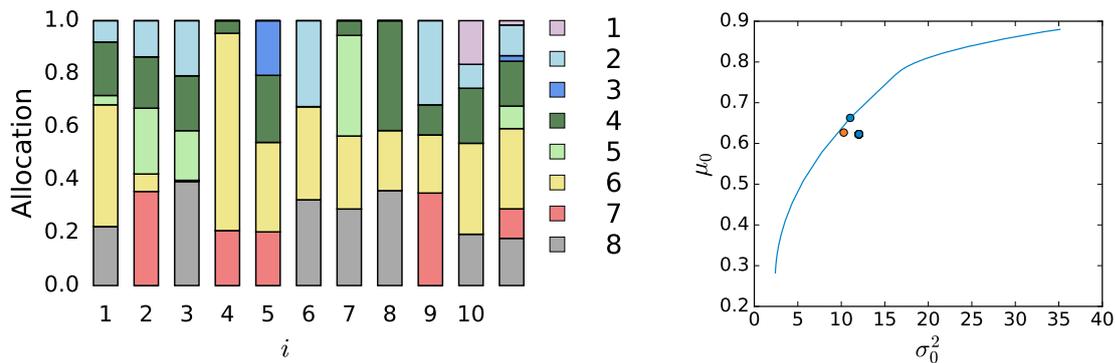
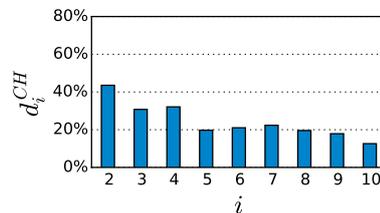


FIGURE 4.11: Distance to convex hull of already obtained portfolios, for λ_{14} , $J = 200$, $c_1 = 0.04$, and $c_2 = 1$.



Compare this to the case that $c_1 = 0.08$, $c_2 = 2$, and $J = 200$ (Figure 4.10 to Figure 4.6). It seems that in terms of the allocations, portfolios, w_1, \dots, w_{10} , are more alike for $c_1 = 0.04$ and $c_2 = 1$. We can confirm this by comparing Figure 4.11 to Figure 4.7. We see that the distances towards the convex hull of the already obtained portfolios are lower for $c_1 = 0.04$ and $c_2 = 1$ than they are for $c_1 = 0.08$, $c_2 = 2$. Hence, a restriction of the permissible area can result in a smaller convex hull of maximum distance portfolios.

Let us see what happens when we broaden the permissible area. Take $c_1 = 0.16$ and $c_2 = 4$. In accordance with the observations above, we expect to be able to obtain portfolios

that are at larger Euclidean distance from one another than in the case of $c_1 = 0.08$ and $c_2 = 2$. Figure 4.12 and Figure 4.13 show the results.

FIGURE 4.12: Allocations and risk and return levels of maximum distance portfolios (blue dots) plus a convex combination portfolio (orange dot), for λ_{14} , $J = 200$, $c_1 = 0.16$, and $c_2 = 4$.

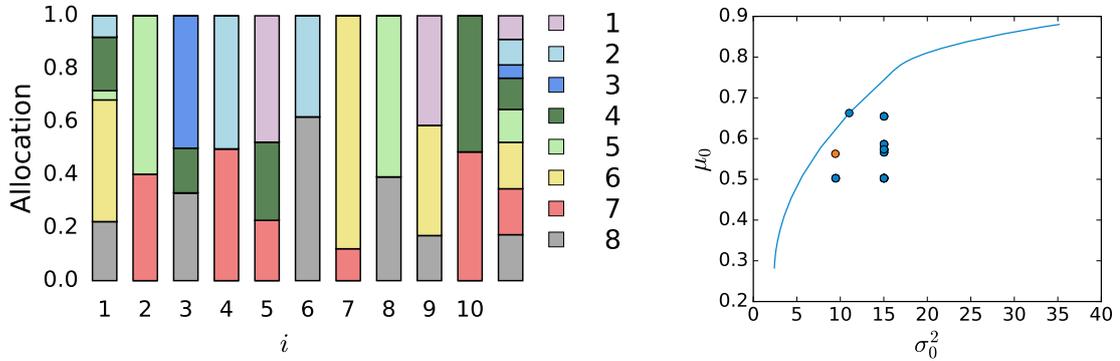
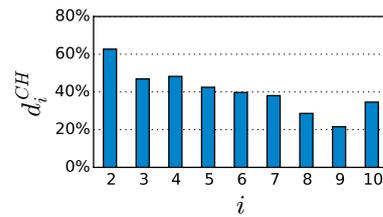


FIGURE 4.13: Distance to convex hull of already obtained portfolios, for λ_{14} , $J = 200$, $c_1 = 0.16$, and $c_2 = 4$.



These portfolios indeed differ more from one another than in the case that $c_1 = 0.08$ and $c_2 = 2$. We can see this when we look at the allocations by comparing Figure 4.12 to Figure 4.6 and we can see this when we look at the distances towards the convex hull of the already obtained portfolios by comparing Figure 4.13 to Figure 4.7. All the distances, except for $i = 9$, are higher for $c_1 = 0.16$ and $c_2 = 4$.

We now know what can happen if we restrict or broaden the permissible area. Let us continue our investigation and see what happens at a high risk aversion rate: take λ_{11} , $c_1 = 0.08$, and $c_2 = 2$. Figure 4.14 and Figure 4.15 show the results.

FIGURE 4.14: Allocations and risk and return levels of maximum distance portfolios (blue dots) plus a convex combination portfolio (orange dot), for λ_{11} , $J = 200$, $c_1 = 0.08$, and $c_2 = 2$.

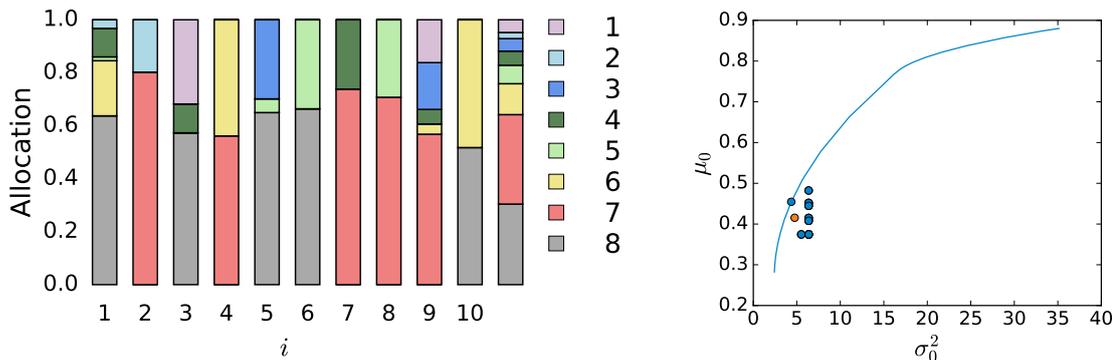
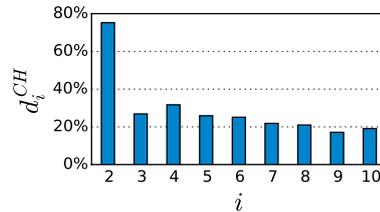


FIGURE 4.15: Distance to convex hull of already obtained portfolios, for λ_{11} , $J = 200$, $c_1 = 0.08$, and $c_2 = 2$.



We see that for each maximum distance portfolio w_i , $i = 1, \dots, 10$, either the seventh or the eighth asset class takes a large share. This is because these are the only low risk asset classes. Also, for this reason, d_2^{CH} is big and the distances of the other iterations are small.

Lastly, let us look at a low risk aversion level: take λ_{19} , $c_1 = 0.08$, and $c_2 = 2$. Figure 4.16 and Figure 4.17 show the results.

FIGURE 4.16: Allocations and risk and return levels of maximum distance portfolios (blue dots) plus a convex combination portfolio (orange dot), for λ_{19} , $J = 200$, $c_1 = 0.08$, and $c_2 = 2$.

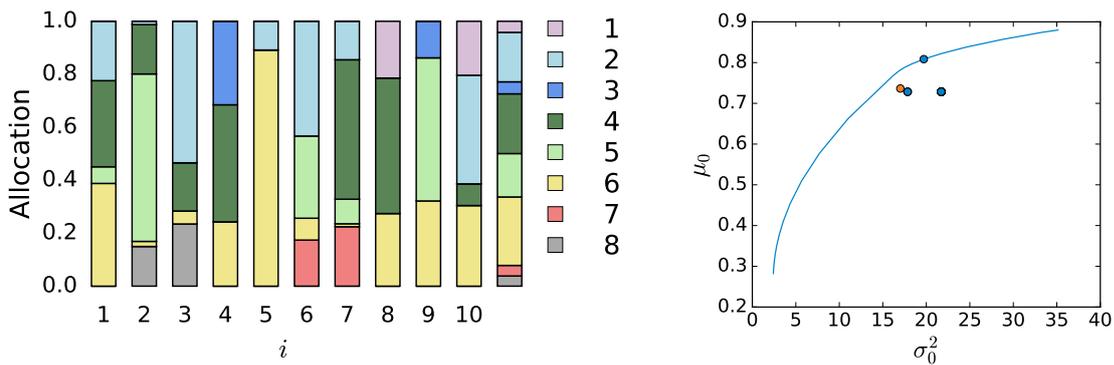
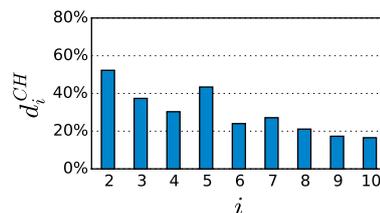


FIGURE 4.17: Distance to convex hull of already obtained portfolios, for λ_{19} , $J = 200$, $c_1 = 0.08$, and $c_2 = 2$.



Here, we seem to have less extreme allocations than in the high risk aversion case (compare Figure 4.16 to Figure 4.14). The distances towards the convex hull of the already obtained portfolios (Figure 4.17) are difficult to compare to those of the high risk aversion case (Figure 4.15), so we can not make any statements about those.

In our analyses above, we look at different risk aversion parameters and different sizes of the permissible area. In all these cases there is one thing that stands out: the maximum distances portfolios all end up on the border of the permissible area. This seems logical, since one can intuitively reason that being able to deviate more in terms of risk and return, means more freedom in choosing asset classes. In every case we investigate we do not

encounter a maximum distance portfolio that is not on the border of the permissible area, but nevertheless, our intuition misleads us, as we show in the next theorem.

Theorem 4.5.1. *MDO has a solution that is on the border of the set determined by the constraints.*

Proof. Consider the Bauer Maximum Principle (Theorem 7.69 from [1]): ‘If C is a compact convex subset of a locally convex Hausdorff space, then every upper semicontinuous convex function on C has a maximizer that is an extreme point.’ We use this theorem to show that we do not necessarily end up on the border of the permissible area.

Use Theorem 4.3.2 to see that we can use the formulation of optimization program (4.1). Let $i = 2, \dots, I$. Note that \mathbb{R}^N is a convex Hausdorff space and from Heine–Borel we know that a subset of \mathbb{R}^N is compact if and only if it is closed and bounded. We have that $\{\mathbf{w}_i : \mathbf{w}_i \geq \mathbf{0}\}$ and $\{\mathbf{w}_i : \mathbf{w}_i^\top \boldsymbol{\mu} \geq \mathbf{w}_1^\top \boldsymbol{\mu} - c_1\}$ are closed half spaces and $\{\mathbf{w}_i : \mathbf{w}_i^\top \mathbf{1} = 1\}$ is a closed hyperplane, hence these sets are closed and convex. We know that $\{\mathbf{w}_i : \mathbf{w}_i^\top \boldsymbol{\Sigma} \mathbf{w}_i \leq \mathbf{w}_1^\top \boldsymbol{\Sigma} \mathbf{w}_1 + c_2\}$ is closed. It is also convex, since sublevel sets of convex functions are convex. Furthermore, the intersection of closed convex sets is also closed and convex, hence the constraints of MDO form a convex set. Since we have \mathbf{w}_i such that $\mathbf{w}_i^\top \mathbf{1} = 1$ and $\mathbf{w}_i \geq \mathbf{0}$ we know that the constraints also form a bounded set. Hence, the constraints of (4.1) form a compact convex subset of a locally convex Hausdorff space.

Next, we show that the function $\mathbb{R}^N \rightarrow \mathbb{R} : \mathbf{w}_i \mapsto \|\mathbf{w}_i - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\|$, is convex and continuous on \mathbb{R}^N . Let $\mathbf{w}'_1, \mathbf{w}'_2 \in \mathbb{R}^N$ (not to be confused with maximum distance portfolios) and let $\alpha_1, \alpha_2 \in \mathbb{R}_{\geq 0}$, with $\alpha_1 + \alpha_2 = 1$. Let $\mathbf{d}_{\mathbf{w}'_1}, \mathbf{d}_{\mathbf{w}'_2} \in \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})$ be such that

$$\begin{aligned} \|\mathbf{w}'_1 - \mathbf{d}_{\mathbf{w}'_1}\| &= \|\mathbf{w}'_1 - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\| \\ \|\mathbf{w}'_2 - \mathbf{d}_{\mathbf{w}'_2}\| &= \|\mathbf{w}'_2 - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\|. \end{aligned}$$

Then, $\alpha_1 \mathbf{d}_{\mathbf{w}'_1} + \alpha_2 \mathbf{d}_{\mathbf{w}'_2} \in \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})$ and we obtain

$$\begin{aligned} \|\alpha_1 \mathbf{w}'_1 + \alpha_2 \mathbf{w}'_2 - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\| &\leq \|\alpha_1 \mathbf{w}'_1 + \alpha_2 \mathbf{w}'_2 - (\alpha_1 \mathbf{d}_{\mathbf{w}'_1} + \alpha_2 \mathbf{d}_{\mathbf{w}'_2})\| \\ &\leq \alpha_1 \|\mathbf{w}'_1 - \mathbf{d}_{\mathbf{w}'_1}\| + \alpha_2 \|\mathbf{w}'_2 - \mathbf{d}_{\mathbf{w}'_2}\|. \end{aligned}$$

Consider Theorem 7.24 from [1]: ‘In a finite dimensional vector space, every convex function is continuous on the relative interior of its domain.’ We have \mathbb{R}^N as the relative interior of itself, hence $\mathbb{R}^N \rightarrow \mathbb{R} : \mathbf{w}_i \mapsto \|\mathbf{w}_i - \text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_{i-1})\|$ is continuous on \mathbb{R}^N .

Apply the Bauer Maximum Principle to optimization program (4.1). So, there is an extreme point of the following set:

$$\{\mathbf{w}_i \in \mathbb{R}^N : \mathbf{w}_i \geq \mathbf{0}, \mathbf{w}_i^\top \mathbf{1} = 1, \mathbf{w}_i^\top \boldsymbol{\mu} \geq \mathbf{w}_1^\top \boldsymbol{\mu} - c_1, \mathbf{w}_i^\top \boldsymbol{\Sigma} \mathbf{w}_i \leq \mathbf{w}_1^\top \boldsymbol{\Sigma} \mathbf{w}_1 + c_2\}$$

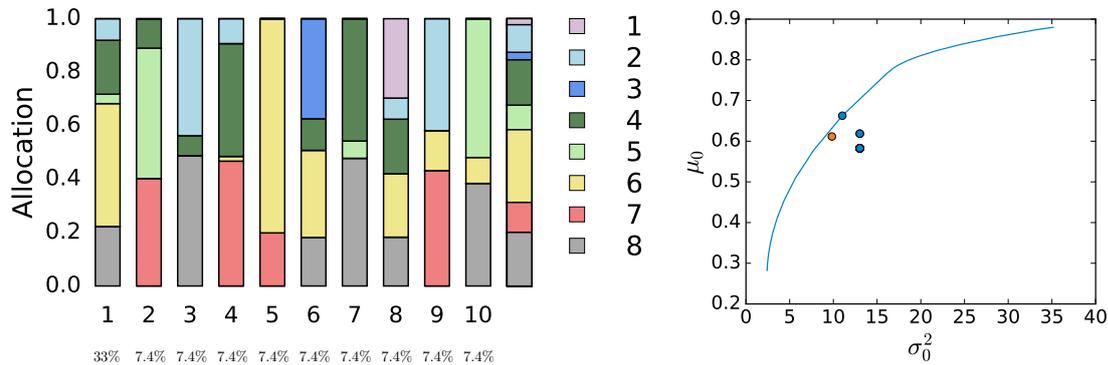
that is a solution to (4.1). □

We make the following remark. Theorem 4.5.1 implies that a solution of MDO is not necessarily on the border of the permissible area. To see this, suppose that \mathbf{w}_i is a solution of MDO with $\mathbf{w}_i \geq \mathbf{0}$ and $\mathbf{w}_i^\top \mathbf{1} = 1$. Then, the conditions on the risk and return level are not necessarily met with equality, hence \mathbf{w}_i does not necessarily lie on the border of the permissible area.

4.5.3 Convex combination portfolio

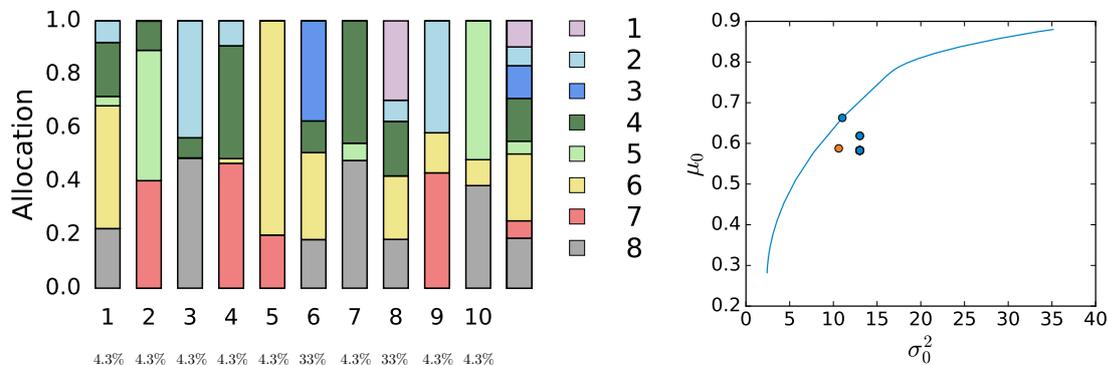
In this section, we look at the possibilities of choosing a convex combination portfolio. Take again risk aversion parameter λ_{14} , $\mathcal{I} = 10$, $J = 200$, $c_1 = 0.08$, $c_2 = 2$, and let $\alpha_1 \mathbf{w}_1, \dots, \alpha_{10} \mathbf{w}_{10}$ be the convex combination portfolio. Suppose we want a lot of the first (MV optimal) portfolio and less of the rest. We take $\alpha_1 = 33\%$ and $\alpha_2, \dots, \alpha_{10} = 7.4\%$, approximately. Figure 4.18 shows the result. Recall that the bar on the right and the orange dot in the figure on the right represent this convex combination portfolio.

FIGURE 4.18: Allocations and risk and return levels of maximum distance portfolios (blue dots) plus a convex combination portfolio (orange dot), for λ_{14} , $J = 200$, $c_1 = 0.08$, and $c_2 = 2$.



We see that the convex combination portfolio is close to the efficient frontier, so in terms of expected risk and return level this is a good portfolio. Also, we see that we obtain a diversified portfolio. However, suppose we want to do better in terms of diversification. We see that our convex combination portfolio has little weight in the first asset class and the third asset class. Looking at all the maximum distance portfolios, we see that we need to increase the coefficient of the sixth and the eighth portfolio to increase the weight of these asset classes. Take $\alpha_6, \alpha_8 = 33\%$ and $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_7, \alpha_9, \alpha_{10} = 4.3\%$, approximately. Figure 4.19 shows the results.

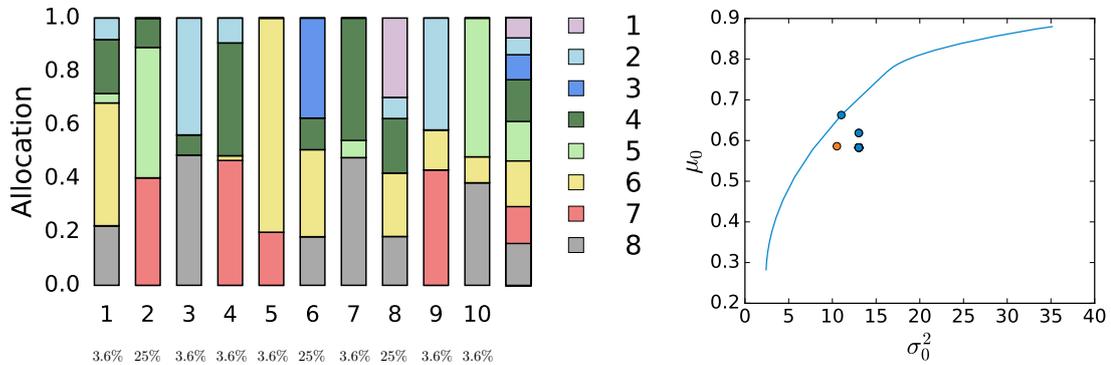
FIGURE 4.19: Allocations and risk and return levels of maximum distance portfolios (blue dots) plus a convex combination portfolio (orange dot), for λ_{14} , $J = 200$, $c_1 = 0.08$, and $c_2 = 2$.



We see that we obtain a more diversified portfolio, but our convex combination portfolio is a bit further away from the efficient frontier. Let us see what happens if we want an

even more diversified portfolio. We need to lose some weight of the sixth asset class and gain a bit of the fifth and the seventh asset class. Hence, take $\alpha_2, \alpha_6, \alpha_8 = 25\%$, $\alpha_5 = 0\%$, and $\alpha_1, \alpha_3, \alpha_4, \alpha_6, \alpha_7, \alpha_9, \alpha_{10} = 4.2\%$, approximately. Figure 4.20 shows the results.

FIGURE 4.20: Allocations and risk and return levels of maximum distance portfolios (blue dots) plus a convex combination portfolio (orange dot), for $\lambda_{14}, J = 200, c_1 = 0.08$, and $c_2 = 2$.



We see that we obtain a more diversified portfolio and that we are approximately on the same expected risk and return level as that of Figure 4.19. Hence, in this case, more diversification does not automatically lead to a less expected portfolio performance.

These are only three different convex combination portfolios, but we can pick any convex combination we like. The requirements we have determine the freedom in picking a convex combination portfolio. We can have requirements like a certain minimum expected return, a maximum expected risk, a certain level of diversification, a predetermined fixed weight for a certain asset class, and so on. We can pick a convex combination that fits our requirements the most. This is an advantage that we do not encounter in other methods.

4.5.4 The diversification effect of MDO

The diversification effect of MDO depends in the first place on the asset classes and their means and covariance matrix. It also depends on a few choices.

First, we look at the influence of the size of the permissible area. If we take c_1 and c_2 large, it is likely that we obtain maximum distance portfolios, that differ highly from one another. Because the larger the permissible area, the more asset classes are able to dominate a maximum distance portfolio and the more variation in allocations of maximum distance portfolios we get. However, the reverse is not necessarily true. If we take c_1 and c_2 , small, the remark below Theorem 4.5.1 shows that we can still obtain maximum distance portfolios that are different in terms of allocations. This depends on the features of the asset classes, over which we optimize. So, for a small permissible area, we can still obtain portfolios that have a wide variety of allocations.

Secondly, the choices for the BH algorithm are of influence. For a large diversification effect, we should choose a large value for the number of maximum distance portfolios that we would like to obtain: \mathcal{I} . Also, we should choose a small value for the extra stopping criterion. That is, we should only stop if the distance between the new portfolio and the convex hull of the already obtained portfolios is too small. Furthermore, we should choose to do a large number of BH iterations, because more iterations means a

higher probability on finding global optima, instead of local optima. The rest of the BH aspects should be chosen appropriately, as we explain in Section 4.4.

Lastly, and most importantly, is the choice of the convex combination portfolio: $\alpha_1 w_1, \dots, \alpha_I w_I$. For example, if we choose $\alpha_i = 1$, for some $i = 1, \dots, I$, it is likely that we do not have a diversified portfolio. Choosing the convex combination portfolio determines not only the diversification effect, but also the expected risk of the portfolio, the expected return of the portfolio, and possibly the change in asset allocation with respect to our previous portfolio. MDO provides us with a lot of freedom in choosing any position in possible tradeoffs between these aspects.

4.5.4.1 Conclusion

The diversification effect of MDO depends on various aspects. Apart from the asset classes under consideration, we should make suitable choices about the permissible area and for the BH algorithm. The diversification effect mainly depends on the choice of the convex combination portfolio. It is possible that we need to give in on the expected portfolio risk or return if we want a more diversified portfolio, or vice versa. However, this is not always the case.

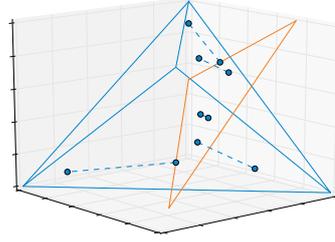
4.5.5 Norms for determining the robustness of MDO

We want to investigate the influence of changing the input parameters μ and Σ on the allocations of the maximum distance portfolios. That is, we want to know how much the convex hull of the maximum distance portfolios changes if we change the input parameters. We note that for $I > N$, $\text{CH}(w_1, \dots, w_I)$ is a N -dimensional polyhedron. If $I \leq N$, $\text{CH}(w_1, \dots, w_I)$ is a $(I - 1)$ -dimensional polyhedron in a N -dimensional space.

We call the convex hull of maximum distance portfolios w_1, \dots, w_I based on the original input parameters μ and Σ as the *original convex hull*. We call the convex hull of maximum distance portfolios corresponding to a variation $m = 1, \dots, M$ of input parameters, the *convex hull of a variation*.

For the moment, suppose that $N = 3$, $\mathcal{I} = 4$, and that we obtain $I = 4$ solutions. We get the original convex hull, $\text{CH}(w_1, \dots, w_4)$, as a three dimensional polygon in \mathbb{R}^3 . Furthermore, suppose we execute the BH algorithm again, using different input parameters: $\mu^{\text{inp}1}$ and $\Sigma^{\text{inp}1}$. We obtain maximum distance portfolios $w_1^{\text{inp}1}, \dots, w_4^{\text{inp}1}$, but suppose that the fourth solution is too close to $\text{CH}(w_1^{\text{inp}1}, \dots, w_3^{\text{inp}1})$, so we do not include it. We have that the convex hull of the variation, $\text{CH}(w_1^{\text{inp}1}, \dots, w_3^{\text{inp}1})$, is a two dimensional polyhedron in \mathbb{R}^3 . We want to be able to analyze the robustness, hence we want to know how much this convex hull differs from the original convex hull. Suppose we initially have a convex combination portfolio based on the original input parameters. That is, we have a portfolio that is a point in $\text{CH}(w_1, \dots, w_4)$. We want to know if our convex combination portfolio is still in the convex hull of the variation. Therefore, we need to calculate the distance from our initial point to $\text{CH}(w_1^{\text{inp}1}, \dots, w_3^{\text{inp}1})$. If this distance is zero, we do not need to change our portfolio. If the distance is greater than zero, we do need to change our portfolio and the distance tells us how much we need to change for the closest possible portfolio. So, if we want to investigate the robustness, we take $H \in \mathbb{N}$ random points in $\text{CH}(w_1, \dots, w_4)$ and calculate their distance towards $\text{CH}(w_1^{\text{inp}1}, \dots, w_3^{\text{inp}1})$. Figure 4.21 shows this graphically for $N = 3$ and $H = 5$. The original convex hull is blue, the convex hull corresponding to the variation of input parameters is orange.

FIGURE 4.21: Example of calculating the distance between from the original convex hull (blue) towards the convex hull of a variation (orange).



We see that there is one point for which we need only a small change our in portfolio. For the other points the distance towards the new convex hull is bigger.

Note that we have a fixed number of portfolios that we like to obtain, \mathcal{I} , but for each variation we can have a different number of portfolios that we actually obtain. We have that I represents the number of portfolios that we obtain based on the original input parameters. We denote the number of portfolios that we obtain based on variation m of input parameters by I_m .

We vary the input parameters a total of M times. Let $m = 1, \dots, M$ and $h = 1, \dots, H$. We denote the distance of point h in $\text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_I)$ towards $\text{CH}(\mathbf{w}_1^{\text{inp}m}, \dots, \mathbf{w}_{I_m}^{\text{inp}m})$ by $d_h^{\text{inp}m}$. We define three norms to judge the robustness of MDO. First, we calculate the average distance of all points $h = 1, \dots, H$:

$$\bar{d}^{\text{inp}m} = \frac{1}{H} \sum_{h=1}^H d_h^{\text{inp}m}.$$

We interpret $\bar{d}^{\text{inp}m}$ as the average change we need to make in an initial portfolio to end up in the convex hull of a variation m . We average over all variations, define:

$$\bar{d}^{\text{MDO}} = \frac{1}{M} \sum_{m=1}^M \bar{d}^{\text{inp}m}.$$

We interpret \bar{d}^{MDO} as the overall average change we need to make in an initial portfolio to end up in the convex hull of a variation.

Secondly, we calculate the minimum distance:

$$d_{\min}^{\text{inp}m} = \min_{h=1, \dots, H} d_h^{\text{inp}m}.$$

We interpret $d_{\min}^{\text{inp}m}$ as most favorable case we can have for variation m . The portfolio corresponding to $d_{\min}^{\text{inp}m}$ is the portfolio where we have to change the least to end up in the convex hull of a variation m . We average the value of this minimum distance over all variations, define:

$$d_{\min}^{\text{MDO}} = \frac{1}{M} \sum_{m=1}^M d_{\min}^{\text{inp}m}.$$

So, d_{\min}^{MDO} represents the overall lower bound of the change we need to make.

Lastly, we calculate the maximum distance:

$$d_{\max}^{\text{inp}m} = \max_{h=1, \dots, H} d_h^{\text{inp}m}$$

We interpret $d_{\max}^{\text{inp}m}$ as the most unfavorable case we can have. The portfolio corresponding to $d_{\max}^{\text{inp}m}$ is the portfolio for which we need to change the most to end up in the convex hull of a variation m . We average this maximum distance over all variations, define:

$$d_{\max}^{\text{MDO}} = \frac{1}{M} \sum_{m=1}^M d_{\max}^{\text{inp}m}.$$

Then, d_{\max}^{MDO} represents the overall upper bound of the change we need to make.

We also define a norm to judge about the robustness of MV optimization (just like we do in Section 1). We calculate the difference between the MV optimal portfolio based on the original input parameters and the MV optimal portfolio based on variation m :

$$\|\mathbf{w}_1 - \mathbf{w}_1^{\text{inp}m}\|.$$

This represents the change we need to make if we want to keep an MV optimal portfolio with the same risk aversion level. We compute the average over all variations, define:

$$\bar{d}^{\text{MV}} = \frac{1}{M} \sum_{m=1}^M \|\mathbf{w}_1 - \mathbf{w}_1^{\text{inp}m}\|.$$

So \bar{d}^{MV} represents the average change in MV optimal portfolios. Note that here, we use a different notation for the robustness of MV optimization than we do in Section 1.3.1 and Section 2.2.2, but the calculation is the same.

Note that we use different measures to judge the robustness: for MV optimization, we use a straightforward measure, but for MDO, we use three less obvious measures. We should keep this in mind when we compare these norms.

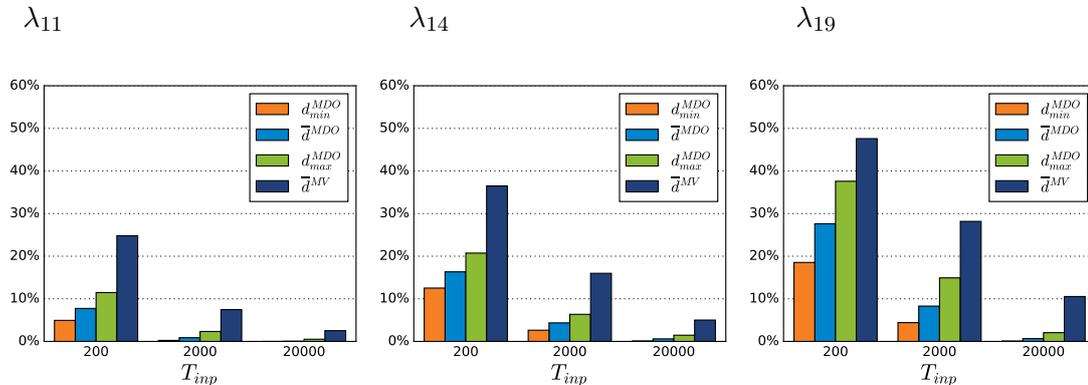
4.5.6 The robustness effect of MDO

Now that we have norms to judge the robustness of MDO, let us look at our example. Use again risk aversion parameters $\lambda_k = ((k-1)/25 + 1)^{-9}$, $k = 1, \dots, 25$. We vary the input parameters using T_{inp} drawings from $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, just as we do in Section 2.2.2. Take $M = 100$ variations of input parameters, $H = 200$ distance calculations from points in the original convex hull towards the convex hull of a variation, a permissible area corresponding to $c_1 = 0.08$ and $c_2 = 2$, and levels of input parameter deviation corresponding to $T_{\text{inp}} = 200, 2000, 20000$. We have that $T_{\text{inp}} = 200$ means that we are dealing with modified input parameters, which highly differ from the original input parameters and that $T_{\text{inp}} = 20000$ means that the modified input parameters do not differ much.

Recall that \bar{d}^{MV} denotes the average change in MV optimal portfolios, hence, it represents the robustness of MV optimization. Furthermore, for MDO, we have that \bar{d}^{MDO} denotes the average change in convex combination portfolios, d_{\min}^{MDO} represents the (average) lower bound of the change in a convex combination portfolio, and d_{\max}^{MDO} represents the (average) upper bound of the change in a convex combination portfolio.

Figure 4.22 shows the results for three different risk aversion parameters. Note that we again scale to a maximum distance of $\sqrt{2}$, corresponding to 100%.

FIGURE 4.22: Evaluating the robustness for different risk aversion parameters of MV optimization and MDO, with $c_1 = 0.08$ and $c_2 = 2$.

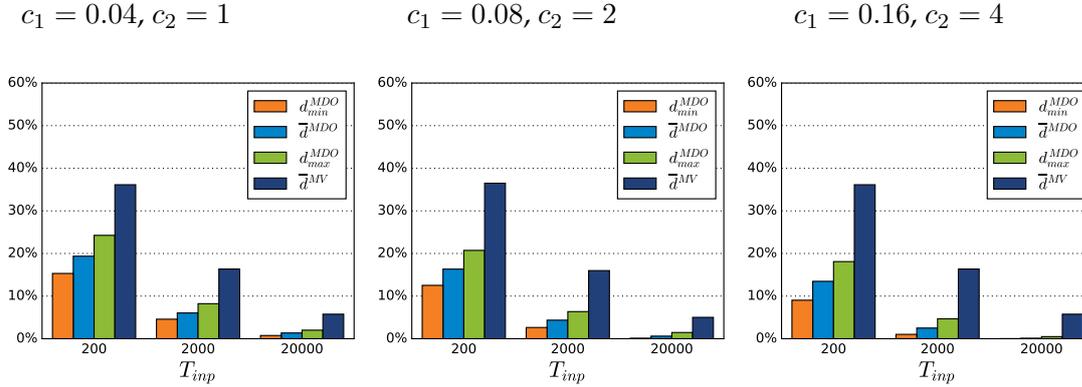


Because we compare two different measures for MV optimization and MDO, it is difficult to directly draw any conclusions. However, let us interpret the results. When we look at the results of λ_{11} with $T_{inp} = 200$, we see that varying the input parameters results for MV optimization in an average deviation of approximately 25%. For MDO, we see that the lower bound of deviation is 5%, the average deviation is 8%, and the upper bound of deviation is 11%, approximately. Let us explain what this means. Note that these values are based on $M = 100$ variations and we look at the average results of those variations. To simplify the explanation, we assume here that we deal a single variation of input parameters that represents these average results. Suppose we initially have a convex combination portfolio and we repeat the MDO procedure, this time based on new input parameters. Then, in the best possible case we need to change 5% of our initial portfolio to end up in the convex hull of the variation, on average we need to change 8%, and in the worst possible case we need to change about 11%. So, in the worst case, we still need to change less than when we do in MV optimization. We see that this is the case for all three risk aversion parameters and all levels of input parameter deviation. The convex hull of maximum distance portfolios is thus more robust than a single MV optimal portfolio.

The fairest way to judge the robustness of MV optimization and MDO is to compare the average deviation of an MV optimal portfolio, \bar{d}^{MV} , to the average deviation of a maximum distance portfolio, \bar{d}^{MDO} . We see that the difference between these two is biggest for λ_{19} and $T_{inp} = 200$: for MV optimization we have an average deviation of 47%, approximately, while for MDO we have an average deviation of about 28%. While we know that we can not bluntly compare them, we set this aside for a moment and we conclude that an increase in robustness of about 19% is advantageous. Also in other cases, we have big improvements. Look for example at the case of λ_{14} with $T_{inp} = 2000$. Here, we have an increase in robustness from 16% to 4%, approximately. We can conclude that for different risk aversion parameters and different orders of input parameter deviation, we experience a higher level of robustness in MDO than we do in MV optimization.

Let us analyze the influence of the size of the permissible area. Take λ_{14} and take three different sizes of the permissible area. We take again a small area: $c_1 = 0.04$, $c_2 = 1$, a medium sized area: $c_1 = 0.08$, $c_2 = 2$, and a large area: $c_1 = 0.16$, $c_2 = 4$. Figure 4.23 shows the results.

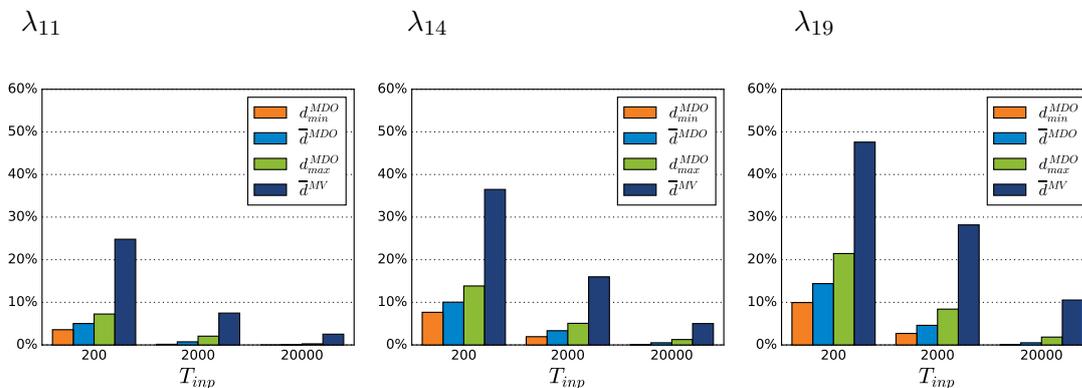
FIGURE 4.23: Evaluating the robustness for different sizes of the permissible area of MV optimization and MDO, with λ_{14} .



We see that for a larger size of the permissible area we obtain smaller values of d_{min}^{MDO} , \bar{d}^{MDO} , and d_{max}^{MDO} . However, the differences are small: multiplying c_1 and c_2 by a factor of 4 results for example for $T_{inp} = 200$, in a decrease of \bar{d}^{MDO} from 19% to 13%. Hence, enlarging the permissible area has a small but positive effect on the robustness of MDO.

Let us analyze the case of $c_1 = 0.08$ and $c_2 = 2$ in more detail. We see from Figure 4.23 that for example for λ_{14} and $T_{inp} = 200$, we have $\bar{d}^{MDO} = 16\%$. This means that when we have a portfolio in the original convex hull, we need to deviate our portfolio on average 16% to end up in the convex hull of a variation. But what happens when we turn this around? Suppose we pick a certain convex combination portfolio in the convex hull of a variation and we calculate the distance towards the original convex hull. That is, for variation m , pick a random point $h = 1, \dots, H$ in $\text{CH}(\mathbf{w}_1^{\text{inp}m}, \dots, \mathbf{w}_{I_m}^{\text{inp}m})$ and calculate the distance towards $\text{CH}(\mathbf{w}_1, \dots, \mathbf{w}_I)$. What is now the average, minimum, and maximum distance? We let d_{min}^{MDO} , \bar{d}^{MDO} , and d_{max}^{MDO} denote these distances. Beware that we use the same notation as we introduce in Section 4.5.5, but this time the norms are based on different calculations. For risk aversion parameters, λ_{11} , λ_{14} , and λ_{19} , and deviation of input parameters corresponding to $T_{inp} = 200$, $T_{inp} = 2000$, and $T_{inp} = 20000$, we obtain the results of Figure 4.24.

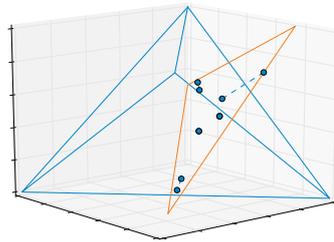
FIGURE 4.24: Evaluating the robustness for different risk aversion parameters of MV optimization and MDO, with $c_1 = 0.08$ and $c_2 = 2$. We pick points in the convex hull of a variation and calculate the distance towards the original convex hull.



When we compare these results to those of Figure 4.22, we see that here all values for d_{\min}^{MDO} , \bar{d}^{MDO} , and d_{\max}^{MDO} are lower. Note that the difference between the two calculations is biggest for $T_{\text{inp}} = 200$, small but visible for $T_{\text{inp}} = 2000$, and almost not visible for $T_{\text{inp}} = 20000$.

These differences tell us that it matters whether we do the calculation from the original convex hull towards the convex hull of a variation, or the other way around. We can explain this difference by looking at the number of maximum distance portfolios for variation m : I_m . To get an idea, let us go back to the three dimensional example from Section 4.5.5. We calculate the distances of 5 points in the convex hull of a variation (orange) towards the original convex hull (blue). Figure 4.25 shows the results.

FIGURE 4.25: Example of calculating the distance between from the convex hull of a variation (orange) towards the original convex hull (blue).

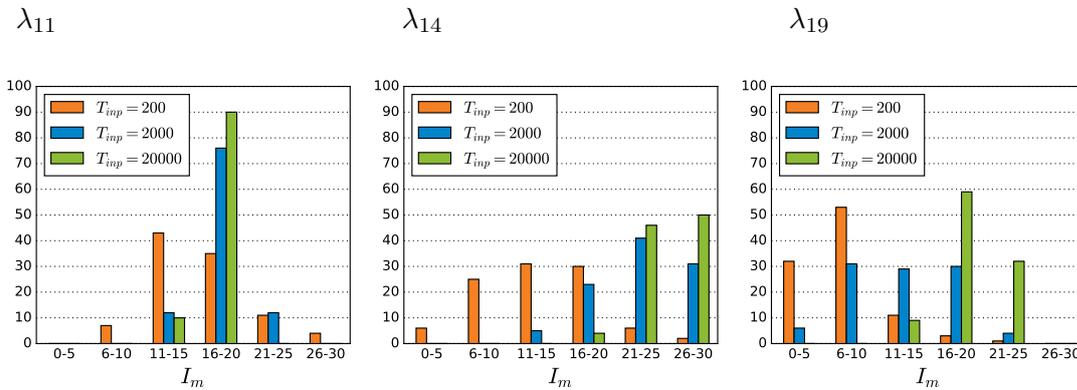


We see that here, in two cases we have a distance of zero and in the other three cases we have a positive distance. Note that these distances are in general smaller than the distances from Figure 4.21. From these two figures, it is clear that it matters which way we do the calculation. We expect the distances from a convex hull corresponding to a large number of maximum distance portfolios, towards a convex hull corresponding to a small number of maximum distance portfolios, to be big. Analogously, we expect the distances from a convex hull corresponding to a small number of maximum distance portfolios, towards a convex hull corresponding to a large number of maximum distance portfolios, to be small.

Let us investigate the number of maximum distance portfolios that we obtain. Recall that we want to obtain $\mathcal{I} = 30$ maximum distance portfolios, but we obtain $I \leq \mathcal{I}$ portfolios based on the original input parameter and we obtain $I_m \leq \mathcal{I}$ portfolios based on variation m . Based on the original input parameters we have the following number of maximum distance portfolios. For λ_{11} : $I = 18$, for λ_{14} : $I = 29$, and for λ_{19} : $I = 17$. To analyze the number of maximum distance portfolios of the variations, we divide these values in the following buckets: 0-5 portfolios, 6-10 portfolios, 11-15 portfolios, 16-20 portfolios, 21-25 portfolios, and 26-30 portfolios. We display the values for $T_{\text{inp}} = 200$, $T_{\text{inp}} = 2000$, and $T_{\text{inp}} = 20000$, separately. Figure 4.26 shows the results. We see that, for example for λ_{14} and $T = 20000$, there are 50 of the total of 100 variations in which we obtain between the 26 and 30 maximum distance portfolios.

Let us look at the case of λ_{11} . We see that for $T_{\text{inp}} = 200$ we have a wide range of values of I_m , for $T_{\text{inp}} = 2000$ the range is less wide and more concentrated in the bucket of 16-20 portfolios, and for $T_{\text{inp}} = 20000$ almost all variations obtain between the 16 and 20 maximum distance portfolios. It seems that for higher values of T_{inp} , we are more likely to obtain a value for I_m that is close to the number of maximum distance portfolios based on the original input parameters: $I = 18$. We see this also for λ_{14} and λ_{19} : for λ_{14} we have $I = 29$ and we have that for $T_{\text{inp}} = 20000$, I_m is most concentrated in the bucket

FIGURE 4.26: Evaluating the number of maximum distance portfolios for variations $m = 1, \dots, 100$, with $c_1 = 0.08$ and $c_2 = 2$ and different risk aversion parameters and levels of input parameter deviation.



of 26-30 maximum distance portfolios, and for λ_{19} we have $I = 17$ and we have that for $T_{inp} = 20000$, I_m is most concentrated in the bucket of 16-20 maximum distance portfolios.

We see that for all risk aversion parameters, in general, we obtain less portfolios for a smaller value of T_{inp} . An explanation for this can be that for small T_{inp} , a certain balance between the asset classes is disturbed. Maybe only a few asset classes attain favorable characteristics, like a high expected return and/or a low expected risk, and the other asset classes attain unfavorable characteristics. Therefore, we can not include the other asset classes when we search for maximum distance portfolios. This way, we obtain only a few maximum distance portfolios for small T_{inp} .

We can now explain the difference between calculations from the original convex hull towards the convex hull of a variation (Figure 4.22) and the calculations of the convex hull of a variation towards the original convex hull (Figure 4.24). Recall that the difference between the two calculations is biggest for $T_{inp} = 200$, small but visible for $T_{inp} = 2000$, and almost not visible for $T_{inp} = 20000$. We see in Figure 4.26 that for $T_{inp} = 200$ we obtain in general less maximum distance portfolios than the number of maximum distance portfolios based on the original input parameters, I . Hence we are in a similar situation as that of Figure 4.25. Therefore, the distance from the convex hull of a variation towards the original convex hull is in general smaller than the other way around.

4.5.6.1 Conclusion

Although we measure the robustness of MDO and MV optimization in two different ways, we can conclude that we obtain a higher level of robustness in MDO than we do in MV optimization. The difference in robustness between MDO and MV optimization depends highly on the chosen convex combination portfolios, on the risk aversion parameter under consideration and on the order of input parameter deviation, and it depends slightly on the size of the permissible area.

Chapter 5

Applications

In this chapter, we briefly discuss the applications of the three methods that we consider in this thesis: resampling, shrinkage, and maximum distance optimization. We explain per method the robustness performance, the choices we need to make, and what advantages and disadvantages we encounter. Based on these aspects we formulate an advice to Ortec Finance. We note that there are more methods in the literature than we discuss in this thesis, hence the first advice we give is to do further research to other methods as well. We first discuss the shrinkage method and we continue with resampling and maximum distance optimization.

5.1 Shrinkage

Shrinkage towards a constant correlation covariance matrix is not useful to Ortec Finance for making portfolios more robust. This method is useful if the number of asset classes is large compared to the sample size, as we show in Section 3.3.2.

In the application Ortec Finance has in mind, the sample size equals a large number of scenarios generated in a Monte Carlo simulation and therefore, this method has little added value. The method, however, can be of use for Ortec Finance for modeling asset classes with little historical data.

5.2 Resampling

In Section 2.2.2, we show that resampling produces more robust portfolios. Moreover, resampling adds asset classes with near optimal characteristics leading to more diversified portfolios.

The robustness effect of resampling follows from the diversification effect. Resampling results in a diversified portfolio and if we repeat the same procedure but this time based on a variation of input parameters, we obtain again a diversified portfolio. And hence, as we showed in Section 2.2.2, the difference between those two portfolios is small.

In resampling, the order of the robustness and diversification effect depends on the number of drawings from a normal distribution, $\mathcal{N}(\mu, \Sigma)$. That is, it depends on the parameter T . A large value for T results in a small effect and a small value for T results in a large effect. However, in the latter case, the effect of robustness and diversification comes at a cost: we give in on the expected portfolio risk and return. Hence, we want to make a smart decision about the value for T but there is no clear guideline to do so. However,

if we have an idea over how uncertain the input parameters are, this helps us to pick a value for T .

We note that resampling can produce portfolios such that small shreds of weights are given to some asset classes. Investing, for example, less than one percent of the budget in some asset class is not desirable.

Although not treated in this thesis, the resampling method is applicable to other risk measures and other optimization programs. A risk measure that is often used at Ortec Finance is the conditional value at risk (CVaR). For an optimization program that maximizes the expected return and minimizes the expected CVaR, this method can also be helpful.

A disadvantage, however, is that resampling lacks a theoretical foundation. Intuitively, the method makes sense, but a proper Bayesian framework for example would give the method a proper theoretical foundation. This issue is often discussed in papers which compare resampling with Bayesian optimization, see for example [5].

Lastly, we want to note that there is a possibility on a convex part in the resampled frontier. In Section 2.2.3, we show that we probably do not have to worry about this phenomenon. We assume a resampling method that only varies the return vector and we prove in Theorem 2.2.2 that the probability of a convex part in the resampled frontier converges to zero as the number of resamplings goes to infinity. Because we use a variation of the resampling method, this is no guarantee for the original resampling method but it is reassuring.

5.2.1 Advice

Resampling has a positive effect on robustness and diversification. Before using resampling, one needs to make a well considered decision about the parameter T . This parameter determines a position in a tradeoff between robustness and diversification effect and the loss in expected portfolio risk and return. Resampling is a useful method and we would recommend to use it. However, we explicitly note that our research is based on only one example. Before using resampling in practice, we suggest further research based on other examples and we suggest to compare resampling to other methods such as Bayesian optimization or Black Litterman [2] inverse optimization.

5.3 Maximum distance optimization

Maximum distance optimization (MDO) searches for portfolios that are close to optimal in terms of risk and return levels and are far away in terms of allocations. This way, we thoroughly investigate the area closely beneath the efficient frontier.

An advantage of MDO compared to other methods is that it fits well with how a consultant of Ortec Finance forms his advice. A consultant combines his expertise with outcomes from the models. In MDO, we obtain a whole range of near optimal portfolios and any convex combination also results in a portfolio that is near optimal. Therefore, a consultant can pick a convex combination portfolio that he considers to be best. Moreover, the allocations of the near optimal portfolios give a lot of insight. We see what asset classes are not selected by MV optimization but result however in portfolios that are almost optimal.

In Section 4.5.6, we see that MDO results in a range of maximum distance portfolios that is more robust than a single MV optimal portfolio is.

The diversification effect of MDO depends highly on what convex combination portfolio we select. Also, both the diversification and the robustness effect of MDO depends on the size of the area beneath the efficient frontier that we investigate, that is, the permissible area. However, the influence of the choice for the permissible area on the robustness of MDO is limited. For the diversification effect, if we choose a large permissible area, we are likely to obtain portfolios that highly differ in terms of allocations and therefore, we are able to select a diversified convex combination portfolio. If we choose a small permissible area, we are likely to obtain portfolios that do not differ a lot in terms of allocations and therefore, it is likely that we are not able to select a diversified convex combination portfolio. However, for a small permissible area, we do not necessarily obtain portfolios with similar allocations, as we discuss in the remarkt below Theorem 4.5.1.

A disadvantage of MDO is that the implementation requires a heuristic method. We use a Basin hopping algorithm as heuristic method and we are satisfied about its performance. Good results, however, require a long computation time.

5.3.1 Advice

We advise Ortec Finance to use maximum distance optimization, since the range of maximum distance portfolios is robust. Moreover, MDO fits well with how Ortec Finance uses models in combination with their own expert views, to form their advice. MDO gives insight in what asset classes are not selected by Mean-Variance optimization but results however in portfolios that are only near optimal. Also, MDO, opposed to other methods, gives freedom in selecting a convex combination portfolio that fits well with the expert views.

We also advise Ortec Finance to extend this method to other risk measures, such as CVaR. The link of Support Vector Machines to calculating the distance towards a convex hull is one that can be used in other optimization programs. Therefore, an extension of MDO to other risk measures is a logical next step. However, we expect that this extension is only possible if the other risk measure is convex. Otherwise, it is possible that a convex combination portfolio ends up outside the permissible area.

Bibliography

- [1] Charalambos D Aliprantis and Kim Border. *Infinite dimensional analysis: a hitchhiker's guide*. Springer Science & Business Media, 2006.
- [2] Dimitris Bertsimas, Vishal Gupta, and Ioannis Ch Paschalidis. "Inverse optimization: A new perspective on the Black-Litterman model". In: *Operations research* 60.6 (2012), pp. 1389–1403.
- [3] Vijay K Chopra and William T Ziemba. "The effect of errors in means, variances, and covariances on optimal portfolio choice". In: *The Kelly Capital Growth Investment Criterion: Theory and Practice* 3 (2011), p. 249.
- [4] *docs.scipy.org*. <https://docs.scipy.org/doc/scipy-0.18.1/reference/generated/scipy.optimize.basinhopping.html>. Accessed: 2016-11-21.
- [5] Campbell R Harvey, John C Liechty, and Merrill W Liechty. "Bayes vs. resampling: a rematch". In: (2008).
- [6] Olivier Ledoit and Michael Wolf. "Honey, I shrunk the sample covariance matrix". In: *UPF Economics and Business Working Paper* 691 (2003).
- [7] Olivier Ledoit and Michael Wolf. "Improved estimation of the covariance matrix of stock returns with an application to portfolio selection". In: *Journal of empirical finance* 10.5 (2003), pp. 603–621.
- [8] Harry Markowitz. "Portfolio selection". In: *The journal of finance* 7.1 (1952), pp. 77–91.
- [9] *nobelprize.org*. http://www.nobelprize.org/nobel_prizes/economic-sciences/laureates/1990/press.html. Accessed: 2016-01-12.
- [10] Bernd Scherer. "Portfolio resampling: Review and critique". In: *Financial Analysts Journal* 58.6 (2002), pp. 98–109.