Caroline BAYR, Bakk.techn.

A comparison of the Markowitz portfolio optimization with some robust counterparts

MASTERARBEIT

zur Erlangung des akademischen Grades einer Diplom-Ingenieurin

Masterstudium Finanz- und Versicherungsmathematik

Graz University of Technology

Technische Universität Graz

Betreuerin:
Ao. Univ.-Prof. Dipl.-Ing. Dr.techn. Eranda DRAGOTI-CELA

Institut für Optimierung und Diskrete Mathematik (Math B)

Graz, im Mai 2010

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitely marked all material which has been quotes either literally or by content from the used sources.

.... date

.... (signature)

Preface

This master thesis is about portfolio optimization considering uncertain input parameters.

In the first chapter, we introduce the classical mean-variance optimization model of Harry Markowitz. We assume that the expected returns and the covariance matrix, which is used as risk measure, are known. Some related portfolio optimization problems and the corresponding solution methods are discussed.

In the second chapter we define robust optimization in general and describe different problems: uncertainty in the objective function, uncertainty in the constraints and the concept of relative robustness. Further we discuss some strategies like resampling of data and solution methods for conic optimization problems needed when dealing with robust optimization problems.

The robust portfolio optimization is considered in the third chapter. There we assume, that the knowledge about expected returns and the covariance matrix is uncertain; for these inputs we only have estimations leading to uncertain input data. We include these data in the optimization problem by assuming that they lie in so-called uncertainty sets. The resulting optimization problems are formulated and solved.

In the last chapter, the models described in the previous chapters of the thesis are applied to an instance of a portfolio optimization problem focussing on the comparison of classical optimization models with their robust counterparts.

I want to thank Ao. Univ.-Prof. Dipl.Ing Dr.techn. Eranda Dragoti-Cela for the faithful supervision, the patience and the helpful advices. I also want to thank all people, who helped me with this master thesis, especially my parents for supporting me during my study time and my boyfriend Andreas Angermann for his motivation all the time.

All errors are my own.

Graz, May 2010 Caroline Bayr

Contents

1 Markowitz: Classical Mean-Variance Optimization

1.1 Introduction

This chapter is based on the paper of Steinbach [1] and the paper of A. and D. Niedermayer [2]. The duality theory is from the script of Burkard [3] and the book of Fabozzi et al. [5], Chapter 9.

In this chapter we discuss the classical portfolio optimization by Harry Markowitz, the so-called *mean-variance optimization*. Markowitz proposed an optimization model to get a high reward with low risk. This model is sufficiently simple to be solved numerically and can be used in practice.
First of all we will analyze the single-period mean-variance model. There we show the optimization problem for several assumptions: First we discuss the problem with risky assets only. Then we include the possibility to invest in riskless cash and the third step is to look at the optimization of investing in an account with guaranteed total loss. We will also consider the influence of constraints with inequalities instead of equations. Since the choice of the risk measure influences the solutions of an optimization problem, we want to discuss the downside risk optimization too where we use the semi-variance instead of the variance as risk measure. We need some certain assumptions of the return distributions.
Then we want to consider the multi-period mean-variance model. In this context we need so-called scenario trees.

Throughout this master thesis we assume that short selling is not allowed.

1.2 Single-period mean-variance model

In our considerations we can invest in n assets over a certain period of time. A portfolio over these n assets is specified by the so-called *portfolio vector* $w \in \mathbb{R}^n$, where w_i is the percentage of capital invested in asset i. Let us denote the random vector of asset returns by $r \in \mathbb{R}^n$. Denote by p_i the price of asset i at time t_i and by p_{i+1} the price of asset i at time t_{i+1} for $1 \leq i \leq n$. Then the return of asset i is given as $r_i = \frac{p_{i+1}-p_i}{p_i}$. At the end of the period $[t_i, t_{i+1}]$ the portfolio return is given as $R = \sum_{i=1}^{n} r_i w_i = r'w$, where r' denotes the transposed vector of r. Assume that the asset returns have expectation $\mu_r := \mathbb{E}[r]$ and covariance matrix

$$\Sigma := \mathbb{E}[(r-\mu_r)(r-\mu_r)'] = \mathbb{E}[rr'] - \mu_r\mu_r'.$$

The total return of the portfolio is a random variable which depends on the portfolio vector (w_i): $R = r'w$. The aim is to determine a portfolio vector w which leads to a return distribution fullfilling the investor's needs. In this context we define two important quantities, reward and risk, as follows:

Definition 1.1: **Reward**
The *reward* of a portfolio is the mean of its return,

$$\mu(w) := \mathbb{E}[R] = \mathbb{E}[r'w] = \mu_r'w.$$

Definition 1.2: **Risk**
The *risk* of a portfolio is the variance of its return,

$$\sigma^2(w) := Var[R] = \mathbb{E}[(r'w - \mathbb{E}[r'w])^2] = \mathbb{E}[w'(r-\mu_r)(r-\mu_r)'w] = w'\Sigma w.$$

We want to maximize the reward and minimize the risk leading to the optimization problem:

$$\max_w \left\{ c\mu(w) - \frac{1}{2}\sigma^2(w) \right\} \tag{1.1}$$

$$\begin{aligned} s.t.\ e'w &= 1, \\ w_i &\geq 0,\ \forall i, \end{aligned}$$

where $e \in \mathbb{R}^n$ denotes the vector of all ones. The investor tries to get a good trade-off between reward and risk. The equation $e'w = 1$ is called the *budget equation* and specifies the initial wealth.
Problem 1.1 can be reformulated as one of the following two optimization problems, which are dual to each other. For understanding we give a short discussion to duality of optimization problems (see Fabozzi et al. [5], Chapter 9).
The variables of a dual problem are related to variables of the corresponding primal problem. If the primal optimization problem is a maximization then the dual optimization problem is a minimization and conversely, if the primal problem is minimization then the dual problem will be maximization. The number of variables of the primal problem is equal to the number of constraints of the dual problem and vice versa. There are several advantages of dual optimization problems:

- The dual problem is often better tractable from a theoretical or computational point of view. This can be used to compute the primal and dual solutions.
- If we have a convex optimization problem, then we can solve the dual problem and the objective value is the same as in the primal problem.

So we use dual problems, because they are often easier to solve than the primal problems.

Dual optimization problems also play a major part in formulations of robust optimization problems, which are discussed in the following chapters. Now we want to summarize briefly how to obtain the dual of a problem given.
Consider the following primal optimization problem:

$$\min_x f(x) \tag{1.2}$$

$$s.t.\ g_i(x) \leq 0, i = 1, \dots, n.$$

First we formulate the Lagrangian to put the constraints of the primal problem in the objective function for which we use n nonnegative multipliers u_i:

$$L(x, u) = f(x) - u'g(x).$$

Now we construct the dual function:

$$L^*(u) = \min_x \left\{ f(x) + u'g(x) \right\}.$$

Finally we can formulate the dual optimization problem:

$$\max_u L^*(u) \tag{1.3}$$

$$s.t.\ u \geq 0.$$

In the following passage we review some basics of duality theory in linear and quadratic optimization. First consider linear optimization problems:

- Primal problem:

$$\max_x c'x \tag{1.4}$$

$$\begin{aligned} s.t.\ Ax &\leq b, \\ x &\geq 0. \end{aligned}$$

- Dual problem:

$$\min_u b'u \tag{1.5}$$

$$\begin{aligned} s.t.\ A'u &\geq c, \\ u &\geq 0. \end{aligned}$$

We can formulate two theorems about the solutions of primal and dual problems:

Theorem 1.1: Weak Duality
For every feasible solution x of the primal problem and every feasible u of the dual problem holds:

$$c'x \leq b'u.$$

Proof:
From $A'u \geq c,\ x \geq 0$ and $Ax \leq b$ follows:

$$c'x \leq u'Ax \leq u'b.$$

□

Theorem 1.2: Duality
If one of two dual linear optimization problems has a finite optimal solution, then the other problem has a finite optimal solution and for the optimal values x^* and u^* of the objective function holds:

$$c'x^* = b'u^*.$$

For the proof, see Burkard [3], page 87.

Now we consider the duality in quadratic optimization:

- Primal problem:

$$\min_x \left\{ \frac{1}{2} x' Q x + c' x \right\} \quad (1.6)$$
$$s.t. \ Ax \geq b.$$

- Dual problem:

$$\max_u \left\{ u' b - \frac{1}{2}(c - A'u)' Q^{-1} (c - A'u) \right\} \quad (1.7)$$
$$s.t. \ u \geq 0.$$

Look up the book of Fazozzi et al. [5], Chapter 9.

Now consider the mean-variance optimization Problem 1.1 again. An equivalent formulation of Problem 1.1 is the following:

$$\max_w \mu(w) \quad (1.8)$$

$$\begin{aligned} s.t. \ & e'w = 1, \\ & \sigma^2(w) \leq s. \\ & w_i \geq 0, \ \forall i. \end{aligned}$$

The equivalence of Problem 1.1 and 1.8 will be shown in Section 1.2.1.

In this model we maximize the reward subject to the budget equation and the constraint that the risk is lower than a fixed level s.

The dual problem is to minimize the risk subject to the budget equation and the constraint that the reward is fixed at level m.

$$\min_w \frac{1}{2}\sigma^2(w) \tag{1.9}$$

$$\begin{aligned} s.t.\ & e'w = 1, \\ & \mu(w) = m, \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

Problem 1.8 has a linear objective function and convex quadratic restrictions and Problem 1.9 has a convex quadratic objective function and linear restrictions. The convexity of the quadratic restrictions or the quadratic goal function, follows from the positive semidefinitness of Σ as a covariance matrix. So numerically Problem 1.9 is easier to solve than Problem 1.8.

Further in this section the relationship between these optimization problems is discussed. We extend the general single-period model by including riskless cash, guaranteed loss, inequalities and downside risk.

1.2.1 Risky assets

Now consider the simplest situation with a portfolio consisting of n risky assets only. We need two assumptions on the return distribution.

Assumptions:

- **(A1)** $\Sigma > 0$, the covariance matrix is positive definite.:
  All n assets and any convex combination of them are risky.

- **(A2)** μ_r is not a multiple of e: $\mu_r \neq ke$ for $k \in \mathbb{N}$:
  This implies $n \geq 2$ and guarantees that the situation does not degenerate. Otherwise the optimal portfolio of Problem 1.1 would always be the same: $w = \frac{\Sigma^{-1}e}{e'\Sigma^{-1}e}$ regardless of the trade-off parameter c. Furthermore the constraints of Problem 1.9 would be violated except of one specific value of the desired reward, namely $m = \frac{\mu_r' e}{n}$.

To solve Problem 1.1 we minimize the negative utility function and get the following optimization problem:

$$\min_{w} \left\{ \frac{1}{2} w' \Sigma w - c\mu_r' w \right\} \tag{1.10}$$

$$s.t. \; e'w = 1,$$
$$w_i \geq 0, \; \forall i.$$

The Lagrangian is

$$L(w, \lambda; c) = \frac{1}{2} w' \Sigma w - c\mu_r' w - \lambda(e'w - 1).$$

$$\frac{\partial L}{\partial w} = w'\Sigma - c\mu_r - \lambda e = 0. \tag{1.11}$$

The optimal solution for the portfolio vector w is

$$w^* = \Sigma^{-1}[c\mu_r + \lambda e].$$

To get the optimal multiplier λ we substitute the optimal w in the budget equation:

$$e'[\Sigma^{-1}[c\mu_r + \lambda e]] = 1$$

$$\lambda^* = \frac{1 - ce'\Sigma^{-1}\mu_r}{e'\Sigma^{-1}e}. \tag{1.12}$$

The optimal reward m is obtained by substitution of the optimal w in $m = \mu_r' w$:

$$m^* = c\mu_r'\Sigma^{-1}\mu_r + \lambda e'\Sigma^{-1}\mu_r = \frac{c(e'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - [e'\Sigma^{-1}\mu_r]^2) + e'\Sigma^{-1}\mu_r}{e'\Sigma^{-1}e} \tag{1.13}$$

This solution is unique because the objective is strongly convex and Constraint 1.11 is of full rank.

Thus Problem 1.9 is easier to solve numerically.

$$\min_{w} \frac{1}{2} w' \Sigma w \tag{1.14}$$

$$\begin{aligned} s.t.\ & e'w = 1, \\ & \mu_r w = m, \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

The Lagrangian is

$$\begin{aligned} L(w, \lambda, c; m) =& \frac{1}{2} w' \Sigma w - \lambda(e'w - 1) - c(\mu_r w - m). \\ \frac{\partial L}{\partial w} =& w' \Sigma - \lambda e - c\mu_r = 0. \end{aligned}$$

The optimal solution for the portfolio vector w is

$$w^* = \Sigma^{-1}(\lambda e + c\mu_r).$$

To get the optimal value c we substitute the optimal value of w in the budget equation:

$$e'(\Sigma^{-1}(\lambda e + c\mu_r)) = 1.$$

Hence we get λ:

$$\lambda = \frac{1 - e'\Sigma^{-1}\mu_r c}{e'\Sigma^{-1}e}. \tag{1.15}$$

By substituting the optimal w in the equation $m = \mu_r' w$ we get

$$m = \mu_r(\Sigma^{-1}(\lambda e + c\mu_r)). \tag{1.16}$$

In Equation 1.16 we substitute Equation 1.15 for λ and we get the optimal c:

$$
\begin{aligned}
m =& \mu_r(\Sigma^{-1}((\frac{1-e'\Sigma^{-1}\mu_r c}{e'\Sigma^{-1}e})e + c\mu_r)). \\
m =& \frac{\mu_r'\Sigma^{-1}e - e'\Sigma^{-1}\mu_r e'\Sigma^{-1}\mu_r c}{e'\Sigma^{-1}e} + c\mu_r'\Sigma^{-1}\mu_r. \\
me'\Sigma^{-1}e - \mu_r\Sigma^{-1}e =& c(-(e'\Sigma^{-1}\mu_r)^2 + \mu_r'\Sigma^{-1}\mu_r e'\Sigma^{-1}e).
\end{aligned}
$$

$$
c^* = \frac{me'\Sigma^{-1}e - \mu_r\Sigma^{-1}e}{\mu_r'\Sigma^{-1}\mu_r e'\Sigma^{-1}e - (e'\Sigma^{-1}\mu_r)^2}. \tag{1.17}
$$

For the optimal λ we substitute the optimal c from 1.17 in Equation 1.15:

$$
\begin{aligned}
\lambda =& \frac{1 - e'\Sigma^{-1}\mu_r(\frac{me'\Sigma^{-1}e - \mu_r\Sigma^{-1}e}{\mu_r'\Sigma^{-1}\mu_r e'\Sigma^{-1}e - (e'\Sigma^{-1}\mu_r)^2})}{e'\Sigma^{-1}e}. \\
\lambda =& \frac{\mu_r'\Sigma^{-1}\mu_r e'\Sigma^{-1}e - (e'\Sigma^{-1}\mu_r)^2 - e'\Sigma^{-1}\mu_r e'\Sigma^{-1}em + (e'\Sigma^{-1}\mu_r)^2}{e'\Sigma^{-1}e(\mu_r'\Sigma^{-1}\mu_r e'\Sigma^{-1}e - (e'\Sigma^{-1}\mu_r)^2)}.
\end{aligned}
$$

$$
\lambda^* = \frac{\mu_r\Sigma^{-1}\mu_r - e'\Sigma^{-1}\mu_r m}{\mu_r'\Sigma^{-1}\mu_r e'\Sigma^{-1}e - (e'\Sigma^{-1}\mu_r)}. \tag{1.18}
$$

Theorem 1.3:
Problem 1.10 with parameter c and Problem 1.14 with parameter m are equivalent if and only if c equals the optimal reward multiplier λ of Problem 1.14, or equivalently, m equals the optimal reward multiplier λ of Problem 1.10.

Proof:
The conditions $c = \frac{e'\Sigma^{-1}em - e'\Sigma^{-1}\mu_r}{e'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - [e'\Sigma^{-1}\mu_r]^2}$ and $m = \frac{c(e'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - [e'\Sigma^{-1}\mu_r]^2) + e'\Sigma^{-1}\mu_r}{e'\Sigma^{-1}e}$ are equivalent. For the optimal reward multipliers of Problem 1.10 and 1.14 follows that they are identical:

$$
\begin{aligned}
\lambda_1 =& \frac{1 - ce'\Sigma^{-1}\mu_r}{e'\Sigma^{-1}e} \\
=& \frac{e'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - [e'\Sigma^{-1}\mu_r]^2 - e'\Sigma^{-1}ee'\Sigma^{-1}\mu_r m + [e'\Sigma^{-1}\mu_r]^2}{e'\Sigma^{-1}ee'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - [e'\Sigma^{-1}\mu_r]^2} \\
=& \frac{e'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - e'\Sigma^{-1}ee'\Sigma^{-1}\mu_r m}{e'\Sigma^{-1}ee'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - [e'\Sigma^{-1}\mu_r]^2} \\
=& \frac{\mu_r'\Sigma^{-1}\mu_r - e'\Sigma^{-1}\mu_r m}{e'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - [e'\Sigma^{-1}\mu_r]^2} = \lambda_2.
\end{aligned}
$$

So the optimal portfolios are equivalent.

□

To continue we need the following definition of an *efficient frontier* and a *Pareto-optimal* solution.

Definition 1.3: Efficient frontier
The *efficient frontier* is a curvature containing points (m, σ^2) where σ^2 is the optimal risk. For a given level of return all points in this curve correspond to portfolios with lowest risk.

The restrictions of Problem 1.10 are restrictions of Problem 1.14, too. The later problem contains one more additional restriction, the so-called reward restriction. These $n+2$ restrictions define for $n+3$ variables w, λ, c, m a one-dimensional affine subspace which is parametrized by c in Problem 1.10 and by m in Problem 1.14. The optimal risk is a quadratic function of m, $\sigma^2(m)$, which graph is called the efficient frontier.

Definition 1.4: Pareto-optimality
Let X be a feasible set. A solution $x^* \in X$ is called *Pareto-optimal*, if there exists no $x \in X : f(x) < f(x^*)$.

Generally, the efficient frontier refers to the set of all Pareto-optimal solutions of an optimization problem. In our considerations it applies to the *upper branch* corresponding to $m \geq \hat{m}$ only, where $\hat{m}$ is the optimal reward. The *lower branch* corresponds to $m \leq \hat{m}$. All feasible portfolios are on the right and below the efficient frontier.

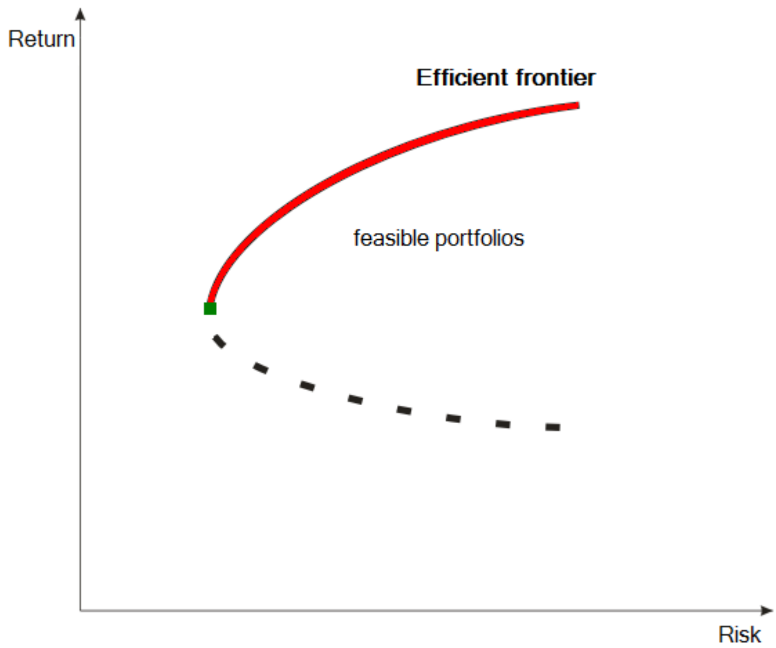

Figure 1.1: Efficient frontier of a portfolio with risky assets

Theorem 1.4:
The optimal risk in Problem 1.10 and 1.14 is

$$\sigma^2(m) = \frac{e'\Sigma^{-1}em^2 - 2e'\Sigma^{-1}\mu_r m + \mu_r'\Sigma^{-1}\mu_r}{e'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - [e'\Sigma^{-1}\mu_r]^2}.$$

It takes the global minimum over all rewards at

$$\hat{m} = \frac{e'\Sigma^{-1}\mu_r}{e'\Sigma^{-1}e}$$

and has the positive value in this case

$$\sigma^2(\hat{m}) = \frac{1}{e'\Sigma^{-1}e}.$$

The associated solution is

$$\hat{w} = \frac{\Sigma^{-1}e}{e'\Sigma^{-1}e},\ \hat{\lambda} = \frac{1}{e'\Sigma^{-1}e},\ \hat{c} = 0.$$

Proof:
We use Definition 1.2 of the optimal risk and the solution of Problem 1.14,

$$\begin{aligned}\sigma^2(m) =& w'\Sigma w = (\lambda e + c\mu_r)'\Sigma^{-1}(\lambda e + c\mu_r)\\ =&\lambda^2 e'\Sigma^{-1}e + 2\lambda c e'\Sigma^{-1}\mu_r + c^2\mu_r\Sigma^{-1}\mu_r\\ =&\lambda(\lambda e'\Sigma^{-1}e + c e'\Sigma^{-1}\mu_r) + c(\lambda e'\Sigma^{-1}\mu_r + c\mu_r\Sigma^{-1}\mu_r).\end{aligned}$$

Using the expressions on the right hand sides of Equations 1.12 and 1.13 we get

$$\sigma^2(m) = \lambda + cm.$$

With Equations 1.17 and 1.18 we get

$$\sigma^2(m) = \frac{e'\Sigma^{-1}em^2 - 2e'\Sigma^{-1}\mu_r m + \mu_r\Sigma^{-1}\mu_r}{e'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - [e'\Sigma^{-1}\mu_r]^2}. \tag{1.19}$$

Differentiating Equation 1.19 with respect to m we obtain

$$\begin{aligned}2e'\Sigma^{-1}em - 2e'\Sigma^{-1}\mu_r =&0\\ \hat{m} =&\frac{e'\Sigma^{-1}\mu_r}{e'\Sigma^{-1}e}\end{aligned}$$

Substituting this optimal minimum in Equations 1.17 and 1.18 we get the associated solution $\hat{w}, \hat{\lambda}$ and $\hat{c}$:

$$\hat{w} = \frac{\Sigma^{-1}e}{e'\Sigma^{-1}e},\ \hat{\lambda} = \frac{1}{e'\Sigma^{-1}e},\ \hat{c} = 0.$$

□

1.2.2 Risky assets and riskless cash

Consider n risky assets and also a cash account w^c with deterministic return $r^c = \mathbb{E}[r^c]$. So the portfolio includes (w, w^c) and the variables w, r, μ_r and Σ belong to the risky part.

Assumptions: We get two similar assumptions as in the previous section, but we replace (A2) by another condition where the portfolio consists just of one risky asset and the cash account.

- **(A1)** $\Sigma > 0$, the covariance matrix is positive semidefinite.
- **(A2)** $\mu_r \neq r^c e$: No degenerate situations can occur.

We formulate the following optimzation problem

$$\min_{w,w^c} \frac{1}{2} \begin{pmatrix} w \\ w^c \end{pmatrix}' \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} w \\ w^c \end{pmatrix} = \frac{1}{2} w' \Sigma w \tag{1.20}$$

$$\begin{aligned} s.t.\ & e'w + w^c = 1, \\ & \mu_r' w + r^c w^c = m, \\ & w^c \geq 0,\ w_i \geq 0,\ \forall i. \end{aligned}$$

The Lagrangian is

$$L(w, w^c, \lambda, c; m) = \frac{1}{2} w' \Sigma w - \lambda(e'w + w^c - 1) - c(\mu_r' w + r^c w^c - m). \tag{1.21}$$

By differentiating the Lagrangian 1.21 with respect to w we get

$$w' \Sigma - \lambda e - c\mu_r = 0$$

$$w = \Sigma^{-1}(\lambda e + c\mu_r). \tag{1.22}$$

By differentiating the Lagrangian 1.21 with respect to w^c we get

$$-\lambda - cr^c = 0$$

$$\lambda = -cr^c. \tag{1.23}$$

Putting Equation 1.23 in Equation 1.22 we get the optimal solution for w

$$w^* = c\Sigma^{-1}(\mu_r - r^c e). \tag{1.24}$$

By using the budget equation and substituting the optimal w from Equation 1.24 in the budget equation we get

$$w^c = 1 - c(e' \Sigma^{-1} \mu_r - r^c e' \Sigma^{-1} e). \tag{1.25}$$

By putting Equations 1.24 and 1.25 on the left hand side of the second restriction of Problem 1.20 we get the optimal c

$$c^* = \frac{m - r^c}{(\mu_r - r^c e)' \Sigma^{-1} (\mu_r - r^c e)}. \tag{1.26}$$

The optimal risk occurs to using the definition of risk $\sigma^2(m) = w'\Sigma w$ and Equations 1.24 and 1.26:

$$\sigma^2(m) = \frac{(m - r^c)^2}{(\mu_r - r^c e)' \Sigma^{-1} (\mu_r - r^c e)}.$$

Differentiating with respect to m we get

$$2(m - r^c) = 0$$

and therefore the global minimum is taken at $\hat{m} = r^c$ with $\sigma^2(\hat{m}) = 0$. The associated solution is to invest 100% of the capital in cash and the risk vanishes: $(w, w^c) = (0, 1)$ and $\hat{\lambda} = \hat{c} = 0$.

The trade-off version of Problem 1.20 is

$$\min_{w, w^c} \left\{ \frac{1}{2} w' \Sigma w - c(\mu_r' w + r^c w^c) \right\} \tag{1.27}$$

$$\begin{aligned} s.t.\ & e'w + w^c = 1, \\ & w^c \geq 0,\ w_i \geq 0,\ \forall i. \end{aligned}$$

Theorem 1.5:
Problem 1.20 with parameter m and Problem 1.27 with parameter c are equivalent if and only if $m = r^c + c(\mu_r - r^c e)' \Sigma^{-1} (\mu_r - r^c e)$.

Proof:
Analogous to the proof of Theorem 1.3.

□

For $m = r^c$ the whole capital is invested in cash. Otherwise $e'w = c(e'\Sigma^{-1}\mu_r - r^c e'\Sigma^{-1}e)$ is invested in risky assets, so the risk is positive because the optimal portfolio is a mixture of the risky portfolio $(\Sigma^{-1}(\mu_r - r^c e), 0)$ and cash $(0, 1)$.

The following theorem shows, how the risk reduces, if we have a portfolio with cash added.

Theorem 1.6:
The risk in Problem 1.20 with cash is lower than in Problem 1.14 without cash as soon as the portfolio invests in risky assets.
If $e'\Sigma^{-1}\mu_r \neq r^c e'\Sigma^{-1}e$: The efficient frontiers touch in the point

$$m = r^c + \frac{(\mu_r - r^c e)'\Sigma^{-1}(\mu_r - r^c e)}{e'\Sigma^{-1}\mu_r - r^c e'\Sigma^{-1}e},$$

$$\sigma^2(m) = \frac{(\mu_r - r^c e)'\Sigma^{-1}(\mu_r - r^c e)}{(e'\Sigma^{-1}\mu_r - r^c e'\Sigma^{-1}e)^2}.$$

If w is an optimal solution of Problem 1.14, then is $(w, 0)$ an optimal solution of Problem 1.20. Vice-versa, the optimal solution of Problem 1.20 has the form $(w, 0)$ and w is then an optimal solution of Problem 1.14.
If $e'\Sigma^{-1}\mu_r = r^c e'\Sigma^{-1}e$: $w^c = 1$, $e'w = 0$ and risks differ by $\frac{1}{e'\Sigma^{-1}e}$:

$$\frac{(m - r^c)^2}{(\mu_r - r^c e)'\Sigma^{-1}(\mu_r - r^c e)} + \frac{1}{e'\Sigma^{-1}e} = \frac{e'\Sigma^{-1}em^2 - 2e'\Sigma^{-1}\mu_r m + \mu_r'\Sigma^{-1}\mu_r}{e'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - [e'\Sigma^{-1}\mu_r]^2}.$$

Proof:
If $e'\Sigma^{-1}\mu_r \neq r^c e'\Sigma^{-1}e$: The solution of Problem 1.20 is $w^c = 0$ because we do not invest in cash with

$$c = \frac{1}{e'\Sigma^{-1}\mu_r - r^c e'\Sigma^{-1}e}, \ \lambda = -\frac{r^c}{e'\Sigma^{-1}\mu_r - r^c e'\Sigma^{-1}e}.$$

This results in the values of m and $\sigma^2(m)$ by substituting it in $m = r^c + c(\mu_r - r^c e)'\Sigma^{-1}(\mu_r - r^c e)$ and $\sigma^2(m) = \frac{(m-r^c)^2}{(\mu_r - r^c e)'\Sigma^{-1}(\mu_r - r^c e)}$. If m is substituted in the solutions of Problem 1.14 for λ and c, the values of both problems will be the same. So the portfolios are equivalent. To get the curvatures of the efficient frontiers we derive both risks twice and get for Problem 1.14 $\frac{\partial^2\sigma^2(m)}{\partial m^2} = \frac{2e'\Sigma^{-1}e}{e'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - [e'\Sigma^{-1}\mu_r]^2}$ and for Problem 1.20 $\frac{\partial^2\sigma^2(m)}{\partial m^2} = \frac{1}{(\mu_r - r^c e)'\Sigma^{-1}(\mu_r - r^c e)}$. We compare these values and get

$$\frac{2e'\Sigma^{-1}e}{e'\Sigma^{-1}e\mu_r'\Sigma^{-1}\mu_r - [e'\Sigma^{-1}\mu_r]^2} - \frac{1}{(\mu_r - r^c e)'\Sigma^{-1}(\mu_r - r^c e)} > 0.$$

Therefore the risk of Problem 1.20 is lower than in Problem 1.14 if $w^c \neq 0$.

If $e'\Sigma^{-1}\mu_r \neq r^c e'\Sigma^{-1}e$: The efficient frontiers of Problems 1.14 and 1.20 touch in $\hat{m} = r^c$ and they have identical curvatures.

□

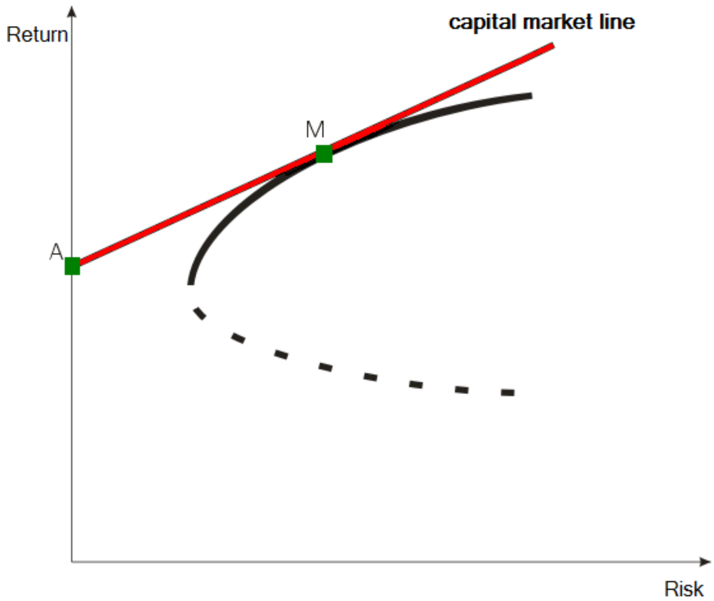

Figure 1.2: Capital market line with the (m, σ^2) combination M

In Figure 1.2 we see the *capital market line*, which corresponds to linear combinations of the riskless portfolio A and the risky portfolio M. In point M, no capital is invested in cash, so we have $w^c = 0$. The portfolio corresponding to M is called *market portfolio*. In the point A there is $m = r^c$, so nothing is invested in risky assets and $w^c = 1$.

The following Lemma shows, that it does not make sense to have more than one riskless asset.

Lemma 1.1: (**Arbitrage**)
Any portfolio with two or more riskless assets can realize any desired reward with zero risk.

Proof:
Consider two riskless assets with different returns r^c and r^d. For every desired return m we choose the weights

$$w^c = \frac{m - r^d}{r^c - r^d},\ w^d = 1 - w^c,$$

and invest nothing in other assets. The expected return of this portfolio is m and its variance is equal to 0.

$$w^c r^c + w^d r^d = \frac{m - r^d}{r^c - r^d} r^c + (1 - w^c) r^d = \frac{m r^c - r^d r^c + r^c r^d - (r^d)^2 - m r^d + (r^d)^2}{r^c - r^d} = m$$

□

1.2.3 Risky assets, riskless cash and guaranteed total loss

Consider again a portfolio with n risky assets, a riskless cash account, but include also an "asset" w^l with guaranteed total loss and $r^l = \mathbb{E}[r^l] = 0$. First it seems to be senseless to invest in an asset with guaranteed loss, but we will see, that it makes sense.

Assumptions: We have the same assumptions (A1) and (A2) as before but also a third one: we require positive cash return. We also assume that the reward of w^l is equal to 0.

- **(A1)** $\Sigma > 0$, the covariance matrix of risky assets is positive definite.
- **(A2)** $\mu_r \neq r^c e$.
- **(A3)** $r^c > 0$ (and $r^c > r^l$ is reasonable.)

We formulate the following optimization problem

$$\min_{w, w^c, w^l} \frac{1}{2} \begin{pmatrix} w \\ w^c \\ w^l \end{pmatrix}' \begin{pmatrix} \Sigma & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} w \\ w^c \\ w^l \end{pmatrix} = \frac{1}{2} w' \Sigma w \tag{1.28}$$

$$\begin{aligned} s.t.\ & e'w + w^c + w^l = 1, \\ & \mu_r' w + r^c w^c = m, \\ & w^c \geq 0,\ w^l \geq 0,\ w_i \geq 0,\ \forall i. \end{aligned}$$

The Lagrangian is

$$L(w, w^c, w^l, \lambda, c, \eta; m) = \frac{1}{2} w'\Sigma w - \lambda(e'w + w^c + w^l - 1) - c(\mu_r' w + r^c w^c - m) - \eta(w^l - 0). \tag{1.29}$$

By differentiating the Lagrangian we get the optimal solutions according to the following theorem.

Theorem 1.7:
Problem 1.28 has unique solutions $w, w^c, w^l, \lambda, c, \eta$, where η is the multiplier of the constraint $w^l \geq 0$.

- For $m > r^c$: The solution is identical to the solution of Problem 1.20. We have $w^l = 0$ and $\eta = -\lambda > 0$.
- For $m \leq r^c$: The optimal solution is to invest in a linear combination of the two riskless assets:

$$w = 0, w^c = \frac{m}{r^c}, w^l = 1 - \frac{m}{r^c}, \lambda = c = \eta = 0.$$

Proof:
We differentiate the Lagrangian 1.29 with respect to w and w^c and obtain Equations 1.22 and 1.23. If we differentiate the Lagrangian with respect to w^l we obtain the additional equation

$$\eta = -\lambda.$$

If $m > r^c$:
Then $c > 0$, which leads to $\lambda > 0$ and $-\eta > 0$. With the restriction $w^l \eta = 0$ follows $w^l = 0$ and we obtain Problem 1.20.

If $m \leq r^c$:
Then $c = 0$ which leads to $\lambda = -\eta = 0$ and $w = 0$.
From the budget equation we get with $w = 0$: $w^c + w^l = 1$ and it follows $w^l = 1 - w^c$.
From the reward equation we get with $w = 0$: $r^c w^c = m$ and it follows $w^c = \frac{m}{r^c}$ and $w^l = 1 - \frac{m}{r^c}$.

□

The trade-off version of Problem 1.28 is

$$\min_{w,w^c,w^l} \left\{ \frac{1}{2} w'\Sigma w - c(\mu_r w + r^c w^c) \right\} \tag{1.30}$$

$$s.t.\ e'w + w^c + w^l = 1,$$
$$w^c \geq 0,\ w^l \geq 0,\ w_i \geq 0,\ \forall i.$$

Theorem 1.8:
Problem 1.30 with $c > 0$ and Problem 1.28 with $m > r^c$ are equivalent if and only if $m = r^c + c(\mu_r - r^c e)'\Sigma^{-1}(\mu_r - r^c e)$.
Problem 1.30 with $c = 0$ has the same solutions as Problem 1.28 with $m \leq r^c$.
Problem 1.30 with $c < 0$ is unbounded and has no solution.

Proof:
The restrictions of both problems are nearly the same, only the reward condition in Problem 1.28 is included. The Lagrangian of Problem 1.30 is

$$L(w, w^c, w^l, \lambda, c, \eta; m) = \frac{1}{2} w'\Sigma w - c(\mu_r' w + r^c w^c) - \lambda(e'w + w^c + w^l - 1) - \eta(w^l - 0).$$

Differentiating according to w^c we get

$$-cr^c = \lambda. \tag{1.31}$$

By differentiating with respect to w^l we get

$$\eta = -\lambda. \tag{1.32}$$

Putting Equation 1.31 in Equation 1.32 we get $\eta = r^c c$.
If $c > 0$: $\eta > 0$ and $m > r^c$. If $c = 0$: $\eta = 0$ and $m \leq r^c$. If $c < 0$: $\eta < 0$ and there Problem 1.30 has no solution. For these results see Theorem 1.5 where the equation $m = r^c + c(\mu_r - r^c e)'\Sigma^{-1}(\mu_r - r^c e)$ holds.

□

At the beginning of this section we pointed out, that it is strange to invest in an asset with guaranteed loss. But why does it make sense?
The model describes that the investor wants to minimize the risk of earning exactly the prescribed reward. So when the variance is reduced, it is okay to loose money. It

minimizes the risk for $m < \hat{m}$ with $\hat{m}$ as the reward where the risk is minimal.
Another interpretation is, that the capital invested in w^l is just surplus capital: We get the desired reward without any risk and without that amount. This interpretation of the riskless but inefficient solutions becomes clear in Lemma 1.2.

Lemma 1.2:
Problem 1.28 is equivalent to the modification of Problem 1.20 where the budget equation $e'w + w^c = 1$ is replaced by $e'w + w^c \leq 1$. This means it is allowed to invest less than 100% of capital.

Proof:
Let $s \geq 0$ be a slack variable. We can rewrite the inequality $e'w+w^c \leq 1$ as $e'w+w^c+s = 1$. So Problem 1.20 is identical to Problem 1.28 with the slack variable $s = w^l$.

□

1.2.4 The influence of inequalities

Instead of specifying the restrictions by equalities, the budget equation and the reward equation, we consider the influence of inequalities. Let m be a lower bound for the desired reward. The following theorem is proved in Steinbach [1].

Theorem 1.9:
Consider the modification of Problem 1.14 where the reward equation is replaced by $\mu_r w \geq m$ and the modification of Problem 1.20 and 1.28 where $\mu_r w + r^c x^c = m$ is replaced by $\mu_r w + r^c x^c \geq m$. We define the modified problems with 1.14M, 1.20M and 1.28M. Then the following holds:

- Let $m \geq \hat{m}$ where $\hat{m}$ is the optimal reward. The solutions of the original Problem 1.14, 1.20 or 1.28 is also the unique solution of the modified Problem 1.14M, 1.20M or 1.28M, respectively.
- Let $m \leq \hat{m}$. The solution of Problem 1.14 or 1.20 with reward $\hat{m}$ is also the unique solution of the Problem 1.14M or 1.20M.
- Any solution of Problem 1.28 with $\mu(w, w^c, w^l) \in [m, r^c]$ with $m < r^c$ is a riskless solution of the Problem 1.28M. That is, any portfolio $(0, w^c, 1-w^c)$ with $w^c \in [\frac{m}{r^c}, 1]$ is optimal.

Other inequalities, like upper bounds on the assets, will restrict the range of feasible rewards and increase the risk. Markowitz handles this case by dummy assets and constraints $Aw = b, w \geq 0$, where $A = e'$ with $w \geq 0$ and $\mu(w) \geq m$ is called the standard case. He devised an algorithm to trace the segments of the efficient frontier, the so-called "Critical Line Algorithm", which we discuss in the next section.

1.2.5 Critical line algorithm

This section is based on the paper of A. and D. Niedermayer [2], pages 2-11.

First of all, we recall and define some variables we need in this section.
Consider a portfolio with n assets and nonnegative weights:

- Σ: $(n \times n)$ positive definite covariance matrix of asset returns.
- μ_r: n-dimensional vector of the assets' expected returns.
- w: n-dimensional vector of asset weights.
- $\mathbb{K}$: subset of $\{1, 2, \dots, n\}$ consisting of indexes of those assets on which a positive amount of money has been invested.
  k is the number of elements in $\mathbb{K}$, $k := |\mathbb{K}|$.
- Σ_k: $(k \times k)$ covariance matrix of the returns of non-zero weighted assets.
- μ_k: k-dimensional vector of expected returns of non-zero weighted assets.
- w_k: k-dimensional vector of the non-zero weighted asset weights.

We also need the definition of a *turning point*.

Definition 1.5: **Turning point**
A *turning point* is a point on the efficient frontier with an according portfolio if a point in its neighborhood (a next lower or higher point on the efficient frontier) with another according portfolio corresponding to another set of non-zero weighted assets.

We reformulate Problem 1.14 with the notation of the k-dimensional vectors.

$$\min_{w_k} \frac{1}{2} w_k' \Sigma w_k \tag{1.33}$$

$$\begin{aligned} s.t.\ e' w_k &= 1, \\ \mu_k' w_k &= m, \\ w_i &\geq 0,\ \forall i. \end{aligned}$$

The Lagrangian is

$$L(w, \lambda, c; m) = \frac{1}{2} w_k' \Sigma_k w_k - \lambda(e_k' w_k - 1) - c(\mu_k' w_k - m) \tag{1.34}$$

Now we can start to describe the algorithm:

- **Input:** Constraints as a system of linear inequalities.
- **1.Step:** Find the turning point with the lowest expected return value.
- **Next Steps:** Calculate the next higher turning points according to the portfolio with the next higher expected return.
- **Output:** Weights of the turning points on the efficient frontier. All other portfolios on the efficient frontier can be constructed as a linear combination of their neighboring turning points which are already found.

To move upwards from a turning point to the next one, c will increase:

$$c_1 < c_2 < c_3 \ldots.$$

For the starting solution we define for the minimal expected return μ^{min} the weight

$$w_1^{min} = 1,$$

and for the other expected returns μ^i the weights

$$w_1^i = 0.$$

When we move from a turning point to the higher one, either one non-zero weighted asset becomes zero, or one zero weighted asset becomes non-zero.

Consider these two cases separately.

One non-zero weighted asset becomes zero:

Let $c^{current}$ correspond to a turning point. Differentiating the Lagrangian 1.34 with respect to w_k and setting the differential equal to 0 we get

$$w_k = c\Sigma_k^{-1}\mu_k + \lambda\Sigma_k^{-1}e_k.$$

We can calculate the value $c^{(i)}$ and $\lambda^{(i)}$ of c and λ for the given subset $\mathbb{K}$ and an asset $i \in \mathbb{K}$ where the weight of asset i is zero:

$$0 = w_i = c^{(i)}(\Sigma_k^{-1}\mu)_i + \lambda^{(i)}(\Sigma_k^{-1}e)_i.$$

Solving this equation for $c^{(i)}$ leads to

$$c^{(i)} = \frac{(\Sigma_k^{-1}e_k)_i}{e_k'\Sigma_k^{-1}\mu_k(\Sigma_k^{-1}e_k)_i - e_k'\Sigma_k^{-1}e_k(\Sigma_k^{-1}e_k)_i}.$$

The next $c > c^{current}$ where an asset would leave the subset $\mathbb{K}$ is

$$c^{inside} = \min_i \left\{ c^{(i)} | c^{(i)} > c^{current} \right\}, \; i \in \mathbb{K}.$$

If we can not find a $c^{(i)} > c^{current}$ then there exists no solution for c^{inside}.

One zero weighted asset becomes non-zero:

In this case we have to redefine the subset $\mathbb{K}$ and include in $\mathbb{K}$ the index of asset i, which weight becomes non-zero:

$$\mathbb{K}_i = \mathbb{K} \cup \{i\},$$

where $i \notin \mathbb{K}$.
Analogously as before setting $w_i = 0$ we get for $c^{(i)}$

$$c^{(i)} = \frac{(\Sigma_{k_i}^{-1}e_{k_i})_i}{e_{k_i}'\Sigma_{k_i}^{-1}\mu_{k_i}(\Sigma_{k_i}^{-1}e_{k_i})_i - e_{k_i}'\Sigma_{k_i}^{-1}e_{k_i}(\Sigma_{k_i}^{-1}e_{k_i})_i}.$$

In order to find the next $c > c^{current}$ where the weight of a zero-weighted asset would become non-zero we define

$$c^{outside} = \min_i \left\{ c^{(i)} | c^{(i)} > c^{current} \right\}, \; \forall i \notin \mathbb{K}.$$

If we can not find a $c^{(i)} > c^{current}$ then there exists no solution for $c^{outside}$.

Finally we compare the values of c^{inside} and $c^{outside}$ to find out, which of the above cases occurs.

- If solutions exist for both c^{inside} and $c^{outside}$, the next turning point has a c defined as
$$c^{new} = min\left\{c^{inside}, c^{outside}\right\}.$$
- If a solution only occurs for c^{inside} the new c is set to $c^{new} = c^{inside}$ and if a solution only occurs for $c^{outside}$ the new c is set to $c^{new} = c^{outside}$.
- We replace $\mathbb{K}$ by $\mathbb{K} \setminus \{i\}$ if a non-zero weighted asset becomes zero or we replace $\mathbb{K}$ by $\mathbb{K}_i$ if a zero weighted asset becomes non-zero. We also replace $c^{current}$ by c^{new} depending on which case occurs .
- If no solution exist for c^{inside} or $c^{outside}$, we have reached the highest turning point and the algorithm terminates.

1.2.6 Downside risk

In this section we consider the distribution of returns, define the semi-variance and discuss optimization problems involving so-called *downside risk* measures. The disadvantage of the variance as risk measure is, that the positive and negative deviation from the mean is considered as equally risky. The most common downside risk measures are the *Value at Risk* and the *Conditional Value at Risk*.

Definition 1.6: Downside risk
For a function f of the random vector r with distribution P, the *downside risk* of order $q > 0$ with target $\tau \in \mathbb{R}$ is given as

$$\underline{\sigma}^q_\tau(f) := \mathbb{E}[|min(f(r) - \tau, 0)|^q] = \int_{\mathbb{R}^n} |min(f(r) - \tau, 0)|^q dP.$$

Thus the downside risk of order 1 with target τ is a partial moment of order 1. For $q = 1$ and $\tau = \mathbb{E}(f)$ we get the downside expected value and for $q = 2$ and $\tau = \mathbb{E}(f)$ we get the downside variance or semi-variance.

We are interested in quadratic downside risk of the portfolio with $f_{w,w^c}(r,r^c) = r'w+r^cw^c$ only. The standard risk is replaced by the downside risk $\underline{\sigma}_m^2$ with target $\tau = m = \mu(w)$.

Now we define the semi-variance matrix of a portfolio specified by its weights w:

$$\Sigma(w) := \int_{\mu_r+H(w,0)} (r-\mu_r)(r-\mu_r)'dP, \;\; w \neq 0,$$

where $H(w,0) := \{r \in \mathbb{R}^n : r'w < 0\}$ is the open half-space.

Now we substitute the objective functions of Problem 1.20 and 1.28 by the downside risk version and obtain the following problems.

We minimize downside risk $\underline{\sigma}_m^2(w,w^c)$ for risky assets and cash with fixed desired reward $\mu(w,w^c) = m$:

$$\min_{w,w^c} \frac{1}{2} \int_{\mathbb{R}^n} min(r'w + r^cw^c - m, 0)^2 dP \tag{1.35}$$

$$\begin{aligned} s.t. \;\; & e'w + w^c = 1, \\ & \mu_r'w + r^cw^c = m, \\ & w^c \geq 0, \;\; w_i \geq 0, \;\; \forall i. \end{aligned}$$

We minimize downside risk $\underline{\sigma}_m^2(w,w^c,w^l)$ for risky assets, cash and loss with fixed desired reward $\mu(w,w^c,w^l) = m$, where the asset with guaranteed loss has an expected return of zero:

$$\min_{w,w^c,w^l} \frac{1}{2} \int_{\mathbb{R}^n} min(r'w + r^cw^c - m, 0)^2 dP \tag{1.36}$$

$$\begin{aligned} s.t. \;\; & e'w + w^c + w^l = 1, \\ & \mu_r'w + r^cw^c = m, \\ & w^c \geq 0, \;\; w^l \geq 0, \;\; w_i \geq 0, \;\; \forall i. \end{aligned}$$

In general, closed solutions of Problem 1.35 and 1.36 cannot be found, because the downside risk is not linear. But we can discuss some important properties of the solutions and compare these problems with Problem 1.20 and 1.28. We make the same assumptions (A1), (A2) and (A3) as before. See also Steinbach [1], Lemma 4 and Lemma 8.

Lemma 1.3:

- In Problems 1.35 and 1.36 there exist always optimal solutions.
- The resulting downside risk is nonnegative and not greater than the optimal risk in Problem 1.20 and 1.28.
- The riskless solutions of Problems 1.35 and 1.20, or Problems 1.36 and 1.28, are identical and the solutions are not unique in general.

Proof: See Steinbach [1].

Theorem 1.10:
In Problem 1.35, choose optimal portfolios $(w_\pm, w_\pm^c)$ for $m_\pm := r^c \pm 1$, respectively. Then $(aw_\pm, aw_\pm^c - a + 1)$ is optimal for $m = r^c \pm a$, if $a \geq 0$. Moreover, $w_\pm \neq 0$ and $w_+ \neq w_-$.

Proof: See Steinbach [1].

Theorem 1.11:
There exist constants $c_\pm \in (0, 1)$ so that the optimal risk in Problem 1.35 is c_+ or c_- times the optimal risk of Problem 1.20 on the upper or lower branch of the reward.

Proof: See Steinbach [1].

Theorem 1.12:

- For the upper branch in Problem 1.36 holds the same statements as in Theorems 1.10 and 1.11.
- On the lower branch one has the unique riskless solution $(w, w^c, w^l) = (0, \frac{m}{r^c}, 1 - \frac{m}{r^c})$.

Proof: See Steinbach [1].

We have seen, that downside risk behaves similar as standard risk. The difference is, that the uniqueness is not guaranteed any more and the curvatures of upper and lower branches of the efficient frontier may differ. The reason for the similarity is that we fix the reward. The variance and semi-variance become identical if the return distribution is symmetric with respect to its expected value μ_r. In this case $\Sigma(w) = \frac{1}{2}\Sigma$ for all w and $c_+ = c_- = \frac{1}{2}$ holds.

1.3 Multi-period mean-variance model

In this section we use the previous results of the single-period model in developing multi-period analysis.

Consider a planning horizon of $T+1$ periods in discrete time $t = 0, \ldots, T+1$. At $t = 0$ the portfolio is allocated and at $t = 1, \ldots, T$ it is restructured, before the investor finally obtains his reward at time $T+1$. The portfolio vector at time t is denoted by $w_t \in \mathbb{R}^n$ and the reward vector at time t is denoted by $r_{t+1} = \left\{r_{t+1}^1, \ldots, r_{t+1}^n\right\} \in \mathbb{R}^n$ for $t = 0, \ldots, T$. Just before decision time t we have asset capitals $R_t := r_t w_{t-1}$. The decision at time t is taken after the observation of the realizations $r_1, \ldots, r_t$ but before the observation of $r_{t+1}, \ldots, r_{T+1}$ and leads finally to a policy $w = (w_0, \ldots, w_T)$.
We suppose, that the distribution of returns is given by a so-called *scenario tree* which we see in Figure 1.3. Each r_t has finitely many realizations r_j with probabilities $p_j > 0$, $j \in L_t$, where L_t denotes a level set of the tree. By $V := \cup_{t=0}^{T} L_t$ is denoted the set of all nodes, by $L := L_T$ the set of leaves, by $0 \in L_0$ the root, by $j \in L_t$ the current node, by $i = \pi(j) \in L_{t-1}$ the parent node and by $S(j) \subseteq L_{t+1}$ the set of child nodes.

The conditional expectation of the return is given as $\mu_T := \mathbb{E}[r_{T+1}|L_T]$ and its covariance matrix is given as

$$\Sigma_T := \mathbb{E}[(r_{T+1} - \mu_T)(r_{T+1} - \mu_T)'|L_T] = \mathbb{E}[r_{T+1} r'_{T+1}|L_T] - \mu_T \mu'_T,$$

with realizations μ_j, Σ_j on L_T.

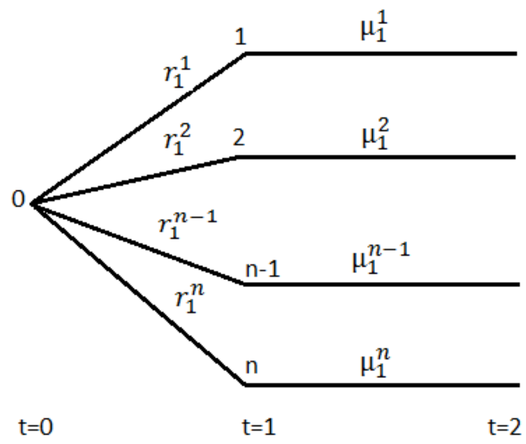

Figure 1.3: Scenario tree with t=2 periods

The discrete decision vector is a vector where the decision takes place at certain times $j \in V$ and is denoted by $w = (w_j)_{j \in V}$, $w_j \in \mathbb{R}^n$. The expectation of the total return R_{T+1} and the risk are defined as follows.

Definition 1.7: Expectation

$$\mu(w) := \mathbb{E}[R_{T+1}] = \mathbb{E}[r'_{T+1} w_T] = \mathbb{E}[\mathbb{E}[r'_{T+1}|L_T]]w_T = \mathbb{E}[\mu'_T]w_T = \sum_{j \in L} p_j \mu'_j w_j.$$

Definition 1.8: Risk

$$\sigma^2(w) := Var[R_{T+1}] = Var[r'_{T+1} w_T] = \mathbb{E}[(r'_{T+1} w_T - \mu'_T w_T)^2].$$

Lemma 1.4:
The risk is given by

$$\sigma^2(w) = \mathbb{E}[w'_T(\Sigma_T + \mu_T \mu'_T)w_T] - \mu(w)^2 = \sum_{j \in L} p_j w'_j(\Sigma_j + \mu_T \mu'_T)w_j - \mu(w)^2.$$

Proof:
By Definition 1.8, the risk is

$$\begin{aligned}\sigma^2(w) =&\mathbb{E}[r'_{T+1}w_T - \mu(w)^2]\\ =&\mathbb{E}[w'_T r_{T+1} r'_{T+1} w_T] - \mu(w)^2\\ =&\mathbb{E}[\mathbb{E}[w'_T r_{T+1} r'_{T+1} w_T | L_T]] - \mu(w)^2\\ =&\mathbb{E}[w'_T \mathbb{E}[r_{T+1} r_{T+1} | L_T] w_T] - \mu(w)^2\\ =&\mathbb{E}[w'_T (\Sigma_T + \mu_T \mu'_T) w_T] - \mu(w)^2\\ =&\sum_{j\in L} p_j w'_j (\Sigma_j + \mu_T \mu'_T) w_j - \mu(w)^2.\end{aligned}$$

□

Corollary 1.1:
The conditional reward and risk of the final period is $\mu_T(w_T) := \mu'_T w_T$ and $\sigma^2_T(w_T) := w'_T \Sigma_T w_T$ with realizations $\mu_j(w_j) := \mu'_j w_j$ and $\sigma^2_j(w_j) := w'_j \Sigma_j w_j$. The risk can be separated in a continuous and discrete part $\sigma^2(w) = \sigma^2_c(w) + \sigma^2_d(w)$ with

$$\begin{aligned}\sigma^2_c(w) :=&\mathbb{E}[\sigma^2_T(w_T)] = \sum_{j\in L} p_j w'_j \Sigma_j w_j,\\ \sigma^2_d(w) :=&\mathbb{E}[\mu_T(w_T)^2] - \mu(w)^2 = \sum_{j\in L} p_j \mu_j(w_j)^2 - \mu(w)^2.\end{aligned}$$

Proof:
By Lemma 1.4 we have

$$\begin{aligned}\sigma^2(w) =&\sum_{j\in L} p_j w'_j (\Sigma_j + \mu_T \mu'_T) w_j - \mu(w)^2\\ =&\sum_{j\in L} p_j w'_j \Sigma_j w_j + \sum_{j\in L} p_j w'_j \mu_T \mu'_T w_j - \mu(w)^2\\ =&\sigma^2_c(w) + \sigma^2_d(w).\end{aligned}$$

□

The continuous part σ_c^2 is the expectation of the conditional variance of R_{T+1} , which measures the average final-period risk. The discrete part σ_d^2 is the variance of the conditional expectation, which measures how well the individual scenario returns are balanced.

We would also be able to discuss several optimization problems with risky assets only, with a cash account or guaranteed loss added in the multi-period model, but we do not do this in this section. Instead we refer to the paper of Steinbach [1].

1.4 Conclusion

In our considerations in this chapter on the classical mean-variance optimization we have seen, that the single-period and multi-period models are similar in many aspects. It is possible to avoid overperformance when we allow to remove capital. There is zero risk at small desired rewards $m \leq r^c$, so that all the capital is invested in cash or removed. The problems of minimizing the variance versus minimizing the semi-variance or any other downside risk measures are equivalent. We consider the multi-period model with $w = (w_j, w_j^c)_{j \in V}$ and m_j as a fixed value for the reward in every node $j \in V$. The problem

$$\min_w \sigma_m^2(w) \tag{1.37}$$

$$\begin{aligned} s.t.\ & \mu(w) = m, \\ & e'w_j + w_j^c \leq m_j \ \forall j \in V, \end{aligned}$$

is equivalent to the downside risk problem

$$\min_w \underline{\sigma}_m^2(w) \tag{1.38}$$

$$\begin{aligned} s.t.\ & \mu(w) \geq m, \\ & e'w_j + w_j^c = m_j \ \forall j \in V. \end{aligned}$$

For $m > r^c$ one can not avoid overperformance completely, but Problem 1.37 still tends to minimize the semi-variance. The discrete part σ_d^2 approximates its downside version because of the existence of subtrees with zero risk. If m increases the quality of approximation becomes worse and the risk measures becomes a mixture of variance and semi-variance for large values of m.

2 Robust Optimization

2.1 Introduction

In this section we discuss robust optimization problems in general and take a look at portfolio optimization as an example in the subsection of objective robustness. We consider robust portfolio optimization problems in Chapter 3 in detail. This chapter is based on the book of Cornuejols and Tütüncü [4], Chapter 19 and on the book of Frank J. Fabozzi et al. [5], Chapter 10.

Mostly, inputs with real data are uncertain and optimization solvers are sensitive to small fluctuations in the input parameters. Reasons for uncertainty are estimation errors, uncertain inputs in the constraints or objective functions. We have to handle the uncertainty in our optimization problems. The oldest method is the sensitivity analysis, where we treat uncertainty after a solution is obtained. There exist also other methods dealing with uncertainty during the computation:

- stochastic programming,
- dynamic programming,
- robust optimization.

The fields of these methods overlap, but historically they have evolved independently of each other.

First we want to consider stochastic and dynamic programming shortly before we put our considerations on robust optimization.
In *stochastic programming* methods the uncertainty is represented by scenarios which are generated in advance and the objective function over all scenarios is optimized on average. The three most common types of problems are multi-period models, models with risk measures and chance-constrained models, see Frank J. Fabozzi et al. [5], Chapter 10.
Dynamic programming is used for multi-period models. The main idea is to solve the problem recursively. We separate a large problem in smaller ones for each possible stage

and start in the last stage going backward to get the optimal solution.

Our main aspect is the *robust optimization*. This is an attractive alternative to stochastic and dynamic programming because it is often difficult to obtain exact informations about the probabilistic distributions of the uncertain parameters. In the case of robust optimization we only make general assumptions on these distributions and thus have to work with problem formulations that are more tractable computationally. Uncertainty of the parameters is described in uncertainty sets where the possible values for the uncertain parameters are contained. These sets are based on statistical estimates and probabilistic guarantees on the solution. The problems are solved for the worst-case realization of the uncertain parameters. When they have special shapes the problem can be solved efficiently. We can distinguish between

- **Constraint robustness**:
  The constraints contain uncertain parameters, resulting at the uncertainty of solution feasibility.
- **Objective robustness**:
  Feasibility constraints are certain and the uncertainty affects the coefficients of the objective function and hence the optimality of the solutions.

In this chapter, we consider *relative robustness* and *adjustable-robust optimization*, too. But first of all we discuss the *uncertainty sets*.

2.2 Uncertainty sets

As we mentioned before, uncertainty sets can be formed by differences of opinions on future values of certain parameters and/or alternative estimates of parameters which are generated by statistical techniques from historical data. Let $s = (s_i)$ be the vector of uncertain parameters. Some types of uncertainty sets are as follows.

- A finite number of scenarios:
$$\mathbb{U} = \{s_1, s_2, \dots, s_k\}.$$
- The convex hull of a finite number of scenarios:
$$\mathbb{U} = conv(s_1, s_2, \dots, s_k).$$
- Interval description for each uncertain parameter:
$$\mathbb{U} = \{s = (s_i) : l_i \leq s_i \leq u_i,\ i = 1, \dots, n\}.$$

- Polytopic uncertainty sets with the estimator $\hat{s} = (\hat{s}_i)$ for the vector of uncertain parameters $s = (s_i)$ of dimension n and with a constant δ_i:

$$\mathbb{U} = \{s = (s_i) : |s_i - \hat{s}_i| \leq \delta_i,\ i = 1, \ldots, n\}.$$

- Ellipsoidal uncertainty sets with the estimator $\hat{s} = (\hat{s}_i)$ for the vector of uncertain parameters $s = (s_i)$, a constant vector $\delta = (\delta_i)$ of dimension n and a positive semidefinite matrix A:

$$\mathbb{U} = \left\{s = (s_i) : \sqrt{(s - \hat{s})' A^{-1} (s - \hat{s})} \leq \delta,\ i = 1, \ldots, n\right\}.$$

The constant δ can be interpreted as the aversion to the uncertainty. If δ is small then the investor has a low risk aversion.

The polytopic uncertainty set is also called a *box* as illustrated in Figure 2.1 for $n = 2$:

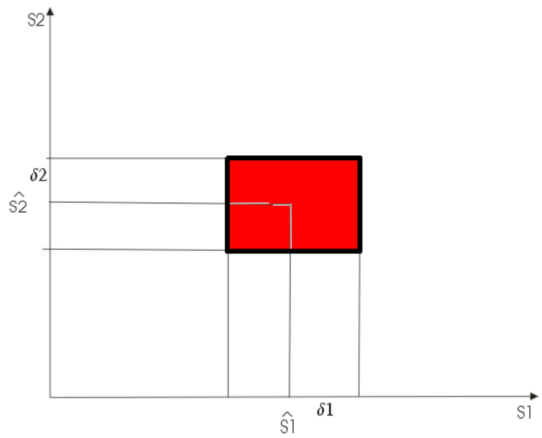

Figure 2.1: Polytopic uncertainty set for $n = 2$

For example, by using linear factor models in a portfolio optimization problem, the multivariate returns can be estimated by linear regression where the uncertainty sets become ellipsoidal sets. The matrix A affects the size and the shape of the uncertainty set which in turn can significantly affect the robustness of generated solutions. Another way to compute uncertainty sets is with the technique of bootstrapping and the use of average returns of historical data. This approach leads to polytopic uncertainty sets. Although polytopic uncertainty sets generally do not contain any second moment information about the distribution of the uncertain parameters like in the case of ellipsoidal uncertainty sets, some attractive computational properties of the original optimization problem can be inherited by this kind of robust counterpart. Optimization problems with ellipsoidal uncertainty sets are more difficult to solve than those with polytopic uncertainty sets. Polytopic uncertainty sets are discussed in Chapter 3. For ellipsoidal uncertainty sets see the paper of Sanyal et al. [7].

2.3 Models of robustness

2.3.1 Constraint robustness

Constraint robustness is one of the most important models in robust optimization. The uncertain parameters are in the constraints and we want to obtain solutions which are feasible for all possible uncertain parameters. An example are multi-period optimization problems where the uncertain solutions of the previous stages influence the decisions of the later stages and the decision variables have to satisfy some balance constraints. In order to find robust solutions we consider the following optimization problem:

$$\min_x f(x) \tag{2.1}$$

$$s.t.\ G(x,s) \in \mathbb{K},$$

where x are the decision variables and f is the certain objective function. G and $\mathbb{K}$ are certain structural elements of the constraints and s are the uncertain parameters. Let $\mathbb{U}$ be an uncertainty set with all possible values of parameters s. Then the constraint-robust optimization problem is given as follows:

$$\min_x f(x) \tag{2.2}$$

$$s.t.\ G(x,s) \in \mathbb{K},\ \forall s \in \mathbb{U}.$$

Thus the robust feasible set is the intersection of the feasible set $S(s) = \{x : G(x,s) \in \mathbb{K}\}$ for all $s \in \mathbb{U}$.

If we have an ellipsoidal feasible set, where $\hat{s}_i$ is the uncertain center of the ellipse corresponding to parameter s_i, then the robust feasible set is the red intersection in Figure 2.2.

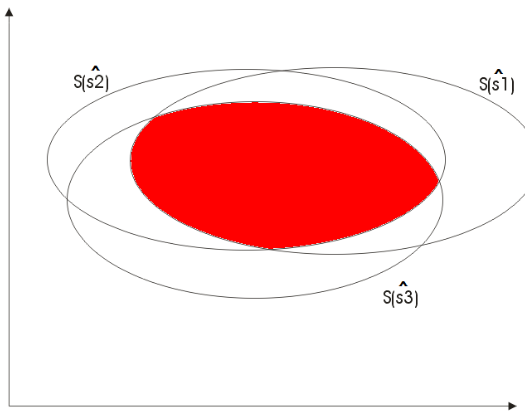

Figure 2.2: Robust feasible set

If there are uncertain parameters in the objective function and in the constraints, the optimization problem

$$\min_x f(x, s) \tag{2.3}$$
$$s.t.\ G(x, s) \in \mathbb{K}$$

can be written as

$$\min_{x,y} y \tag{2.4}$$

$$s.t.\ y - f(x, s) \geq 0,$$
$$G(x, s) \in \mathbb{K}.$$

In Problem 2.4 all uncertain parameters are in the constraints.

2.3.2 Objective robustness

Objective robustness is another important part of robust optimization. In this model we look for solutions which are close to the optimal solutions for all possible values of the uncertain parameters. Such solutions are difficult to obtain, especially when the uncertainty set is quite large. Thus an alternative aim of objective robustness is to optimize the worst case, which means finding a solution which is optimal for the worst possible realization of the uncertain parameters. Consider the following optimization problem of objective robustness:

$$\begin{aligned} &\min_x f(x,s) \\ &s.t.\ x \in F, \end{aligned} \tag{2.5}$$

where F is a certain feasible set and the objective function f depends on the uncertain parameter s. Again, let $\mathbb{U}$ be the uncertainty set. For handling with objective robustness we reformulate Problem 2.5 as a constraint-robust optimization problem. Then we consider the worst case and get the objective-robust optimization problem:

$$\min_{x \in F} \max_{s \in \mathbb{U}} f(x,s). \tag{2.6}$$

The reformulation in the end of the previous subsection, Problem 2.4, and Problem 2.6 lead to two different classes of optimization problems, called *semi-infinite* and *min-max optimization problems*.

As an example, we consider a portfolio optimization problem with n risky assets. We want to maximize the reward subject to the budget equation. Let w be the vector of asset weights as in Chapter 1 and let $\mu_r := \mathbb{E}[r]$ be the expectation of asset returns. The vector μ_r is the vector of uncertain parameters because we do not know the expected returns at the moment of portfolio construction. So we discuss the following problem where the uncertainty is in the objective function:

$$\begin{aligned} &\max_w \mu_r' w \\ &s.t.\ e'w = 1, \\ &\quad w_i \geq 0,\ \forall i. \end{aligned} \tag{2.7}$$

In Problem 2.7 we maximize the expected portfolio return depending on the uncertain expected asset returns. We model the uncertainty in terms of uncertainty sets and maximize the expected portfolio return for the worst realization of expected asset returns in the uncertainty set.

We define the uncertainty set with the estimated expected returns $\hat{\mu}_r = (\hat{\mu}_i)_{1 \leq i \leq n}$ as follows:

$$\mathbb{U}(\hat{\mu}_r) = \left\{ \mu_r = (\mu_i) : (\mu_r - \hat{\mu}_r)' \Sigma^{-1} (\mu_r - \hat{\mu}_r) \leq \delta^2, \ i = 1, \ldots, n \right\}, \tag{2.8}$$

with the covariance matrix Σ. This uncertainty set means that the deviation of the expected returns from the realized returns scaled with the inverse covariance matrix is bounded by δ, the limit at which the investor wants to be protected from a larger deviation from the optimum.

The robust counterpart of Problem 2.7 is

$$\max_{w} \min_{\mu_r \in \mathbb{U}(\hat{\mu}_r)} \mu_r' w \tag{2.9}$$

$$\begin{aligned} s.t.\ & e'w = 1, \\ & w_i \geq 0, \ \forall i. \end{aligned}$$

Problem 2.9 is hard to be solved by standard software because typically such a software cannot solve a two stage optimization problem. So we rewrite Problem 2.9 by using duality. We consider the first stage of Problem 2.9. We fix w in Problem 2.9 and optimize for the worst case over the uncertain parameter μ_r:

$$\min_{\mu_r} \mu_r' w. \tag{2.10}$$

$$s.t. \left\| \Sigma^{-\frac{1}{2}} (\mu_r - \hat{\mu}_r) \right\| \leq \delta,$$

with the Euclidean norm $\|x\| = \sqrt{x_1^2 + \ldots + x_n^2}$ for a n-dimensional vector x and $(\Sigma^{-\frac{1}{2}})' \Sigma^{-\frac{1}{2}} = \Sigma^{-1}$.
To find the dual problem of the conic Problem 2.10 we consider the following primal and dual conic problems:

- Primal problem:

$$\min_{x} c'x \tag{2.11}$$

$$s.t. \ \|C_i x + d_i\| \leq c_i' x + e_i, \ \forall i.$$

- Dual problem:

$$\max_{u,v} \left\{ -\sum_{i=1}^{n} u_i' d_i + v_i e_i \right\} \tag{2.12}$$

$$s.t.\ \sum_{i=1}^{n} u_i C_i + v_i c_i = c,$$
$$\|u_i\| \leq v_i,\ \forall i.$$

For conic optimization and duality theory see Fabozzi et al. [5], Chapter 9.

We see two examples for conic optimization later in this chapter.

With $x = \mu_r, c = w, c_i = 0, v_i = \lambda, e_i = \delta, C_i = \Sigma^{-\frac{1}{2}}, d_i = -\Sigma^{-\frac{1}{2}}\hat{\mu}_r$ and $u_i = u$ the dual problem of Problem 2.10 is

$$\max_{u,\lambda} \left\{ -(-u'\Sigma^{-\frac{1}{2}}\hat{\mu}_r) - \delta\lambda \right\} \tag{2.13}$$

$$s.t.\ \Sigma^{-\frac{1}{2}}u + 0\lambda = w,$$
$$\|u\| \leq \lambda.$$

The first constraint of Problem 2.13 leads to the equation

$$u = \Sigma^{\frac{1}{2}}w. \tag{2.14}$$

Using Equation 2.14 we get

$$\max_{\lambda} \left\{ w'\hat{\mu}_r - \delta\lambda \right\} \tag{2.15}$$
$$s.t.\ \left\| \Sigma^{\frac{1}{2}}w \right\| \leq \lambda.$$

By using duality, Problem 2.15 is equivalent to Problem 2.10 and the worst case leads to the equation

$$w'\hat{\mu}_r - \delta \left\| \Sigma^{\frac{1}{2}}w \right\| = w'\hat{\mu}_r - \delta\sqrt{w'\Sigma w}, \tag{2.16}$$

for any fixed set of weights w. Equation 2.16 does not depend on the uncertain parameter μ_r. Now we replace the objective function of Problem 2.9 by Equation 2.16 and optimize over w:

$$\max_w \left\{ w'\hat{\mu}_r - \delta\sqrt{w'\Sigma w} \right\} \tag{2.17}$$

$$s.t.\ e'w = 1,$$
$$w_i \geq 0,\ \forall i.$$

Problem 2.17 can be solved by a nonlinear optimization software.

2.3.3 Relative robustness

The models above are not consistent with the risk tolerances of many decision-makers. We need the worst-case in a relative context to the best possible solution under each scenario. Therefore we need the relative robustness. Consider Problem 2.3 again. Now we want the relative robust optimization problem.
For a fixed $s \in \mathbb{U}$, let $t^*(s)$ be the optimal value function

$$t^*(s) = \min_x f(x, s)$$

$$s.t.\ x \in F(s),$$

where $F(s)$ is the set of feasible solutions of Problem 2.3. Let $x^*(s)$ be the optimal solution map

$$x^*(s) = argmin_x f(x, s)$$

$$s.t.\ x \in F(s).$$

We define a measure of *regret* which is associated with a decision after the uncertainty is resolved.

$$r(x, s) = f(x, s) - f(x^*(s), s) = f(x, s) - t^*(s) \geq 0.$$

For a fixed x in the feasible set we maximize the regret function:

$$R(x) := \max_{s\in\mathbb{U}} r(x, s) = \max_{s\in\mathbb{U}} \{f(x, s) - t^*(s)\}.$$

The relative robustness model is the minimum of the maximized regret:

$$\min_{x \in F(s)} \max_{s \in \mathbb{U}} \{f(x,s) - t^*(s)\}. \tag{2.18}$$

Problem 2.18 contains a three-level optimization problem because $t^*(s)$ involves an optimization problem itself and is hard to analyze. So relative robustness is more difficult to solve than the two-stage optimization Problem 2.9.
There is an easier way to formulate Problem 2.18. By limiting the maximum regret to M we get the following problem where we have to find an x which satisfies $G(x) \in \mathbb{K}$ such that

$$f(x,s) - t^*(s) \leq M, \ \forall s \in \mathbb{U}. \tag{2.19}$$

As another possibility of relative robustness we consider the proximity of our chosen solution to the optimal solution set. We need the *distance* between x and the set of optimal solutions for a fixed s.

$$d(x,s) = \inf_{x^* \in x^*(s)} \|x - x^*\|.$$

Now we consider the maximum distance of a solution x and the optimal solution $x^*(s)$ for $s \in \mathbb{U}$:

$$D(x) := \max_{s \in \mathbb{U}} d(x,s) = \max_{s \in \mathbb{U}} \inf_{x^* \in x^*(s)} \|x - x^*\|.$$

We look for a solution x minimizing the above maximum:

$$\min_{x \in F(s)} \max_{s \in \mathbb{U}} d(x,s). \tag{2.20}$$

This model is attractive when we have time to revise our decision variables x, if s is revealed. It can also be used for multi-period problems, where it can be time consuming to revise the decisions from one period to another. Particular examples are portfolio rebalancing problems with transaction costs. These are problems where it is allowed to rebalance the portfolio from one period to another while paying transaction costs. We will see such a multi-period portfolio optimization problem with transaction costs later in Chapter 3.

2.3.4 Adjustable robust optimization

In multi-period models some of the uncertain parameters are often revealed during the decision process. Adjustable robust optimization allows recourse actions, so that solutions which are not optimal can be corrected in later stages.

Now we consider a two-stage linear optimization problem with x_1 being the decision variable of the first stage and x_2 being the decision variable of the second stage. The second variable x_2 can be chosen after the uncertain parameters A_1, A_2 and b of the following problem are realized and does not appear in the objective function:

$$\min_{x_1,x_2} \left\{ c'x_1 : A_1x_1 + A_2x_2 \leq b \right\}. \tag{2.21}$$

Let $\mathbb{U}$ be the uncertainty set for the uncertain parameters A_1, A_2 and b. Before the uncertain parameters are observed, both sets of variables must be chosen and cannot depend on these parameters. The standard robust counterpart can be formulated as follows:

$$\min_{x_1} \left\{ c'x_1 : \exists x_2 \forall (A_1, A_2, b) \in \mathbb{U} : A_1x_1 + A_2x_2 \leq b \right\}, \tag{2.22}$$

or equivalently

$$\min_{x_1,x_2} \left\{ c'x_1 : A_1x_1 + A_2x_2 \leq b,\ \forall (A_1, A_2, b) \in \mathbb{U} \right\}. \tag{2.23}$$

In adjustable robust optimization problems the decision variable of the second stage x_2 depends on the uncertain parameters A_1, A_2 and b and the problem can be written as follows:

$$\min_{x_1} \left\{ c'x_1 : \forall (A_1, A_2, b) \in \mathbb{U}, \exists x_2 \equiv x_2(A_1, A_2, b) : A_1x_1 + A_2x_2 \leq b \right\}. \tag{2.24}$$

The feasible set of Problem 2.24 is larger than the feasible set of Problem 2.22. If the robust counterpart is unnecessarily conservative, adjustable robust optimization problems are useful. Although these problems are flexible in modeling, the formulations are more difficult. Another disadvantage is that the feasible sets of the second-period decisions depend on the uncertain parameters and pn the decision variable of the first stage. To chance this for a better we can consider simpler assumptions on the feasible set and the dependence structure.

2.4 Strategies

In this section we discuss some techniques to solve robust optimization problems. The strategies are based on reformulating the problems such that no uncertainty occurs. We want that the new optimization problem is not much bigger than the original one and that it can be solved efficiently.

2.4.1 Sampling

Sampling is one of the simplest strategies to find the solution of a robust optimization problem. We sample several scenarios for the uncertain parameters from a set with possible values. It can be done with or without distributional assumptions and we get a robust optimization problem with a finite uncertainty set.
If the uncertainty appears in the constraint we copy each such constraint of each scenario. If uncertain parameters are in the objective function we can handle it similarly. Consider Problem 2.3 again.
For the uncertainty set $\mathbb{U} = \{s_1, s_2, \ldots, s_k\}$ we put all uncertain parameters in the constraints and obtain the robust formulation:

$$\min_{x,y} y \tag{2.25}$$

$$\begin{aligned} s.t.\ y - f(x, s_i) \geq 0,\ i = 1, \ldots, k, \\ G(x, s_i) \in \mathbb{K},\ i = 1, \ldots, k. \end{aligned}$$

Since the uncertainty set of Problem 2.25 is finite and the new constraints have the same structural properties as the constraints of the non-robust original problem, Problem 2.25 is not more difficult than Problem 2.3.

2.4.2 Conic optimization

In this section we replace the finite uncertainty sets by continuous sets like intervals or ellipsoids. There are infinitely many constraints but finitely many variables, so this leads to a semi-infinite optimization problem. It is possible to formulate semi-infinite optimization problems with finite sets of conic constraints. A conic optimization problem has the form:

$$\min_{x} c'x \tag{2.26}$$

$$s.t.\ Ax - b \in C,$$

with a closed convex cone C.

A special cone is the so-called *second-order cone*:

$$C_2 = \left\{ x \in \mathbb{R}^n : x_1 \geq \sqrt{\sum_{i=2}^{n} x_i^2} \right\}. \tag{2.27}$$

The cone C_2 leads to a second-order cone programming problem which is a special case of semidefinite programming. A general formulation of semidefinite problems is:

$$\min_x \langle C, X \rangle \tag{2.28}$$

$$\begin{aligned} s.t.\ & \left\langle A^{(ij)}, X \right\rangle = b_{ij},\ i,j = 1,\dots,n, \\ & X \succeq 0, \end{aligned}$$

with a matrix $A^{(ij)},\ i,j = 1,\dots,n$ and $\langle C, X \rangle$ as the trace of the matrix product CX of two symmetric matrices C and X, which is equal to the sum of the diagonal elements of the product CX.

Semidefinite programming is computationally more intensive than second-order cone programming, so if it is possible, it is better to formulate a robust problem as a second-order cone programming problem. See also Fabozzi et al. [5], Chapter 9.

Two examples with constraint robustness of conic optimization follow:

- Linear constraint optimization problem:

  $$\min_x c'x \tag{2.29}$$

  $$s.t.\ a'x + b \leq 0,\ \forall a, b \in \mathbb{U},$$

  with the ellipsoidal uncertainty set for the uncertain parameters a and b:

  $$\mathbb{U} = \left\{ [a;b] = [a^0;b^0] + \sum_{j=1}^{k} u_j [a^j;b^j],\ \|u\| \leq 1 \right\},$$

  with the vectors a and b concatenated to a vector $[a;b]$.

  This problem is equivalent to the second-order cone programming problem:

$$\min_{x,t} c'x \tag{2.30}$$

$$\begin{aligned} s.t.\ a'x + b_j = t_j,\ j = 0, \ldots, k, \\ (t_0, t_1, \ldots, t_k) \in C_2, \end{aligned}$$

where C_2 is the second-order cone 2.27.

See Cornuejols and Tütüncü. [4], Chapter 9.

- Quadratically constrained optimization problem:

$$\min_{x} c'x \tag{2.31}$$

$$s.t.\ \ -x'(A'A)x + 2b'x + s \geq 0,\ \forall A, b, s \in \mathbb{U}$$

with the ellipsoidal uncertainty set for the uncertain parameters A, b and s

$$\mathbb{U} = \left\{ [A; b; s] = [A^0; b^0; s^0] + \sum_{j=1}^{k} u_j [A^j; b^j; s^j],\ \|u\| \leq 1 \right\}.$$

This problem is equivalent to the semidefinite programming problem:

$$\min_{x,t^0,\ldots,t^k,v,\lambda} c'x \tag{2.32}$$

$$\begin{aligned} s.t.\ & A^j x = t_j,\ j = 0, \ldots, k, \\ & (b^j)'x = v^j,\ j = 0, \ldots, k, \\ & \lambda \geq 0, \\ & \begin{pmatrix} s^0 + 2v^0 - \lambda & [v^1 + \frac{1}{2}s^1 \cdots v^k + \frac{1}{2}s^k] & (t^0)' \\ v^1 + \frac{1}{2}s^1 & & (t^1)' \\ \vdots & \lambda I & \vdots \\ v^k + \frac{1}{2}s^k & & (t^k)' \\ t^0 & [t^1 \cdots t^k] & I \end{pmatrix} \succeq 0, \end{aligned}$$

where $A \succeq 0$ means that the matrix A is positive semidefinite.

See Cornuejols and Tütüncü. [4], Chapter 9.

2.5 Conclusion

In this chapter we got to know how to handle uncertain input parameters. We have seen some special cases of uncertainty sets:

- A finite number of scenarios,
- the convex hull of a finite number of scenarios,
- interval description for each uncertain parameter,
- polytopic uncertainty sets and
- ellipsoidal uncertainty sets.

We also got to know relative robustness which is very useful when we measure the performance relative to ones peers. We minimize the maximized regret function leading to a more difficult problem than usual robust formulations.
Finally we discussed strategies for dealing with uncertainty in the optimization problems:

- Sampling and
- conic optimization.

In the next chapter, we see, that we need the resampling technique, the second-order cone programming and the semidefinite programming in the case of robust portfolio optimization.

3 Robust portfolio optimization

This chapter is based on the book of Cornuejols and Tütüncü. [4], Chapter 20 and on the book of Fabozzi et al. [5], Chapter 12. For the resampling technique see the master thesis of Jiao [6], too.

The classical mean-variance optimization by Markowitz which we discussed in the first chapter, is very sensitive to the uncertainty of the inputs. Especially in finance future values are often used in the optimization problems and have to be estimated or forecasted. There can occur errors at the estimation and modeling, and we have to make the problem robust against these risks. The computational overhead of robust optimization problems is minimal.

First we consider briefly the technique of resampling a portfolio, then we discuss portfolio allocation with several robustified variations of the mean variance optimization problem. We consider uncertainty in the expected values and the covariance matrix of returns and the corresponding robust optimization problems.

We discuss an example of relative robust portfolios and finally we consider multi-period models with cash, sales and purchases.

3.1 Portfolio resampling technique

One way to deal with estimation errors is by resampling the estimated inputs.

First we consider the original estimates for the expected return vector $\hat{\mu}$ and the covariance matrix $\hat{\Sigma}$ and solve the global minimum variance portfolio problem and the maximum return portfolio problem. The standard deviation of the global minimum variance portfolio is denoted with σ_{GMV} and the deviation of the maximum return portfolio is denoted with σ_{MR}. We assume that the inequality $\sigma_{GMV} < \sigma_{MR}$ holds. We partition the interval $[\sigma_{GMV}, \sigma_{MR}]$ in m points so that $\sigma_{GMV} = \sigma_1 < \ldots < \sigma_m = \sigma_{MR}$. We solve the maximum return portfolio problem for each standard deviation $\sigma_i, 1 \leq i \leq m$. Denote by $w_1, \ldots, w_m$ the corresponding portfolio weights. Now we perform portfolio resampling as follows:

- **Step 1:** Compute t random samples from the multivariate distribution $N(\hat{\mu}, \hat{\Sigma})$ where t can be seen as a control parameter which reflects the degree of uncertainty in the inputs. We use these samples to estimate a new expected return vector $\hat{\mu}_i$ and a new covariance matrix $\hat{\Sigma}_i$. The details of this procedure will be described in the next chapter, in Section 4.1.1.

- **Step 2:** With these new estimates $\hat{\mu}_i$ and $\hat{\Sigma}_i$ we solve the corresponding global minimum variance and maximum return portfolio problems and get $\sigma_{GMV,i}, \sigma_{MR,i}$. As before, partition the interval $[\sigma_{GMV,i}, \sigma_{MR,i}]$ in m points and solve the corresponding maximum return portfolios for each standard deviation and get $w_{1,i}, \ldots, w_{m,i}$.
- **Step 3:** Compute an efficient frontier with the results of step 2.
- **Step 4:** Repeat Step 1 and Step 2 N times with a large N, for example 100 or 500.

If the resampling is finished for each point in the partition we calculate the resampled weights for a portfolio of rank m as the average:

$$w_m^{rs} = \frac{1}{N} \sum_{i=1}^{N} w_{m,i},$$

where w_m^{rs} denotes the resampled weight and $w_{m,i}$ stands for the weight of the mth portfolio of the frontier for the ith resampling.
Now consider the efficient frontier with the original inputs $\hat{\mu}$ and $\hat{\Sigma}$ and with the resampled portfolio weights. The resampled efficient frontier is below the original one. The reason is, that the weights $w_{1,i}, \ldots, w_{m,i}$ are efficient relative to the estimates $\hat{\mu}_i$ and $\hat{\Sigma}_i$ but inefficient relative to the estimates $\hat{\mu}$ and $\hat{\Sigma}$. The effect of estimation error is incorporated into determination of the resampled weights by resampling and reestimation.
From the simulated data, we can compute a distribution of the portfolio weights. For example, a large standard deviation of the portfolio weights means that the original portfolio weights are not very precise if the size of random samples t is small. With a test statistic we can check if two portfolios are statistically different or not. Denote by Σ_{rs}^{-1} the inverse covariance matrix of the resampled portfolio weights. Two portfolios with two different portfolio weights w_1 and w_2 are statistically equivalent if the following inequality holds:

$$d_1(w_1, w_2) = (w_1 - w_2)'\Sigma_{rs}^{-1}(w_1 - w_2) \leq C$$

with a constant C. See also the master thesis of Jiao [6], Chapter 3.

3.2 Robust portfolio allocation

Another way to cope with uncertain inputs is to consider them in the optimization process directly.
First we discuss the mean-variance optimization problem, where the uncertainty occurs in the expected returns and then we consider the problem with uncertainty in the covariance matrix, too.

3.2.1 Uncertainty in the expected returns

We assume that the uncertainty occurs in the expected returns and denote the uncertainty set by $\mathbb{U}(\hat{\mu}_r)$. The robust formulation of the mean-variance optimization problem is

$$\max_{w} \min_{\mu_r \in \mathbb{U}(\hat{\mu}_r)} \left\{ \mu_r' w - \frac{1}{2} w' \Sigma w \right\} \tag{3.1}$$

$$\begin{aligned} s.t.\ e'w &= 1, \\ w_i &\geq 0,\ \forall i. \end{aligned}$$

The objective function can be interpreted as maximization of the expected return in the worst case, i.e., maximization of the smallest expected return. In Problem 3.1 the inner optimization problem has to be solved first while the vector of weights is fixed. In this case we compute the worst case expected return over the uncertainty set. We want to solve Problem 3.1 with two special uncertainty sets.

- The polytopic uncertainty set:

$$\mathbb{U}(\hat{\mu}_r) = \{\mu_r = (\mu_i) : |\mu_i - \hat{\mu}_i| \leq \delta_i,\ i = 1, \ldots, n\}, \tag{3.2}$$

  where $\hat{\mu}_r = (\hat{\mu}_i)_{1 \leq i \leq n}$ is the estimation of μ_r.
  In uncertainty set 3.2 we assume that the realized expected return of asset i does not differ more than a small account δ_i from the estimated expected return of this asset, $\forall i,\ 1 \leq i \leq n$.

- The ellipsoidal uncertainty set:

$$\mathbb{U}(\hat{\mu}_r) = \left\{\mu_r = (\mu_i) : (\mu_r - \hat{\mu}_r)' \Sigma_\mu^{-1} (\mu_r - \hat{\mu}_r) \leq \delta^2,\ i = 1, \ldots, n\right\}, \tag{3.3}$$

where $\hat{\mu}_r = (\hat{\mu}_i)_{1 \leq i \leq n}$ is the estimation of μ_r and with a constant δ.
Uncertainty set 3.3 can be interpreted as an n-dimensional confidence region for the parameter vector μ_r with Σ_μ as the covariance matrix of the errors in the estimation of the expected returns. It means that the investor is protected in the case that the total scaled deviation of the realized average returns from the estimated returns is smaller than δ.

First we consider the polytopic uncertainty set 3.2.
With uncertainty set 3.2 for $\mu_r = (\mu_1, \ldots, \mu_n)$ the robust optimization problem for the worst case of μ_r can be formulated as

$$\max_w \left\{\hat{\mu}_r' w - \delta' w - \frac{1}{2} w' \Sigma w\right\} \tag{3.4}$$

$$s.t.\ e'w = 1,$$
$$w_i \geq 0,\ \forall i.$$

The worst case of the expected return is $\hat{\mu}_r - \delta$ with $\delta = (\delta_i)_{1 \leq i \leq n}$ and in this case we want to loose the smallest possible amount. The computational complexity of Problem 3.4 is the same as that of the non-robust mean-variance formulation.
The solution of this problem is given as

$$w^* = \Sigma^{-1}(\hat{\mu}_r - \delta - \lambda e) \tag{3.5}$$

$$\lambda^* = \frac{e'\Sigma(\hat{\mu}_r - \delta) - 1}{e'\Sigma^{-1}e}$$

as we saw in Section 1.2.1.

Now we consider the ellipsoidal uncertainty set 3.3.
Again, we consider the worst case of the estimates for the expected returns in Problem 3.1. First we solve the inner optimization problem and compute the worst case of μ_r for a fixed vector of weights w:

$$\min_{\mu_r} \left\{ \mu_r' w - \frac{1}{2} w'\Sigma w \right\} \tag{3.6}$$

$$s.t.\ (\mu_r - \hat{\mu}_r)'\Sigma_\mu^{-1}(\mu_r - \hat{\mu}_r) \leq \delta^2.$$

The Lagrangian is

$$L(\mu_r; c) = \mu_r' w - \frac{1}{2} w'\Sigma w - c(\delta^2 - (\mu_r - \hat{\mu}_r)'\Sigma_\mu^{-1}(\mu_r - \hat{\mu}_r)). \tag{3.7}$$

By differentiating Equation 3.7 with respect to μ_r, we get the first-order condition

$$\frac{\partial L}{\partial \mu_r} = w + 2c\Sigma_\mu^{-1}(\mu_r - \hat{\mu}_r) = 0$$

and obtain

$$\mu_r^* = \hat{\mu}_r - \frac{1}{2c}\Sigma_\mu w \tag{3.8}$$

as a solution. We obtain the optimal value c^* by plugging Expression 3.8 in Equation 3.7 and get the Lagrangian:

$$L(\mu_r^*; c) = (\hat{\mu}_r - \frac{1}{2c}\Sigma_\mu w)' w - \frac{1}{2} w'\Sigma w - c[\delta^2 - (\hat{\mu}_r - \frac{1}{2c}\Sigma_\mu w - \hat{\mu}_r)'\Sigma_\mu^{-1}(\hat{\mu}_r - \frac{1}{2c}\Sigma_\mu w - \hat{\mu}_r)]$$

$$L(\mu_r^*; c) = \hat{\mu}_r' w - \frac{1}{2} w'\Sigma w - \frac{1}{4c} w'\Sigma_\mu w - c\delta^2. \tag{3.9}$$

Differentiating the Lagrangian 3.9 with respect to c and putting it equal to zero, we get the first order condition:

$$\frac{\partial L}{\partial c} = \frac{1}{4c^2} w'\Sigma_\mu w - \delta^2 = 0.$$

The optimal value c^* is $c^* = \frac{1}{2\delta}\sqrt{w'\Sigma_\mu w}$ and hence $\mu_r^* = \hat{\mu}_r - \frac{\delta \Sigma_\mu w}{\sqrt{w'\Sigma_\mu w}}$.
By substituting the optimal value μ_r^* of the inner optimization problem in 3.1 we get the outer optimization problem as follows:

$$\max_{w} \left\{ \hat{\mu}_r' w - \frac{1}{2} w' \Sigma w - \delta \sqrt{w' \Sigma_\mu w} \right\} \tag{3.10}$$

$$\begin{aligned} s.t.\ & e'w = 1, \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

The term $\delta \sqrt{w' \Sigma_\mu w}$ represents the penalty for estimation risk where δ is the degree of the investor's aversion to estimation risk.

The Lagrangian of Problem 3.10 is

$$L(\mu_r; \lambda) = \hat{\mu}_r' w - \frac{1}{2} w' \Sigma w - \delta \sqrt{w' \Sigma_\mu w} - \lambda(e'w - 1) \tag{3.11}$$

Differentiating with respect to w, we get the first order condition:

$$\frac{\partial L}{\partial w} = \hat{\mu}_r - w' \Sigma - \frac{\delta w' \Sigma_\mu}{\sqrt{w' \Sigma_\mu w}} - \lambda e = 0.$$

It is difficult to compute the optimal value w^* from this Lagrangian, but we can solve Problem 3.10 numerically.

Finally a few words about the estimation of Σ_μ. There exist several methods to estimate Σ_μ, for example the Bayesian statistics and regression-based methods. Some effective techniques include least square regression models, the James-Stein estimator and the Black-Litterman model. See Fabozzi et al. [5], Chapter 12.

3.2.2 Uncertainty in covariance matrix of returns

In this section we consider the uncertainty in the estimation of the covariance matrix of returns, too. Notice that in comparison to errors in the estimates of expected values, the optimization is less sensitive against errors in the estimates of the covariance matrix of returns. Denote by $\mathbb{U}(\hat{\mu}_r)$ and $\mathbb{U}(\Sigma)$ the uncertainty sets of the expected returns and the covariance matrix, respectively. The robustified mean variance optimization problem can be written as

$$\max_{w} \left\{ \min_{\mu_r \in \mathbb{U}(\hat{\mu}_r)} \left\{ \mu_r' w \right\} - \frac{1}{2} \max_{\Sigma \in \mathbb{U}(\Sigma)} \left\{ w' \Sigma w \right\} \right\} \tag{3.12}$$

$$\begin{aligned} s.t.\ & e'w = 1, \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

We consider uncertainty set 3.2 for the expected returns and a confidence interval for the covariance matrix:

$$\mathbb{U}(\Sigma) = \left\{ \Sigma = (\Sigma_{ij}) : \underline{\Sigma}_{ij} \leq \Sigma_{ij} \leq \overline{\Sigma}_{ij},\ i,j = 1,\ldots,n \right\}. \tag{3.13}$$

Assume that the matrix $\overline{\Sigma} = (\overline{\Sigma}_{ij})$ is positive semidefinite for $i,j = 1,\ldots,n$. Problem 3.12 becomes then

$$\max_{w} \left\{ (\hat{\mu}_r - \delta)' w - \frac{1}{2} w' \overline{\Sigma} w \right\} \tag{3.14}$$

$$\begin{aligned} s.t.\ & e'w = 1, \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

This is a classical mean-variance portfolio optimization problem with expected asset returns given by $(\hat{\mu}_r - \delta)$ and covariance matrix of asset returns by $\overline{\Sigma}$. As such the solution of this problem is given by

$$w^* = \overline{\Sigma}^{-1} (\hat{\mu}_r - \delta - \lambda e) \tag{3.15}$$

and

$$\lambda^* = \frac{e' \overline{\Sigma}^{-1} (\hat{\mu}_r - \delta) - 1}{e' \overline{\Sigma}^{-1} e},$$

see also Section 1.2.1.

In the general case Problem 3.12 cannot be simply reformulated as Problem 3.14, because the upper bounds $\overline{\Sigma}_{ij}$ generally do not represent a valid covariance matrix, since $(\overline{\Sigma}_{ij})$ is not necessarily positive semidefinite. Therefore we use second-order cone optimization and duality.

Recall that we have already solved the first inner optimization problem of Problem 3.12 as Problem 3.4 with uncertainty set 3.2 for the expected returns. Now we discuss the second inner optimization problem of Problem 3.12 in general.
We consider the worst case for the risk $w'\Sigma w$ of the portfolio if the estimator of the covariance matrix Σ lies in uncertainty set 3.13. For a fixed weight vector w the worst case risk is given by

$$\max_{\Sigma} w'\Sigma w \tag{3.16}$$

$$\begin{aligned} s.t.\ \underline{\Sigma} \leq \Sigma &\leq \overline{\Sigma} \\ \Sigma &\succeq 0, \end{aligned}$$

where these order relations have to be understood as component-wise.
Problem 3.16 is a semidefinite program.

Recall the general formulation of dual problems in semidefinite programming, for example see Fabozzi et al. [5], Chapter 9 or Vandenberghe and Boyd [8]. We obtain this general formulation by rewriting Problem 3.16 as follows:

$$\max_{X} \langle C, X \rangle \tag{3.17}$$

$$\begin{aligned} s.t.\ \left\langle A^{(ij)}, X \right\rangle &\leq b^1_{ij},\ i,j = 1,\ldots,n \\ -\left\langle A^{(ij)}, X \right\rangle &\leq -b^2_{ij},\ i,j = 1,\ldots,n \\ X &\succeq 0, \end{aligned}$$

with $C = ww',\ X = \Sigma,\ b^1_{ij} = \overline{\Sigma}_{ij},\ b^2_{ij} = \underline{\Sigma}_{ij}$ and a $(n \times n)$ matrix

$$A^{(ij)} = \begin{pmatrix} 0 & \dots & \dots & \dots & 0 \\ \vdots & \ddots & & & \vdots \\ \vdots & & 1 & & \vdots \\ \vdots & & & \ddots & \vdots \\ 0 & \dots & \dots & \dots & 0 \end{pmatrix}$$

with all entries equal to zero except of a one in the i-th row and j-th column. The dual problem of Problem 3.17 is

$$\min_{u^1_{ij}, u_{ij2}} \left\{ \sum_{i=1}^{n} \sum_{j=1}^{n} (u^1_{ij} b^1_{ij} - u^2_{ij} b^2_{ij}) \right\} \tag{3.18}$$

$$\begin{aligned} s.t.\ -S + \sum_{i=1}^{n} \sum_{j=1}^{n} (u^1_{ij} A^{(ij)} - u^2_{ij} A^{(ij)}) = C, \\ S \succeq 0,\ u^1_{ij} \geq 0, u^2_{ij} \geq 0,\ i,j = 1, \dots, n \end{aligned}$$

with the dual variables $u^1_{ij} = \overline{\Lambda}_{ij},\ u^2_{ij} = \underline{\Lambda}_{ij}$ and the dual slack variable S. With this reformulation we obtain the dual problem of Problem 3.16.

$$\min_{\underline{\Lambda}, \overline{\Lambda}} \left\{ \langle \overline{\Lambda}, \overline{\Sigma} \rangle - \langle \underline{\Lambda}, \underline{\Sigma} \rangle \right\} \tag{3.19}$$

$$\begin{aligned} s.t.\ & -S + \overline{\Lambda} - \underline{\Lambda} = ww' \\ & S \succeq 0,\ \overline{\Lambda} \geq 0,\ \underline{\Lambda} \geq 0 \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

with the dual variables $\overline{\Lambda} = (\overline{\Lambda}_{ij})$ and $\underline{\Lambda} = (\underline{\Lambda}_{ij})$. Problem 3.19 can be rewritten as

$$\min_{\underline{\Lambda},\overline{\Lambda}} \left\{ \left\langle \overline{\Lambda}, \overline{\Sigma} \right\rangle - \left\langle \underline{\Lambda}, \underline{\Sigma} \right\rangle \right\} \tag{3.20}$$

$$\begin{aligned} s.t.\ & \overline{\Lambda} - \underline{\Lambda} - ww' \succeq 0 \\ & \overline{\Lambda} \geq 0,\ \underline{\Lambda} \geq 0 \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

Notice that the variable Σ does not occur in Problem 3.20. According to duality theory of semidefinite programming the optimal values of the primal and the dual problem are equal, provided that one of these problems is strictly feasible and bounded. The dual Problem 3.20 is bounded, because Σ is positive semidefinite and lies between the lower bound $\underline{\Sigma}$ and the upper bound $\overline{\Sigma}$, which are nonnegative matrices and the weights are nonnegative too. For the proof of the feasibility of Problem 3.20 we consider a $(n \times n)$ matrix $B = \alpha I + J$ with I as the identity matrix, J as a matrix with all entries equal to one and α as a positive constant. We set $B = \overline{\Lambda} - \underline{\Lambda}$ with $\overline{\Lambda} \geq 0, \underline{\Lambda} \geq 0$ (component-wise). So the eigenvalues of $-ww' + B$ are $\lambda(-ww' + B) = \lambda(-ww' + \alpha I + J) = \lambda(-ww' + J) + \alpha$. If we choose α large, then all eigenvalues $\lambda(-ww' + B) > 0$, which means that $-ww' + B$ is positive definite and $B > 0$ (component-wise). Thus Problem 3.20 is strictly feasible and we can substitute the objective function of Problem 3.16

$$\max_{\Sigma} w'\Sigma w$$

by the objective function of Problem 3.20

$$\min_{\underline{\Lambda},\overline{\Lambda}} \left\{ \left\langle \overline{\Lambda}, \overline{\Sigma} \right\rangle - \left\langle \underline{\Lambda}, \underline{\Sigma} \right\rangle \right\}$$

in the objective function of Problem 3.12.

Now we use the so-called *Schur complements* to make the optimization problem amenable to optimization software. The Schur complement of a square matrix D in a larger square matrix E

$$E = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$$

is defined as $A - BD^{-1}C$.
The first constraint of Problem 3.20 can be reformulated in a linear matrix inequality.

Consider the constraints $Q(x) - S(x)R(x)^{-1}S(x)' \succeq 0$ and $R(x) \succeq 0$, where x is a vector variable. Then these constraints can be reformulated as a linear matrix inequality:

$$\begin{pmatrix} Q(x) & S(x) \\ S(x)' & R(x) \end{pmatrix} \succeq 0,$$

see also Fabozzi et al. [5], Chapter 9.

In our case $Q(x) = \overline{\Lambda} - \underline{\Lambda}$, $S(x) = w$ and $R(x) = 1$ and the first constraint of Problem 3.20 can be reformulated as

$$\begin{pmatrix} \overline{\Lambda} - \underline{\Lambda} & w \\ w' & 1 \end{pmatrix} \succeq 0.$$

Putting things together in Problem 3.12 we get the following optimization problem:

$$\max_{w,\underline{\Lambda},\overline{\Lambda}} \left\{ (\hat{\mu}_r - \delta)'w - \frac{1}{2}(\langle \overline{\Lambda}, \overline{\Sigma} \rangle - \langle \underline{\Lambda}, \underline{\Sigma} \rangle) \right\} \tag{3.21}$$

$$\begin{aligned} s.t.\ & e'w = 1, \\ & w_i \geq 0,\ \forall i, \\ & \overline{\Lambda} \geq 0,\ \underline{\Lambda} \geq 0 \\ & \begin{pmatrix} \overline{\Lambda} - \underline{\Lambda} & w \\ w' & 1 \end{pmatrix} \succeq 0. \end{aligned}$$

Problem 3.21 can be solved by a nonlinear optimization software, for example by using the cvx-package in MATLAB 7.8.0.347 (R2009a).

3.3 Relative robustness

In Chapter 2 we have considered relative robust optimization problems in general. In this section we discuss the relative robustness of portfolio optimization. We are looking for the worst case relative to the best possible solutions in each scenario.
For a fixed $\mu_r \in \mathbb{U}(\hat{\mu}_r)$, where $\mathbb{U}(\hat{\mu}_r)$ is the uncertainty set for the expected returns μ_r, we consider the optimal value function $t^*(\mu_r)$.

$$t^*(\mu_r) = \max_w \left\{ \mu_r' w - \frac{1}{2} w' \Sigma w \right\}$$

$$\begin{aligned} s.t.\ & e'w = 1 \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

Let $w^*(\mu_r)$ be the optimal solution map:

$$w^*(\mu_r) = argmax_w \left\{ \mu_r' w - \frac{1}{2} w' \Sigma w \right\}$$

$$\begin{aligned} s.t.\ & e'w = 1 \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

Let us define the measure of regret as follows:

$$r(w, \mu_r) = \mu_r' w^*(\mu_r) - \frac{1}{2}(w^*(\mu_r))' \Sigma w^*(\mu_r) - (\mu_r' w - \frac{1}{2} w' \Sigma w) = t^*(\mu_r) - (\mu_r' w - \frac{1}{2} w' \Sigma w) \geq 0.$$

For a fixed w we maximize the regret function:

$$R(w) = \max_{\mu_r \in \mathbb{U}(\hat{\mu}_r)} r(w, \mu_r) = \max_{\mu_r \in \mathbb{U}(\hat{\mu}_r)} \left\{ t^*(\mu_r) - (\mu_r' w - \frac{1}{2} w' \Sigma w) \right\}.$$

The relative robust optimization problem is the minimization of the maximized regret:

$$\min_w \max_{\mu_r \in \mathbb{U}(\hat{\mu}_r)} \left\{ t^*(\mu_r) - (\mu_r' w - \frac{1}{2} w' \Sigma w) \right\}. \tag{3.22}$$

$$\begin{aligned} s.t.\ & e'w = 1, \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

Problem 3.22 is a three level optimization problem and is difficult to solve, see also Problem 2.18 in Chapter 2.

A possibility to solve relative robust portfolio optimization problems is to limit the maximized regret to a level M.

$$\textit{Find } w \tag{3.23}$$

$$\begin{aligned} s.t.\ & t^*(\mu_r) - \mu_r' w \leq M,\ \forall \mu_r \in \mathbb{U}(\hat{\mu}_r) \\ & e'w = 1, \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

Now we consider an example where the objective function is the maximum of the expected returns of a portfolio with two assets and a third one which represents the part of capital which is not invested. A portfolio which invests half of half of the capital in each asset is considered as a "benchmark". If w denotes the vector of weights for the portfolio and w_{BM} denotes the vector of weights for the benchmark, then the tracking error $TE(w)$ is defined as

$$TE(w) := \sqrt{(w - w_{BM})'\Sigma(w - w_{BM})},$$

where Σ is the covariance matrix. Assume that we want a tracking error $TE(w)$ which is smaller than 10%, while the budget equation holds and no short sellings are allowed.

This optimization problem reads as follows

$$\max_w \mu_1 w_1 + \mu_2 w_2 + \mu_3 w_3 \tag{3.24}$$

$$\begin{aligned} s.t.\ & TE(w_1, w_2, w_3) \leq 0.10 \\ & w_1 + w_2 + w_3 = 1 \\ & w_1 \geq 0, w_2 \geq 0, w_3 \geq 0. \end{aligned}$$

We assume that Σ is given as $\Sigma = \begin{pmatrix} 0.1764 & 0.09702 & 0 \\ 0.09702 & 0.1089 & 0 \\ 0 & 0 & 0 \end{pmatrix}$, then

$$TE(w_1, w_2, w_3) = \sqrt{\begin{pmatrix} w_1 - 0.5 \\ w_2 - 0.5 \\ w_3 \end{pmatrix}' \begin{pmatrix} 0.1764 & 0.09702 & 0 \\ 0.09702 & 0.1089 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} w_1 - 0.5 \\ w_2 - 0.5 \\ w_3 \end{pmatrix}}.$$

The first asset has a standard deviation of $\sqrt{0.1764} = 42\%$ and the second asset a standard deviation of $\sqrt{0.1089} = 33\%$ and both have a correlation coefficient of $\frac{0.09702}{\sqrt{0.1764 \cdot 0.1089}} = 0.7$.

We consider a relative robustness model where the covariance matrix is certain, but the expected returns are uncertain and their uncertainty set consists of the three possible scenarios for (μ_1, μ_2, μ_3):

- $(6, 4, 0)$ with the optimal solution $w^*(\mu_r) = (0.831, 0.169, 0)$ and the objective value $t^*(\mu_r) = 5.662$.
- $(4, 6, 0)$ with the optimal solution $w^*(\mu_r) = (0.169, 0.831, 0)$ and the objective value $t^*(\mu_r) = 5.662$.
- $(5, 5, 0)$ with the optimal solution $w^*(\mu_r) = (0.5, 0.5, 0)$ and the objective value $t^*(\mu_r) = 5.0$.

The corresponding relative robust optimization problem can be formulated as follows:

$$\min_{w,y} y \tag{3.25}$$

$$\begin{aligned} s.t.\ & 5.662 - (6w_1 + 4w_2) \le y \\ & 5.662 - (4w_1 + 6w_2) \le y \\ & 5.0 - (5w_1 + 5w_2) \le y \\ & TE(w_1, w_2, w_3) \le 0.10 \\ & w_1 + w_2 + w_3 = 1 \\ & w_1 \ge 0, w_2 \ge 0, w_3 \ge 0 \end{aligned}$$

For the optimal solution we use MATLAB 7.8.0.347 (R2009a). The optimal value for y is given as $y^* = 0.662$ with the optimal weight $w^* = (0.5, 0.5, 0)$.

An easier strategy is to determine a tolerable level of regret M, for example 0.75, and find portfolios that do not exceed this level as in Problem 3.23.

$$Find\ w \tag{3.26}$$

$$\begin{aligned} s.t.\ & 5.662 - (6w_1 + 4w_2) \leq 0.75 \\ & 5.662 - (4w_1 + 6w_2) \leq 0.75 \\ & 5.0 - (5w_1 + 5w_2) \leq 0.75 \\ & TE(w_1, w_2, w_3) \leq 0.10 \\ & w_1 + w_2 + w_3 = 1 \\ & w_1 \geq 0, w_2 \geq 0, w_3 \geq 0 \end{aligned}$$

The optimal weight is given as $w^* = (0.5, 0.5, 0)$.

3.4 Multi-period robustness

We consider a time horizon of $T + 1$ periods and try to maximize the total final wealth at the end of the last period. Currently the investor holds a portfolio of n assets with weights $w^0 = (w_1^0, \ldots, w_n^0)$ and a riskless cash account with weight w_c^0. There occur sales and purchases too for which we need transaction costs. Let $b^t = (b_1^t, \ldots, b_n^t)$ be the percentage of capital invested in the shares of purchases at the beginning of period t and let $s^t = (s_1^t, \ldots, s_n^t)$ be the percentage of capital invested in the shares of sales at the beginning of period t, $t = 0, \ldots, T$. Denote by $w^t = (w_1^t, \ldots, w_n^t)$ the percentage of shares at the beginning of period t:

$$w^t = w^{t-1} - s^t + b^t,\ t = 1, \ldots, T. \tag{3.27}$$

So the percentage of shares at the beginning of period t is the weight of the beginning of period $t - 1$ minus the sales plus the purchases at the beginning of period t.
Define with $P^t = (P_1^t, \ldots, P_n^t)$ the price of a share in period t for $t = 0, \ldots, T$ and assume that for the cash account $P_c^t = 1,\ t = 0, \ldots, T.$, holds. The proportional transaction costs for the asset sales and purchases $\alpha^t = (\alpha_1^t, \ldots, \alpha_n^t)$ and $\beta^t = (\beta_1^t, \ldots, \beta_n^t),\ t = 0, \ldots, T,$

respectively, are assumed to be known at the beginning of period 0 and they change from period to period and from asset to asset. For the cash account we get the following equation where the transaction costs are paid from this cash account:

$$w_c^t = w_c^{t-1} + (1-\alpha^t)(P^t)'s^t - (1+\beta^t)(P^t)'b^t,\ t = 1,\dots,T. \tag{3.28}$$

Thus the cash account at the beginning of period t is obtained from the cash account of the previous period $t-1$ by subtracting or adding the transaction, selling and purchasing costs, respectively. In our optimization problem we will allow the investor to "burn" some money if he wants and we replace the equation by an inequality:

$$w_c^t \leq w_c^{t-1} + (1-\alpha^t)(P^t)'s^t - (1+\beta^t)(P^t)'b^t,\ t = 1,\dots,T. \tag{3.29}$$

The reason for the use of inequalities is that robust problems with equalities constraints get easily infeasible.

We also do not allow short selling and hence get non-negative constraints for w^t. Thus the multi-period optimization problem reads

$$\max_{w,s,b} \left\{ P^{T+1} w^{T+1} + P_c^{T+1} w_c^{T+1} \right\} \tag{3.30}$$

$$\begin{aligned}
s.t.\ & w_c^t \leq w_c^{t-1} + (1-\alpha^t)(P^t)'s^t - (1+\beta^t)(P^t)'b^t, && t = 1,\dots,T \\
& w^t = w^{t-1} - s^t + b^t, && t = 1,\dots,T \\
& w_c^t \geq 0,\ w^t \geq 0, && t = 1,\dots,T \\
& s^t \geq 0,\ b^t \geq 0, && t = 1,\dots,T.
\end{aligned}$$

Problem 3.30 could be solved easily as a linear program if we would know the prices P^{T+1} and P_c^{T+1}.
In reality we do not know the prices P^t for $t = 1,\dots,T$ and so we cannot solve Problem 3.30 as a linear program. All we have are estimators $\hat{\mu}_P^t$ for the expected values of the prices $P^t, t = 1,\dots,T$. Thus we have to make the problem robust to the uncertainty of the prices. We reformulate the problem to get all uncertain prices in the constraints.

$$\max_{w,s,b,y} y \tag{3.31}$$

$$\begin{aligned} s.t.\ & y \leq P^{T+1}w^{T+1} + P_c^{T+1}w_c^{T+1} & \\ & w_c^t \leq w_c^{t-1} + (1-\alpha^t)(P^t)'s^t - (1+\beta^t)(P^t)'b^t, & t=1,\dots,T \\ & w^t = w^{t-1} - s^t + b^t, & t=1,\dots,T \\ & w_c^t \geq 0,\ w^t \geq 0, & t=1,\dots,T \\ & s^t \geq 0,\ b^t \geq 0, & t=1,\dots,T \end{aligned}$$

We consider a so-called 3-σ approach in order to obtain an uncertainty set for the prices. Denote the expected values of P^t with $\mu_P^t = (\mu_{P_1}^t, \dots, \mu_{P_n}^t)$ and the covariance matrix with $\Sigma^t = (\Sigma_1^t, \dots, \Sigma_n^t)$ for $t = 0, \dots, T$.

Now we take a look at the constraints with the uncertain parameters of Problem 3.31. Consider the first constraint:

$$y \leq P^{T+1}w^{T+1} + P_c^{T+1}w_c^{T+1} \tag{3.32}$$

The expected value of the right hand side of Inequality 3.32 is

$$\mathbb{E}[P^{T+1}w^{T+1} + P_c^{T+1}w_c^{T+1}] = \mu_P^{T+1}w^{T+1} + w_c^{T+1} \tag{3.33}$$

and the variance is

$$Var[P^{T+1}w^{T+1} + P_c^{T+1}w_c^{T+1}] = (w^{T+1})'\Sigma^{T+1}(w^{T+1}). \tag{3.34}$$

If the prices are normally distributed and we consider the inequality

$$\begin{aligned} y \leq & \mathbb{E}[P^{T+1}w^{T+1} + P_c^{T+1}w_c^{T+1}] - 3\sqrt{Var[P^{T+1}w^{T+1} + P_c^{T+1}w_c^{T+1}]} \\ = & \mu_P^{T+1}w^{T+1} + w_c^{T+1} - 3\sqrt{(w^{T+1})'\Sigma^{T+1}(w^{T+1})} \end{aligned}$$

the first constraint of Problem 3.31 would be satisfied with a probability larger than 99% over all possible realizations of prices P^{T+1}. This is the robust counterpart for the first constraint of Problem 3.31.

The second constraint of Problem 3.31 is

$$w_c^t \leq w_c^{t-1} + (1-\alpha^t)(P^t)'s^t - (1+\beta^t)(P^t)'b^t,\ t = 1,\dots,T. \tag{3.35}$$

We put the weight of the cash account at the beginning of period $t-1$ to the left hand side of Inequality 3.35, so that all uncertain parameters are isolated on the right hand side:

$$w_c^t - w_c^{t-1} \leq (1-\alpha^t)(P^t)'s^t - (1+\beta^t)(P^t)'b^t,\ t = 1,\dots,T. \tag{3.36}$$

Again, let us consider the expected value and the variance of the right hand side of Inequality 3.36:

$$\mathbb{E}[(1-\alpha^t)(P^t)'s^t - (1+\beta^t)(P^t)'b^t] = (\mu_P^t)'D_\alpha^t s^t - (\mu_P^t)'D_\beta^t b^t = (\mu_P^t)'(D_\alpha^t; -D_\beta^t)\begin{pmatrix} s^t \\ b^t \end{pmatrix}, \tag{3.37}$$

$$Var[(1-\alpha^t)(P^t)'s^t - (1+\beta^t)(P^t)'b^t] = \begin{pmatrix} s^t \\ b^t \end{pmatrix}' \begin{pmatrix} D_\alpha^t \\ -D_\beta^t \end{pmatrix} \Sigma^t (D_\alpha^t; -D_\beta^t) \begin{pmatrix} s^t \\ b^t \end{pmatrix}, \tag{3.38}$$

with the diagonal matrices

$$D_\alpha^t := \begin{pmatrix} (1-\alpha_1^t) & & \\ & \ddots & \\ & & (1-\alpha_n^t) \end{pmatrix},\ D_\beta^t := \begin{pmatrix} (1+\beta_1^t) & & \\ & \ddots & \\ & & (1+\beta_n^t) \end{pmatrix}.$$

Using the expected value 3.37 and the variance 3.38, we can replace the second constraint of Problem 3.31 with its robust counterpart:

$$w_c^t - w_c^{t-1} \leq (\mu_P^t)'(D_\alpha^t - D_\beta^t)\begin{pmatrix} s^t \\ b^t \end{pmatrix} - 3\sqrt{\begin{pmatrix} s^t \\ b^t \end{pmatrix}' \begin{pmatrix} D_\alpha^t \\ -D_\beta^t \end{pmatrix} \Sigma^t (D_\alpha^t - D_\beta^t) \begin{pmatrix} s^t \\ b^t \end{pmatrix}}. \tag{3.39}$$

Again, assuming that the uncertain parameters are normally distributed, the original constraint would be satisfied with a probability of at least 99%.

This approach (the so-called 3-σ approach) we have used, lead to the following uncertainty set for the prices:

$$\mathbb{U}^t(P^t) = \left\{ P^t : \sqrt{(P^t - \mu_P^t)'(\Sigma^t)^{-1}(P^t - \mu_P^t)} \leq 3 \right\},\ t = 1,\dots,T.$$

The complete uncertainty set $\mathbb{U}$ is the Cartesian product $\mathbb{U} = \mathbb{U}^1 \times \ldots \times \mathbb{U}^{T+1}$. See also Cornuejols and Tütüncü [4], Chapter 20.

Another way to compute an uncertainty set is by assuming that just the covariance matrix of the first period is known. The basic idea is that we know more about the prices of the single-period problem than of prices far in the future. So we get the following uncertainty sets for the prices:

$$\mathbb{U}^t(P^1) = \left\{ P^1 : \sqrt{(P^1 - \mu_P^1)'(\Sigma^1)^{-1}(P^1 - \mu_P^1)} \leq \delta \right\} \tag{3.40}$$

and

$$\mathbb{U}^t(P^t) = \left\{ P^t : \underline{P}^t \leq P^t \leq \overline{P}^t,\ t = 2, \ldots, T \right\} \tag{3.41}$$

with lower and upper bounds $\underline{P}$ and $\overline{P}$. In the case of uncertainty sets 3.40 and 3.41 we account for the risk over the first period and we keep in mind some basic forecasts about the direction of the prices in the following time periods. See the paper of Bertsimas and Pachamanova [9], too.

3.5 Conclusion

In this chapter we got to know robust formulations of the mean-variance optimization problem with uncertain input parameters. We considered two special cases of uncertainty sets for the expected returns: The polytopic and the ellipsoidal uncertainty sets.

We sketched the solution of the problem for expected returns lying in a polytopic uncertainty set and for the covariance matrix lying in an interval. But in general the covariance matrix of the worst case is not alway positive definite. So we formulated the optimization problem with the polytopic uncertainty set for expected returns and the interval for the covariance matrix as a semidefinite program.
We also discussed an example of relative robustness and the robust formulation of the multi-period problem.

In the next chapter we want to apply the theory presented in the previous chapters to compare classical with robust mean-variance optimization problems in terms of a practical example.

4 Portfolio Optimization in Practice

4.1 Data

Our portfolio consists of $n = 5$ indices, namely:

- Nasdaq Composite
- S&P 500
- FTSE 100
- Dax
- Nikkei 225

We consider the weekly historical prices p_t from November 26th, 1990, till January 10th, 2010, and compute the weekly logarithmic returns defined as $\mu_r(t) = log(\frac{p_{t+1}}{p_t})$, where p_t is the price realized at the t-th week of the above described time window. In this way we obtain a time series of length $N = 996$ for each asset.

All statistical computations are done in R 2.10.1. The solution of the optimization problem is done by using the cvx-package of MATLAB 7.8.0.347 (R2009a).

4.1.1 Implementation in R

We use the technique of bootstrapping to estimate the expected returns μ_r and the covariance matrix Σ. There we get the 25%, 50% and 75% quantiles for the expected returns and the covariances σ_{ij}. The 50% quantiles are used as estimators for μ_r and $\Sigma = (\Sigma_{ij})$ and the 25% and 75% quantiles are used as lower and upper bounds of the uncertainty sets in the robust optimization problem.

As an illustrative example we describe the implementation of the bootstrapping approach in the case of Nasdaq. The R code looks as follows.

```
Nasdaqres <- lapply(1:3000, function(i)
sample(Nasdaqlog, replace = T))
Nasdaqmean <- sapply(Nasdaqres, mean)
Nasdaqquantil<-quantile(Nasdaqmean, probs=c(0.25,0.5,0.75))
```

The explanations of the used functions are as follows:

- `lapply:` returns a list of the same length as Nasdaqlog. Each element is the result of applying the function `sample` to the corresponding element of Nasdaqlog.
- `sample:` samples from the vector Nasdaqlog with replacement (`replace=T`).
- `quantile:` returns the sample quantiles for given probabilities. We get the 25%, 50% and 75% quantiles (`probs=c(0.25,0.5,0.75)`).

The resulting vectors are explained below:

- `Nasdaqlog:` a vector of the expected log returns of Nasdaq.
- `Nasdaqres:` 3000 vectors of length 996 which are the 3000 resamples of the time series of length 996.
- `Nasdaqmean:` a vector of length 3000 with the means of each resample.
- `Nasdaqquantil:` the values of the 25%, 50% and 75% quantiles of the expected returns (μ_r^L, μ_r and μ_r^U).

The same code is used for the indices S&P 500, FTSE 100, DAX and Nikkei 225.

The bootstrapping of the covariance matrix was more complicated. We implement it as follows.

```
Nasdaqmat<-matrix(1:2988000, ncol=3000)
SPmat<-matrix(1:2988000, ncol=3000)
FTSEmat<-matrix(1:2988000, ncol=3000)
Daxmat<-matrix(1:2988000, ncol=3000)
Nikkeimat<-matrix(1:2988000, ncol=3000)
```

```
for(i in 1:996){for(j in 1:3000){
Nasdaqmat[i,j]<-matrix(Nasdaqres[[j]][i])
SPmat[i,j]<-matrix(SPres[[j]][i])
FTSEmat[i,j]<-matrix(FTSEres[[j]][i])
Daxmat[i,j]<-matrix(Daxres[[j]][i])
Nikkeimat[i,j]<-matrix(Nikkeires[[j]][i])}}

covariance<-matrix(1:75000,ncol=5)
for(j in 1:3000){
l<-((j*5)-4)
k<-(j*5)
covariance[l:k,1:5]<-cov(matrix(c(Nasdaqmat[,j],SPmat[,j],FTSEmat[,j],
                                  Daxmat[,j],Nikkeimat[,j]),ncol=5))}
```

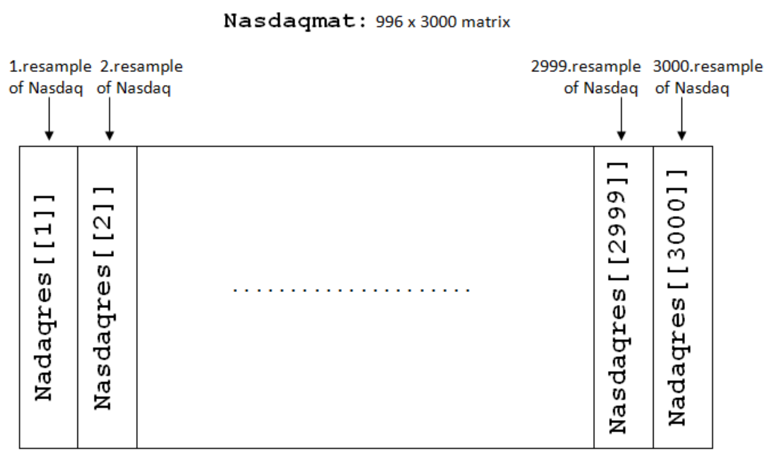

Figure 4.1: Explanation of the matrix Nasdaqmat

First we initialize 5 matrices `Nasdaqmat`, `SPmat`, `FTSEmat`, `Daxmat` and `Nikkeimat` with 996 rows and 3000 columns each. Then we fill the matrices with the columns of the 3000 resampled time series of length 996. Finally we generate a matrix `covariance` with 15000 rows and 5 columns. To this end we generate 3000 matrices with 5 columns and 996 rows filled with the first, second,..., 3000th column of `Nasdaqmat`, `SPmat`, `FTSEmat`, `Daxmat` and `Nikkeimat` by calling
`matrix(c(Nasdaqmat[,j],SPmat[,j],FTSEmat[,j],Daxmat[,j],Nikkeimat[,j]),ncol=5)`.

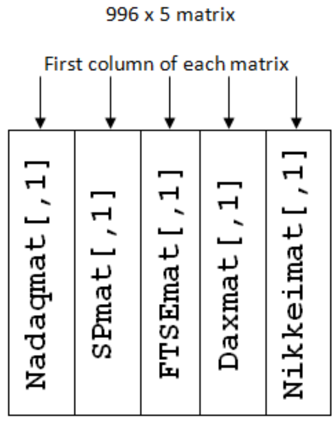

Figure 4.2: Explanation of the 996 x 5 matrix

Then we compute the covariance of all 3000 matrices and obtain 3000 5×5-matrices. These covariance matrices are concatenated in the 1500 × 5 matrix `covariance.`

Next we generate a vector of length 3000 with the first element of each covariance matrix:

```
for(j in 1:3000){l<-((j*5)-4) vec11[j]<-c(covariance[l,1])}
```

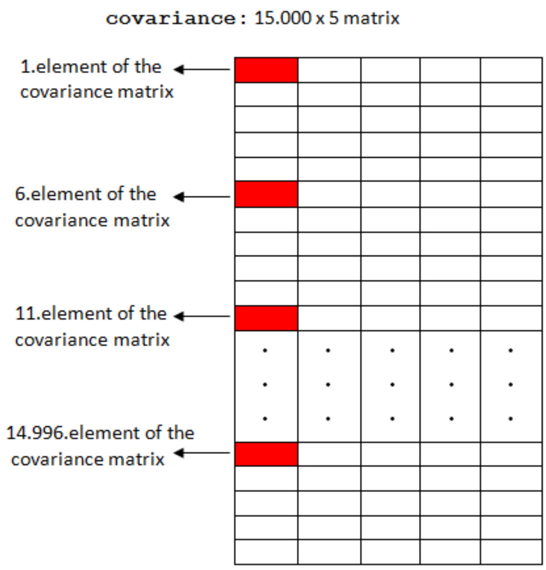

Figure 4.3: Explanation of the vector vec11

This procedure is repeated for all elements of each covariance matrix resulting in 25 vectors called `vecij`, $\forall i, j, 1 \leq i, j \leq 5$ (the red fields are the vector `vec11` of length 3000). For each vector we compute the 25%, 50% and 75% quantiles called `quantij`, $\forall i, j, 1 \leq i, j \leq 5$:

```
quantvec11<-quantile(vec11, probs=c(0.25,0.5,0.75))
```

We summarize the 25% quantiles in the matrix `covlow` ($\underline{\Sigma}$), equivalently the 50% quantiles and 75% quantiles of the covariance matrix in `cov` (Σ) and in `covup` ($\overline{\Sigma}$):

```
covlow<-matrix(c(quant11[[1]],quant21[[1]],quant31[[1]],quant41[[1]],quant51[[1]],
                 quant12[[1]],quant22[[1]],quant32[[1]],quant42[[1]],quant52[[1]],
                 quant13[[1]],quant23[[1]],quant33[[1]],quant43[[1]],quant53[[1]],
                 quant14[[1]],quant24[[1]],quant34[[1]],quant44[[1]],quant54[[1]],
                 quant15[[1]],quant25[[1]],quant35[[1]],quant45[[1]],quant55[[1]]),
                 ncol=5)
```

The estimations of and the bounds on the covariance matrix obtained by applying the bootstrapping approach described above do not have to be positive definite. If the matrix is not positive definite, that means, if the matrix has negative eigenvalues, than we need an approximation to transform the matrix in a positive definite one. Denote by $\hat{\Sigma}$ the estimator of a covariance matrix, which is symmetric, but not neccessarily positive definite. We look for the "closest" positive definite matrix to $\hat{\Sigma}$ and use the Frobenius norm as a measure of closeness:

$$d_F(\Sigma, \hat{\Sigma}) = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} (\Sigma_{ij} - \hat{\Sigma}_{ij})^2}$$

with a positive definite covariance matrix Σ. The optimization problem for the nearest covariance matrix for a given matrix $\hat{\Sigma}$ reads as follows:

$$\min_{\Sigma} d_F(\Sigma, \hat{\Sigma}) \tag{4.1}$$

$$s.t.\ \Sigma \in C_s^n,$$

with C_s^n as the cone of $n \times n$ symmetric and positive definite matrices

$$C_s^n = \left\{ X = \begin{pmatrix} x_{11} & \dots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{n1} & \dots & x_{nn} \end{pmatrix} \in \mathbb{R}^{n \times n} : X \succ 0 \right\},$$

and the notation $X \succ 0$ means that the matrix X is positive definite.
Now we introduce a dummy variable t and rewrite Problem 4.1 as

$$min\ t \tag{4.2}$$

$$s.t.\ d_F(\Sigma, \hat{\Sigma}) \leq t$$
$$\Sigma \in C_s^n.$$

We notice that the first constraint of Problem 4.2 can be written as a second-order cone constraint and so we can transform Problem 4.2 into a conic optimization problem, which can be solved by a nonlinear optimization software. See also Tütüncü and Koenig [4].

4.1.2 Data Analysis

By bootstrapping we get the 50% quantiles for the covariances and the expected returns. First consider the covariance matrix:

$$\Sigma = \begin{pmatrix} 0.0011085553 & 0.0000007558 & -0.0000004174 & 0.0000001376 & 0.0000006867 \\ 0.0000007558 & 0.0005599362 & 0.0000005382 & -0.0000003611 & 0.0000007006 \\ -0.0000004174 & 0.0000005382 & 0.0005654621 & -0.0000003275 & 0.0000001593 \\ 0.0000001376 & -0.0000003611 & -0.0000003275 & 0.0010058322 & -0.0000013193 \\ 0.0000006867 & 0.0000007006 & 0.0000001593 & -0.0000013193 & 0.0009486372 \end{pmatrix}.$$

In this case the covariance matrix Σ is positive definite. Thus no approximation is needed.

The following table shows the variances of the different indices sorted in decreasing order:

index	variance
Nasdaq	0.0011085553
DAX	0.0010058322
Nikkei 225	0.0009486372
FTSE 100	0.0005654621
S&P 500	0.0005599362

Table 4.1: Variances of indices

Now consider the 50% quantile of the expected returns:

$$\mu_r = \begin{pmatrix} 0.001834631 \\ 0.0012650497 \\ 0.0008917582 \\ 0.0014501784 \\ -0.0007872628 \end{pmatrix}.$$

The index with the largest 50% quantile expected return is the Nasdaq, followed by the Dax, S&P 500, FTSE 100 and finally the Nikkei 225 with a negative expected return. In the following we see the charts of all indices from November 1990 till January 2010:

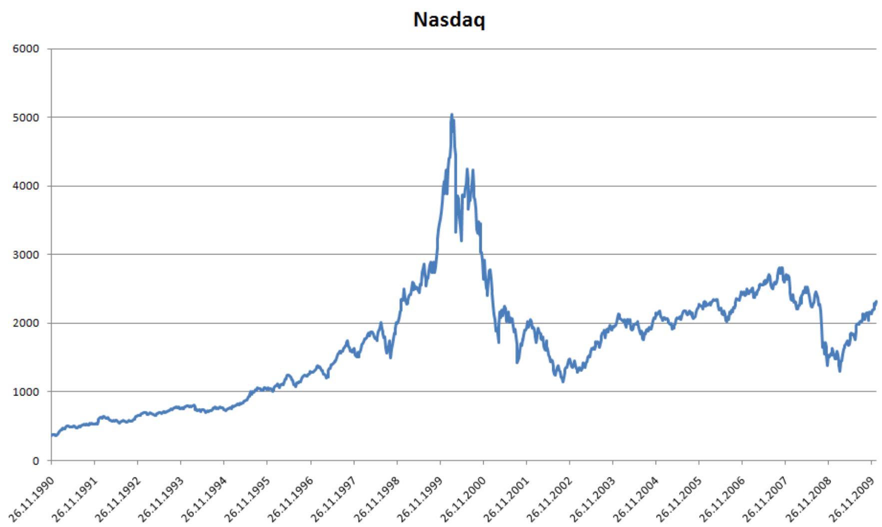

Figure 4.4: Prices of Nasdaq Composite

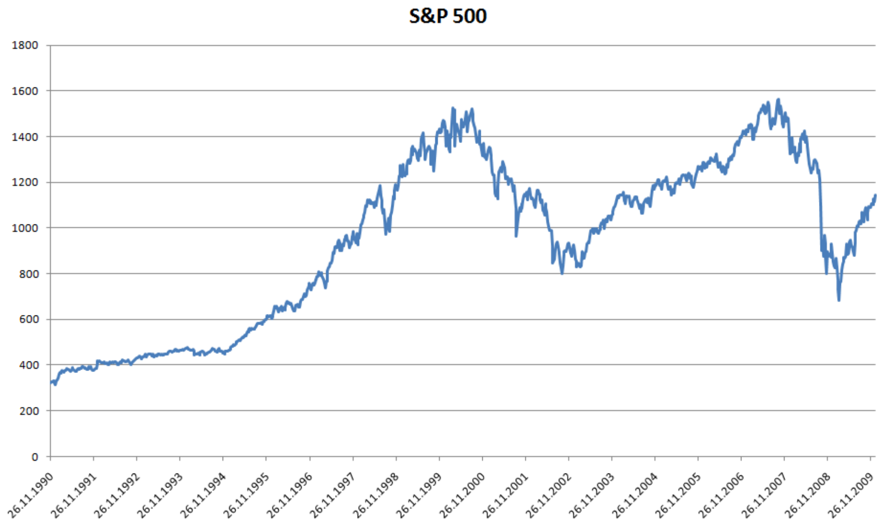

Figure 4.5: Prices of S&P 500

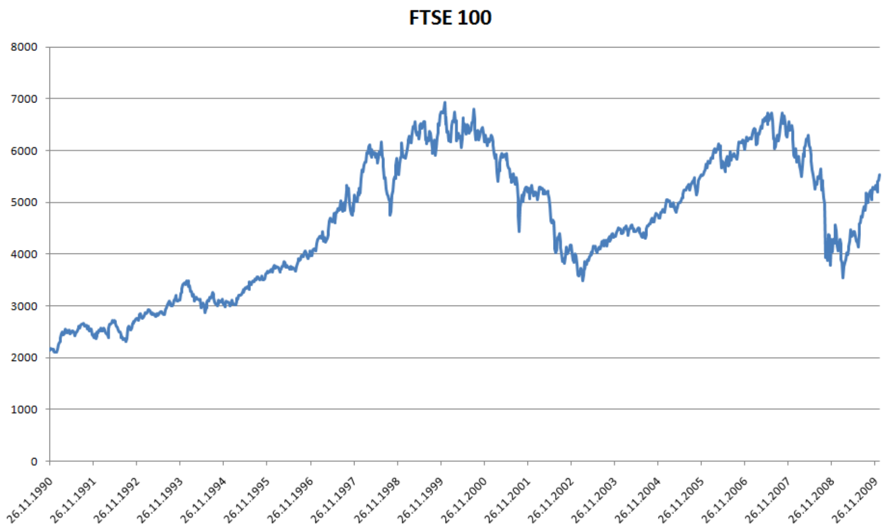

Figure 4.6: Prices of FTSE 100

Figure 4.7: Prices of DAX

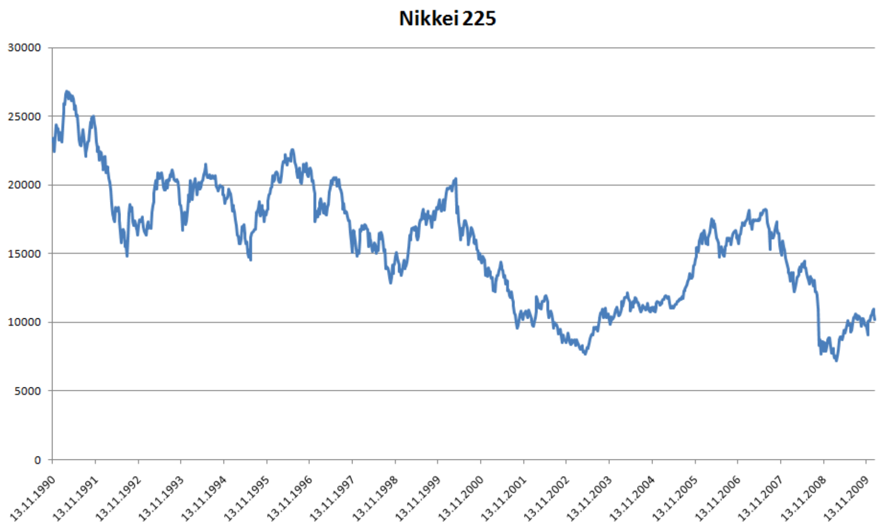

Figure 4.8: Prices of Nikkei 225

At this point a natural question arises: To which extent do the values μ_r (computed as 50% quantiles fo the sampled returns) correspond to the realized returns?

Let us compare the realized prices for each index with the expected prices assuming that the expected returns are μ_r computed as above.

The weekly expected returns are given as e^{μ_r}, where μ_r is the weekly expected log return. The expected returns over the whole time period are given as $(e^{\mu_r})^{996}$. Hence the expected prices of the end of the considered time period are given as $p_o \cdot (e^{\mu_r})^{996}$, where p_0 are the initial prices. The results of these comparisons are summarized in the following table:

index	realized price	expected price
Nasdaq	2317.17	2213.192
S&P 500	1144.98	1136.119
FSTE 100	5534.2	5223.799
Dax	6037.61	6109.554
Nikkei 225	10198.04	10578.58

Table 4.2: Comparison of realized and expected prices

Table 4.2 shows a gap between the estimated expected return and the returns realized in the past. So we have to consider the uncertainty of the parameters in the optimization problem which leads us to our next considerations.

4.2 Comparison of the parameters

In this section we will consider the classical optimization problems:

$$\max_w \mu_r' w \quad (4.3) \qquad \min_w w'\Sigma w \quad (4.4) \qquad \max_w \left\{\mu_r' w - c w'\Sigma w\right\} \quad (4.5)$$

$$\begin{aligned} s.t.\ & e'w = 1 \\ & w'\Sigma w \leq s \\ & w_i \geq 0,\ \forall i \end{aligned} \qquad \begin{aligned} s.t.\ & e'w = 1 \\ & \mu_r' w \geq m \\ & w_i \geq 0,\ \forall i \end{aligned} \qquad \begin{aligned} s.t.\ & e'w = 1 \\ & w_i \geq 0,\ \forall i \end{aligned}$$

where μ_r is the 50% quantile of the resampled returns and Σ is obtained by collecting the 50% quantiles of the resampled entries of the covariance matrix and eventually approximating by a positive definite matrix.

These problems are parametrized with the parameters s, m and c, respectively. We consider their solutions for different values of the parameters. We denote the smallest value of s by s_{min} and its largest value by s_{max}. Analogously we denote by m_{min} and m_{max} the smallest and the largest value of m, respectively. The smallest value of the risk aversion parameter c_{min} is equal to 0. We set s_{min} (s_{max}) equal to the smallest (largest) value of $\Sigma = (\sigma_{ij})$ and m_{min} (m_{max}) equal to the smallest (largest) value of μ_r. The largest value c_{max} is 3.5 because the solutions do not vary a lot for values larger than 3.5. In order to get different values for the parameters, we partition the intervals $[s_{min}, s_{max}]$, $[m_{min}, m_{max}]$ and $[c_{min}, c_{max}]$ in 20 equal parts:

- Interval for the expected return m: $[-0.00078726, 0.0018]$ with stepsize 0.00013109.
- Interval for the variance s: $[-0.0000013193, 0.0011]$ with stepsize 0.000055494.
- Interval for the risk aversion parameter c: $[0, 3.5]$ with stepsize 0.1750.

We compare the solutions of the portfolio optimization problems 4.5, 4.4 and 4.3 for different values of the control parameters c, m and s, respectively. Now we solve the optimization problems and get the portfolio weights for all problems. Let us denote by

- w_1 the percentage invested in Nasdaq,
- w_2 the percentage invested in S&P 500,
- w_3 the percentage invested in FTSE 100,
- w_4 the percentage invested in Dax and
- w_5 the percentage invested in Nikkei 225.

The efficient frontiers of all three problems are depicted in the following graphics.

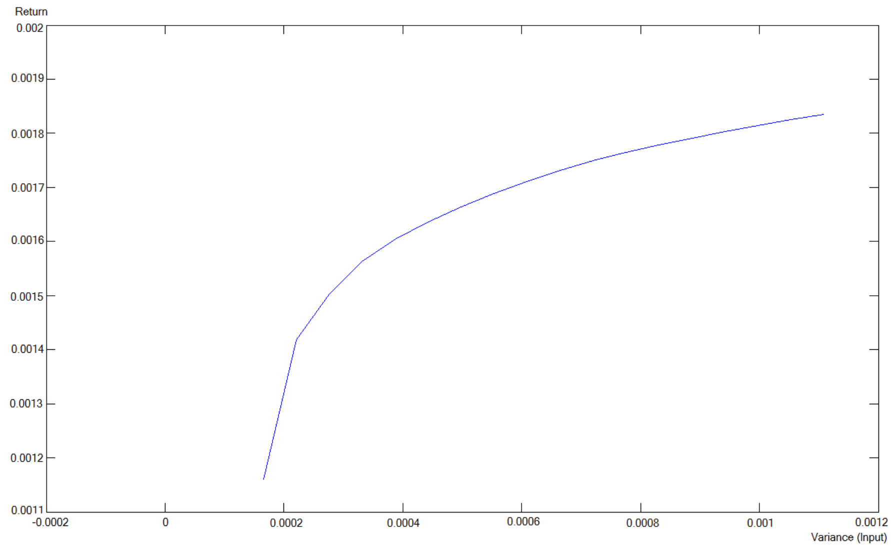

Figure 4.9: Classical efficient frontier of Problem 4.3

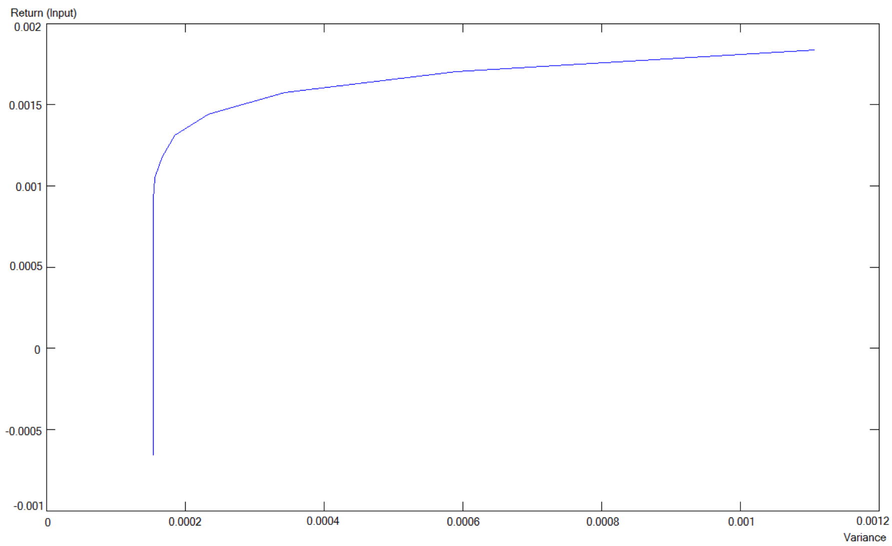

Figure 4.10: Classical efficient frontier of Problem 4.4

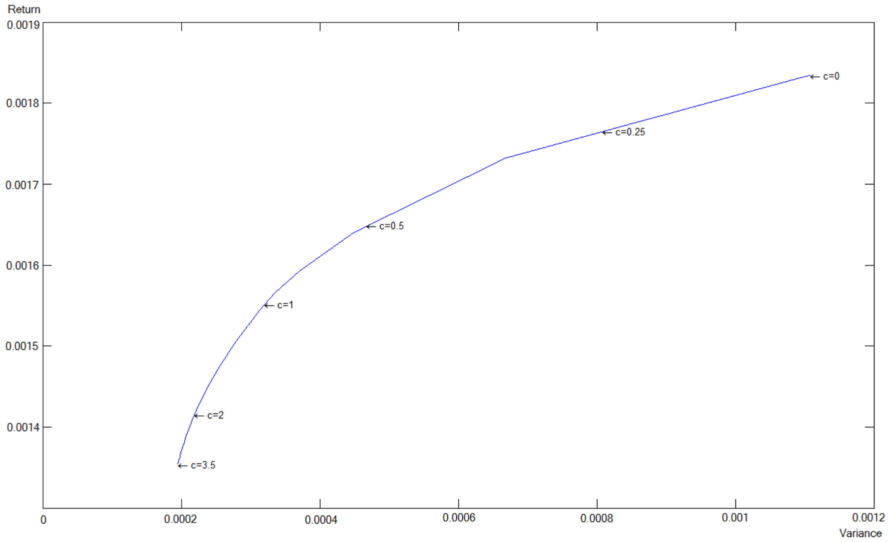

Figure 4.11: Classical efficient frontier of Problem 4.5

In order to interpret the parameter c of Problem 4.5, we compare the composition of the efficient portfolios of Problem 4.3, Problem 4.4 and Problem 4.5, which are illustrated in the following graphics. For every value of the control parameter (depicted in the horizontal axis) the length of each monochrome interval represents the percentage of capital invested in the corresponding asset.

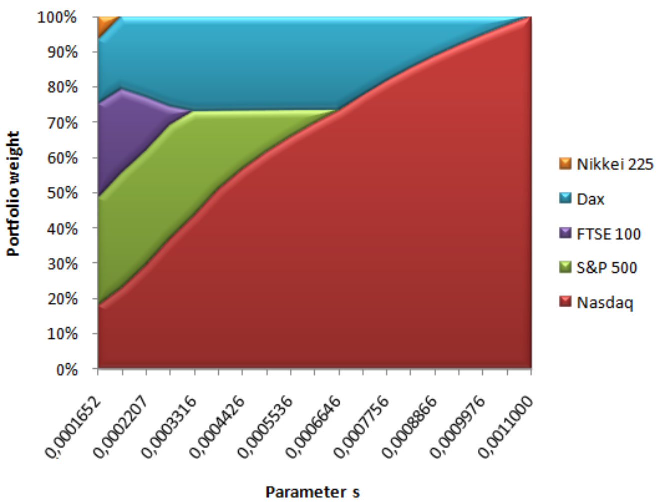

Figure 4.12: Composition of efficient portfolios of Problem 4.3

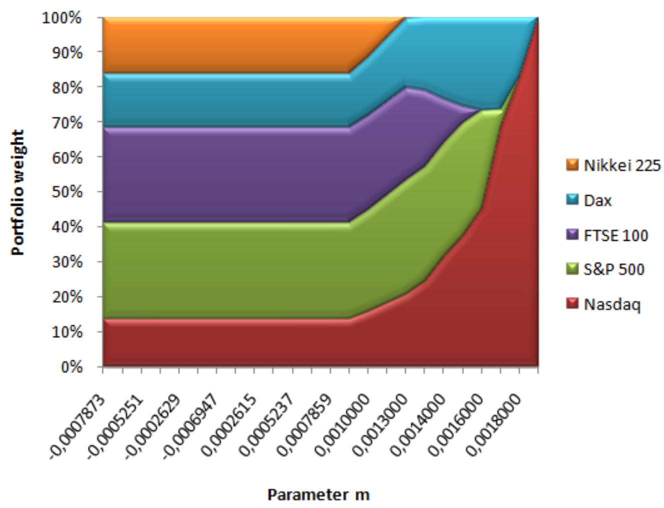

Figure 4.13: Composition of efficient portfolios of Problem 4.4

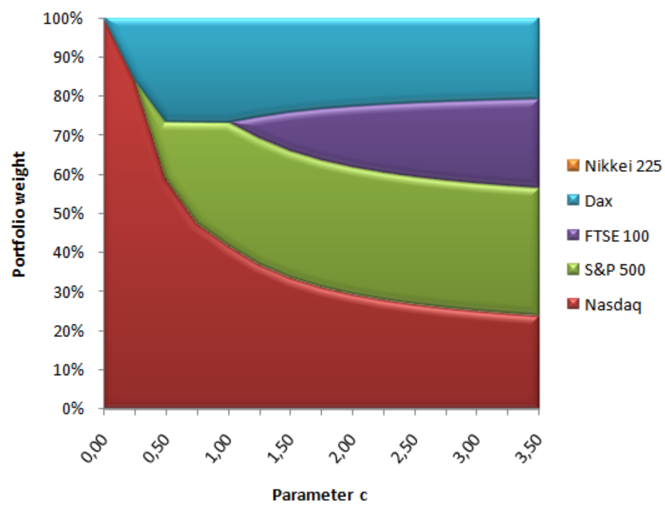

Figure 4.14: Composition of efficient portfolios of Problem 4.5

By comparing Figures 4.12-4.14 we observe that a small risk aversion parameter c in Problem 4.5 corresponds to a high risk upper bound s in Problem 4.3 and a high return lower bound in Problem 4.4. Vice versa, a large risk aversion parameter c in Problem 4.5 corresponds to a low risk upper bound s in Problem 4.3 and a low return lower bound in Problem 4.4.

We also observe that if the upper bound on the portfolio variance is high, then most of the capital is invested in Nasdaq which is the most risky index and has the highest return. If the upper bound on the portfolio variance becomes smaller the optimal portfolio weights are more diversified.

4.2.1 Implementation in MATLAB

The optimization problems are solved by using the cvx-package in MATLAB 7.8.0.347 (R2009a). The efficient frontier of Problem 4.3 is obtained by solving 21 optimization problems of that type for different values s. The corresponding Matlab code is given below.

```
j=1
for i=min(min(Covres)):(max(max(Covres))-min(min(Covres)))/20:max(max(Covres))
cvx_begin
    variable w(n,1);
    maximize (meanres'*w)
    subject to
    ones(1,n)*w == 1;
    w >= 0;
    w'*Covres*w <= i;
cvx_end
Result(j,:)=[i cvx_optval];
j=j+1;
disp(w);
end
x=Result(:,1);
y=Result(:,2);
plot(x,y);
```

`Covres` is the 50% quantile of the resampled covariance matrix. The index i denotes the 20 steps in the interval $[s_{min}, s_{max}]$. In `Result` the index i and the corresponding optimal returns are summarized. Also the optimal weights are displayed (`disp(w)`). Finally we plot the efficient frontier, the values of i (`x=Result(:,1)`) with the corresponding optimal returns (`y=Result(:,2)`). Analogously, we implement Problem 4.4 and 4.5 and their robust counterparts.

4.3 Comparison: Classical optimization problems with the robust counterparts

In this section we compare the classical mean-variance optimization problems with their robust counterparts.

4.3.1 Uncertainty in the expected returns

First we consider Problem 4.3. For the robust counterpart, we assume that the expected returns are uncertain and lie in the following uncertainty set:

$$\mathbb{U}(\mu_r) = \left\{\mu_r = (\mu_i) : \mu_i^L \leq \mu_i \leq \mu_i^U,\ i = 1, \ldots, n\right\} \tag{4.6}$$

with the lower and upper bounds μ_i^L and μ_i^U, $\forall i$.

The formulation of the robust counterpart of Problem 4.3 is

$$\max_{w} \min_{\mu_r \in \mathbb{U}(\mu_r)} \mu_r' w \tag{4.7}$$

$$\begin{aligned} s.t.\ & e'w = 1 \\ & w'\Sigma w \leq s \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

See also Problem 3.1, Section 3.2.1.

The worst case for the expected returns is $\mu_r^L = (\mu_1^L, \ldots, \mu_n^L)$ and we get the following robust optimization problem:

$$\max_{w} (\mu_r^L)' w \tag{4.8}$$

$$\begin{aligned} s.t.\ & e'w = 1 \\ & w'\Sigma w \leq s \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

By bootstrapping from the historical data as described in the beginning of this chapter we get the following lower bounds of the expected returns as 25% quantiles of the sampled data

$$\mu_r^L = \begin{pmatrix} 0.001131463 \\ 0.0007715919 \\ 0.0004058809 \\ 0.0007871138 \\ -0.0014519829 \end{pmatrix} \tag{4.9}$$

and the following upper bounds as 75% quantiles of the sampled data

$$\mu_r^U = \begin{pmatrix} 0.002573538 \\ 0.0017814287 \\ 0.0014111549 \\ 0.0021264350 \\ -0.0001566744 \end{pmatrix}. \tag{4.10}$$

We have solved Problem 4.3 and its robust counterpart for 21 values of parameter s lying in the interval $[s_{min}, s_{max}]$, computed as described above.

The composition of the optimal portfolios for the robust optimization Problem 4.8 is illustrated in the following graphic. The composition of portfolios belonging to the efficient frontier of the corresponding Problem 4.3 is depicted in Figure 4.12.

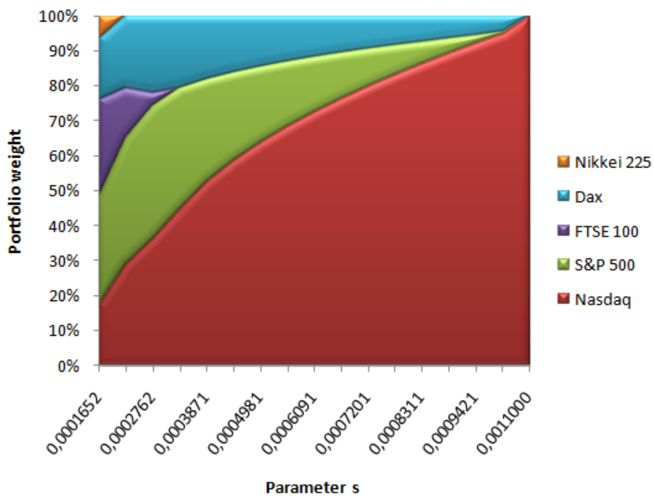

Figure 4.15: Composition of efficient portfolios for Problem 4.8

With the highest level $s = 0.0011$ for the risk, everything is invested in Nasdaq, the most risky index. If the level for the risk is smaller then the optimal weights are more diversified, but the robust weights are a little bit more concentrated in Nasdaq than the classical ones. The optimal value for the maximized reward is higher for the classical optimization problem than for the robust one, just as expected: in the robust optimization problem we use the worst case estimations for the asset returns, which clearly lead to a smaller portfolio return. We assume uncertainty in the expected returns only, so the uncertainty does not affect the feasibility of the solution and there exists no solution for a level of risk lower than 0.00016516.

The following picture shows the efficient frontiers where the blue line corresponds to the classical mean-variance optimization problem and the green one corresponds to the robust counterpart.

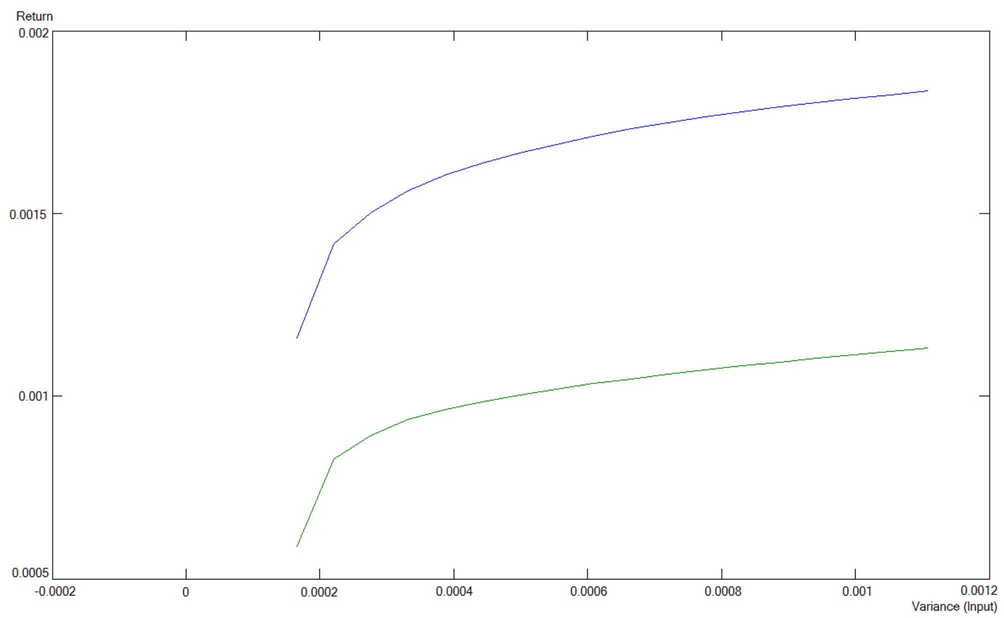

Figure 4.16: Classical and robust efficient frontiers for Problem 4.3 (blue) and Problem 4.8 (green)

We can see that the robust efficient frontier lies below the classical one which means, that the robust optimal portfolio for a certain level of risk has a smaller reward than the portfolio on the classical efficient frontier. This fact is already commented above.

The following table shows the optimal solutions for the classical optimization Problem 4.3 and its robust counterpart for some upper bounds values s on the risk.

s	classical w	optimal return	robust w	optimal return
0.00010967	No solution exists!	-Inf	No solution exists!	-Inf
0.00016516	$\begin{pmatrix} 0.1800 \\ 0.3042 \\ 0.2677 \\ 0.1793 \\ 0.0688 \end{pmatrix}$	0.00115956	$\begin{pmatrix} 0.1734 \\ 0.3108 \\ 0.2757 \\ 0.1743 \\ 0.0658 \end{pmatrix}$	0.000589611
0.00033164	$\begin{pmatrix} 0.4352 \\ 0.2970 \\ 0.0000 \\ 0.2678 \\ 0.0000 \end{pmatrix}$	0.00156249	$\begin{pmatrix} 0.4468 \\ 0.3469 \\ 0.0000 \\ 0.2063 \\ 0.0000 \end{pmatrix}$	0.000935551
0.00083109	$\begin{pmatrix} 0.8547 \\ 0.0000 \\ 0.0000 \\ 0.1453 \\ 0.0000 \end{pmatrix}$	0.00177877	$\begin{pmatrix} 0.8616 \\ 0.0623 \\ 0.0000 \\ 0.0761 \\ 0.0000 \end{pmatrix}$	0.00108284
0.00094207	$\begin{pmatrix} 0.9186 \\ 0.0000 \\ 0.0000 \\ 0.0814 \\ 0.0000 \end{pmatrix}$	0.00180333	$\begin{pmatrix} 0.9201 \\ 0.0228 \\ 0.0000 \\ 0.0571 \\ 0.0000 \end{pmatrix}$	0.00110359
0.0011	$\begin{pmatrix} 1.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \end{pmatrix}$	0.00183463	$\begin{pmatrix} 1.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \end{pmatrix}$	0.00113146

Table 4.3: Solutions of Problem 4.3 and Problem 4.8

4.3.2 Uncertainty in the expected returns and the covariance matrix: Simple case

Next we compare Problem 4.4 with the robust counterpart:

$$\min_{w} \max_{\Sigma \in \mathbb{U}(\Sigma)} w'\Sigma w \tag{4.11}$$

$$\begin{aligned} s.t.\ & e'w = 1 \\ & \min_{\mu_r \in \mathbb{U}(\mu_r)} \mu_r' w \geq m \\ & w_i \geq 0,\ \forall i \end{aligned}$$

with the uncertainty set 4.6 for the expected returns and the uncertainty set for the covariance matrix given as

$$\mathbb{U}(\Sigma) = \left\{ \Sigma = (\Sigma_{ij}) : \underline{\Sigma} \leq \Sigma \leq \overline{\Sigma},\ i, j = 1, \ldots, n \right\} \tag{4.12}$$

where the lower bound $\underline{\Sigma} = (\underline{\Sigma_{ij}})$ and the upper bound $\overline{\Sigma} = (\overline{\Sigma}_{ij})$ for the covariance matrix are obtained as 25% and 75% quantiles of the resampled data.

The lower and upper bounds $\underline{\Sigma}$ and $\overline{\Sigma}$ for the covariance matrix for all $i, j = 1, \ldots, 5$.

$$\underline{\Sigma} = \begin{pmatrix} 0.001035060 & -0.000017004 & -0.000017262 & -0.000022645 & -0.000021411 \\ -0.000017004 & 0.000524554 & -0.000011390 & -0.000015934 & -0.000014725 \\ -0.000017262 & -0.000011390 & 0.000523608 & -0.000016311 & -0.000015499 \\ -0.000022645 & -0.000015934 & -0.000016311 & 0.000949508 & -0.000021178 \\ -0.000021411 & -0.000014725 & -0.000015499 & -0.000021178 & 0.000892139 \end{pmatrix} \tag{4.13}$$

$$\overline{\Sigma} = \begin{pmatrix} 0.0011901086 & 0.0000168514 & 0.0000164742 & 0.0000222886 & 0.0000223994 \\ 0.0000168514 & 0.0005993715 & 0.0000121208 & 0.0000152054 & 0.0000155135 \\ 0.0000164742 & 0.0000121208 & 0.0006138156 & 0.0000160002 & 0.0000162153 \\ 0.0000222886 & 0.0000152054 & 0.0000160002 & 0.0010674866 & 0.0000190276 \\ 0.0000223994 & 0.0000155135 & 0.0000162153 & 0.0000190276 & 0.0010142965 \end{pmatrix} \tag{4.14}$$

These matrices $\underline{\Sigma}$ and $\overline{\Sigma}$ are positive definite. So we do not need an approximation.

If we optimize with the worst case of the covariance matrix, which is $\overline{\Sigma}$, we get the following robust optimization problem:

$$\begin{aligned} \min_w \, & w'\overline{\Sigma}w \\ s.t. \; & e'w = 1 \\ & \mu_r'w \geq m \\ & w_i \geq 0, \; \forall i \end{aligned} \tag{4.15}$$

This problem is of the same type as a classical variance minimization problem, and can be solved by standard methods, see also Problem 3.14, Section 3.2.2. We solve this problem for 20 different values of parameter m in the interval $[-0.00078726, 0.0018]$ as described in Section 4.2.

The composition of the efficient portfolios of Problem 4.4 is depicted in Figure 4.13. The allocation of the weights of its robust counterpart, Problem 4.11 is illustrated in the following graphic.

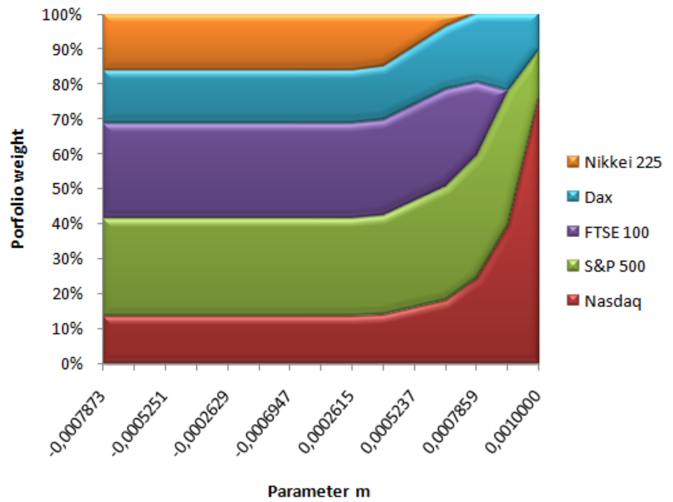

Figure 4.17: Composition of efficient portfolios for Problem 4.11

If we compare Figure 4.13 and Figure 4.17 and consider for example the level of $m = 0.001$, we can observe, that the efficient portfolio of the robust mean-variance optimization problem is less diversified, because we invest in Nasdaq, S&P 500 and Dax only, whereas in the efficient portfolio of the classical mean-variance optimization problem we invest in all indices.

The results of the classical and the robust optimization problems are listed in the following table.

m	classical w	optimal risk	robust w	optimal risk
-0.00078726	$\begin{pmatrix}0.1386\\0.2742\\0.2719\\0.1533\\0.1620\end{pmatrix}$	0.000153829	$\begin{pmatrix}0.1356\\0.2791\\0.2721\\0.1526\\0.1606\end{pmatrix}$	0.000177634
0.00078587	$\begin{pmatrix}0.1385\\0.2742\\0.2720\\0.1533\\0.1620\end{pmatrix}$	0.000153829	$\begin{pmatrix}0.2423\\0.3528\\0.2077\\0.1972\\0.0000\end{pmatrix}$	0.00022433
0.00091697	$\begin{pmatrix}0.1386\\0.2742\\0.2719\\0.1533\\0.1620\end{pmatrix}$	0.000153829	$\begin{pmatrix}0.3945\\0.3843\\0.0000\\0.2212\\0.0000\end{pmatrix}$	0.00033752
0.0010	$\begin{pmatrix}0.1592\\0.2890\\0.2698\\0.1662\\0.1158\end{pmatrix}$	0.00015662	$\begin{pmatrix}0.7638\\0.1329\\0.0000\\0.1033\\0.0000\end{pmatrix}$	0.000723623
0.0012	$\begin{pmatrix}0.1837\\0.3068\\0.2675\\0.1815\\0.0605\end{pmatrix}$	0.000167259	No solution exists!	+Inf

Table 4.4: Solutions of Problem 4.4 and Problem 4.11

By comparing Table 4.4 and Table 4.3 we can see that the differences between the robust and the classical efficient portfolios are more pronounced in this case when we take into account both the uncertainty in expected returns and also in the covariance matrix. Also in this case we observe that the optimal weights of the robust optimization problem are more concentrated on certain indices while the weights of the classical portfolio optimization problem are more diversified. If the level of the prescribed reward is low, we invest in all of the five indices with the most percentages in S&P 500 and FTSE because they have the smallest risk. From a level $m = 0.0012$ upwards there exists no feasible solution for the robust problem because we assume that the worst case for the expected returns occurs. If we compare the optimal values we can see that the minimized risk of the classical problem is lower than its robust counterpart. This is due to the worst case assumption for the covariance matrix. We can see these the efficient frontiers of the robust and classical problem in the following picture.

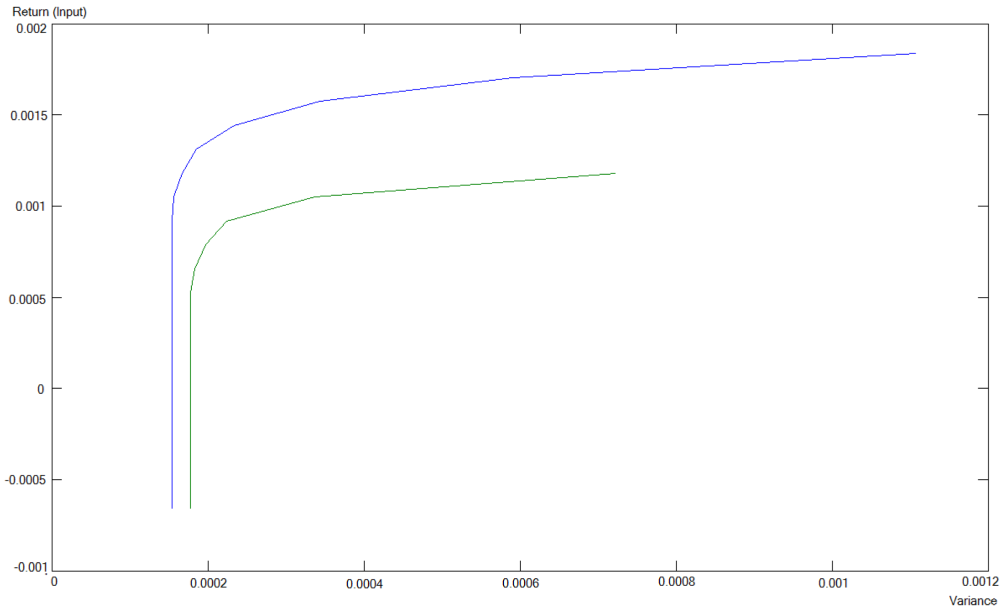

Figure 4.18: Classical and robust efficient frontiers for Problem 4.4 (blue) and Problem 4.11 (green)

The efficient frontier of the robust optimization problem is below and to the right of the classical one.

4.3.3 Uncertainty in the expected returns and the covariance matrix: General case

In our previous chapter we explained that in general lower bounds $\underline{\Sigma}_{ij}$ for the covariances of asset i and j, $1 \leq i, j \leq 5$, do not yield a positive definite matrix $(\underline{\Sigma})_{1 \leq i,j \leq 5}$ and hence $\underline{\Sigma}$ cannot be used as a worst case approximation for the covariance matrix Σ. Therefore we consider the robust risk-adjusted optimization problem with the general case for the uncertainty set of the covariance matrix, i.e., the robust counterpart of Problem 4.5.

$$\max_{w} \left\{ \min_{\mu_r \in \mathbb{U}(\mu_r)} \mu_r' w - c \max_{\Sigma \in \mathbb{U}(\Sigma)} w' \Sigma w \right\} \tag{4.16}$$

$$\begin{aligned} s.t.\ & e'w = 1 \\ & w_i \geq 0,\ \forall i. \end{aligned}$$

Consider the uncertainty set 4.6 for the expected returns, the uncertainty set 4.12 for the covariance matrix and the dual variables $\underline{\Lambda}$ and $\overline{\Lambda}$ corresponding to the restrictions $\underline{\Sigma} \leq \Sigma$ and $\Sigma \leq \overline{\Sigma}$. As shown in Section 3.2.2. (Problem 3.22) the robust optimization Problem 4.16 is equivalent to the following problem.

$$\max_{w, \underline{\Lambda}, \overline{\Lambda}} \left\{ (\mu_r^L)' w - c(\langle \overline{\Lambda}, \overline{\Sigma} \rangle - \langle \underline{\Lambda}, \underline{\Sigma} \rangle) \right\} \tag{4.17}$$

$$\begin{aligned} s.t.\ & e'w = 1 \\ & w_i \geq 0,\ \forall i \\ & \overline{\Lambda} \geq 0,\ \underline{\Lambda} \geq 0 \\ & \begin{pmatrix} \overline{\Lambda} - \underline{\Lambda} & w \\ w' & 1 \end{pmatrix} \succeq 0. \end{aligned}$$

We have solved this optimization problem by using the cvx-package of MATLAB 7.8.0.347 (R2009).

For the optimal weights of Problem 4.5, see Figure 4.14, for the composition of efficient portfolios of Problem 4.17 see the following graphic.

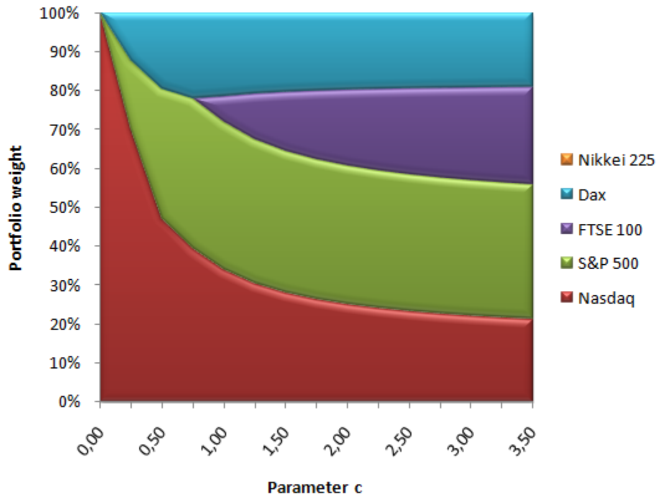

Figure 4.19: Composition of efficient portfolios for Problem 4.17

We consider the efficient frontier of Problem 4.5 and 4.17.

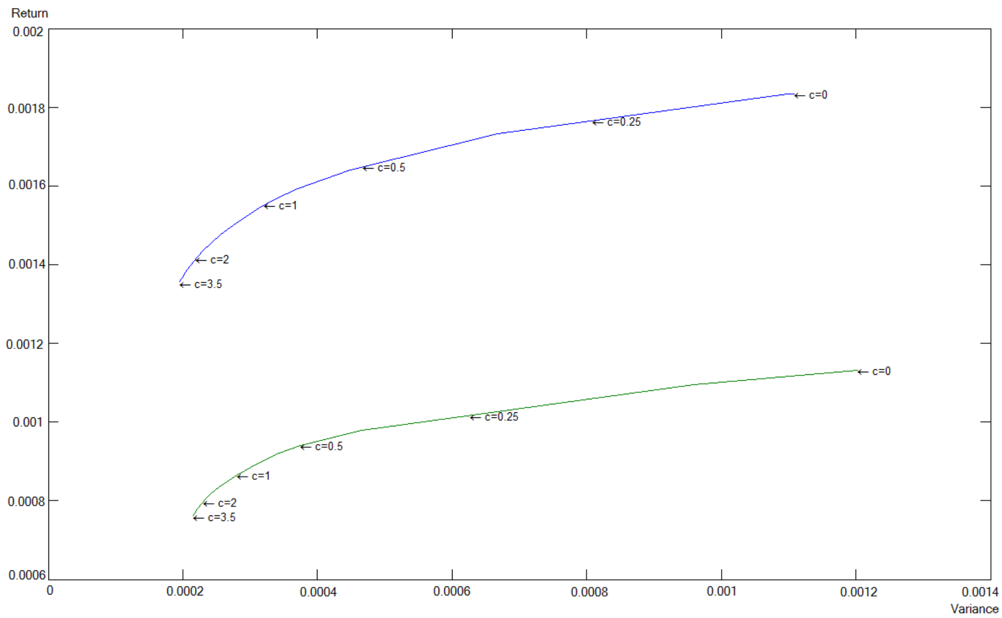

Figure 4.20: Classical and robust efficient frontiers for Problem 4.5 (blue) and Problem 4.17 (green)

The robust efficient frontier of the general case lies below the classical one, and this is similar to pictures we have already seen when comparing the robust and classical counterpart for other models.

The following table shows the optimal weights of Problem 4.5 and 4.17 with the return and risk of each efficient portfolio.

c	classical w	return	risk	robust w	return	risk
0.00	$\begin{pmatrix}1.0000\\0.0000\\0.0000\\0.0000\\0.0000\end{pmatrix}$	0.001834631	0.001108555	$\begin{pmatrix}1.0000\\0.0000\\0.0000\\0.0000\\0.0000\end{pmatrix}$	0.001131463	0.001201982
0.25	$\begin{pmatrix}0.8397\\0.0000\\0.0000\\0.1603\\0.0000\end{pmatrix}$	0.001773003	0.00080745	$\begin{pmatrix}0.7002\\0.1761\\0.0000\\0.1237\\0.0000\end{pmatrix}$	0.001025571	0.00062708
0.75	$\begin{pmatrix}0.4735\\0.2592\\0.0000\\0.2673\\0.0000\end{pmatrix}$	0.001584611	0.00035819	$\begin{pmatrix}0.3941\\0.3846\\0.0000\\0.2213\\0.0000\end{pmatrix}$	0.00091685	0.00033737
1.50	$\begin{pmatrix}0.3345\\0.3234\\0.1005\\0.2416\\0.0000\end{pmatrix}$	0.001437368	0.00024715	$\begin{pmatrix}0.2807\\0.3631\\0.1517\\0.2045\\0.0000\end{pmatrix}$	0.00082033	0.00024357
3.50	$\begin{pmatrix}0.2379\\0.3256\\0.2286\\0.2079\\0.0000\end{pmatrix}$	0.001353618	0.00019522	$\begin{pmatrix}0.2131\\0.3452\\0.2500\\0.1917\\0.0000\end{pmatrix}$	0.00075984	0.00021476

Table 4.5: Solutions of Problem 4.5 and Problem 4.17

We can see that the optimal robust portfolio is mainly invested in Nasdaq and a little bit in S&P 500 while the optimal classical portfolio invests even more in Nasdaq. Nothing is invested in Nikkei 225 because of its negative expected return. If the risk-aversion parameter c is small, the optimal robust and classical portfolios invest most of the capital in Nasdaq. The weights get more diversified while c increases.

4.3.4 Stability of the optimization problems

The goal of the following numerical experiments is the investigation of the stability of classical and robust optimization problems.
In practice, it can often happen that we have wrong estimates for the expected returns and covariance matrices. Since small fluctuations in the input parameters influence the solutions of the problem, wrong estimations would probably lead to bad solutions. In this context the robust portfolio optimization might be an alternative to classical mean-variance optimization. To illustrate this alternative, we discuss the solution of three types of problems: the classical mean-variance optimization, the robust mean-variance optimization with uncertainty in the asset returns and the robust mean-variance optimization with uncertainty in the asset returns and in the covariance matrix (simple case) as described in Section 4.3.2. The optimal portfolios are denoted by w_c, w_{R1} and w_{R2}, respectively. They are summarized in the following table.

w_c	w_{R1}	w_{R2}
0.7026	0.7148	0.6787
0.0776	0.2319	0.2543
0.0000	0.0000	0.0000
0.2198	0.0532	0.0671
0.0000	0.0000	0.0000

Table 4.6: Optimal weights of Problem 4.3 and its robust counterparts

Let us consider the performance of these three portfolios for real values μ_0 and Σ_0 of expected returns and covariance matrix. These values are in general not equal to the estimators. We compare the returns $\mu_0' w_c$, $\mu_0' w_{R1}$ and $\mu_0' w_{R2}$ of the classical and robust optimal portfolios, respectively. Analogously, we compare the variances $w_c' \Sigma_0 w_c$, $w_{R1}' \Sigma w_{R1}$ and $w_{R2}' \Sigma_0 w_{R2}$ of the classical and robust optimal portfolios. If the robust optimization problem gets a higher reward and a lower risk, then the robust optimization would be an alternative to the classical optimization model in this case.

Let us consider different return vectors and covariance matrices from the corresponding uncertainty sets as representatives of real values μ_0 and Σ_0. We pick up 20 vectors μ_i, $1 \leq i \leq 20$, randomly and uniformly distributed over the (μ_r^L, μ_r^U) given in 4.9, 4.10 where the selection is done independently for each index. Analogously, we select $\sigma_{ij} \in (\underline{\Sigma}_{ij}, \overline{\Sigma}_{ij})$ randomly and uniformly distributed for $i, j = 1, \ldots, 5$ and repeat this process 20 times to generate the representatives of the real covariance matrix. The matrices (σ_{ij}) are approximated by positive definite matrices as described in Section 4.1, if necessary. The results of this experiment are shown in Table 4.7.

reward			risk		
$\mu_i' w_c$	$\mu_i' w_{R1}$	$\mu_i' w_{R2}$	$w_c' \Sigma_i w_c$	$w_{R1}' \Sigma_i w_{R1}$	$w_{R2}' \Sigma_i w_{R2}$
0.001061001	**0.001073631**	0.001058443	0.0005511491	0.0005521375	**0.0005066374**
0.001077957	**0.001090796**	0.001075445	0.0005555832	0.0005563026	**0.0005105772**
0.001094931	**0.001105641**	0.001090337	0.0005595174	0.0005602688	**0.0005143530**
0.001106748	**0.001117025**	0.001101646	0.0005629032	0.0005635259	**0.0005174668**
0.001121509	**0.001129687**	0.001114333	0.0005708408	0.0005711929	**0.0005247543**
0.001133438	**0.001141012**	0.001125579	0.0005792811	0.0005796031	**0.0005326798**
0.001149187	**0.001155050**	0.001139597	0.0005809381	0.0005812734	**0.0005342648**
0.001178473	**0.001183283**	0.001167751	0.0005852538	0.0005855010	**0.0005382327**
0.001216752	**0.001219548**	0.001203668	0.0006000069	0.0005999404	**0.0005518912**
0.001245978	**0.001246654**	0.001230833	0.0006034154	0.0006032613	**0.0005550638**
0.001275651	**0.001277930**	0.001261622	0.0006067427	0.0006065504	**0.0005581928**
0.001300774	**0.001301317**	0.001285071	0.0006162982	0.0006160385	**0.0005671602**
0.001403578	0.001400707	0.001383792	0.0006180100	0.0006176606	**0.0005686996**
0.001479743	0.001471578	0.001454361	0.0006216076	0.0006211292	**0.0005720007**
0.001733949	0.001716704	0.001697086	0.0006277695	0.0006271607	**0.0005777233**
0.002060558	0.002025411	0.002004609	0.0006342732	0.0006336522	**0.0005838726**
0.002160838	0.002123019	0.002102007	0.0006381557	0.0006374667	**0.0005874921**
0.002268360	0.002226652	0.002204519	0.0006399795	0.0006393226	**0.0005892473**
0.002301245	0.002259318	0.002236671	0.0006456496	0.0006449328	**0.0005945041**
0.002365520	0.002320049	0.002297019	0.0006517101	0.0006508782	**0.0006000738**

Table 4.7: Comparison of return and risk for classical and robust optimal portfolios

In 60% of the considered cases, the portfolio w_{R1} has the highest reward. When the "real" returns are high, then portfolio w_c performs better in terms of total return. This behavior is understandable and plausible because portfolio w_c maximizes the total return assuming that the expected assets returns are equal to the 50% quantiles of the bootstrapped values. By comparing the risk of portfolio w_c and w_{R1}, we can see that in 12 of 20 cases portfolio w_{R1} has a lower risk. So we can conclude that in most of the cases the robust optimal portfolio w_{R1} outperforms the classical optimal portfolio w_c. The reward of portfolio w_{R2} is the smallest one. In contrast, portfolio w_{R2} shows the lowest risk in all cases. Summarizing we can conlude that the robust portfolio w_{R1} represent a good trade off between low risk and high return.

4.4 Conclusion

In this chapter we applied the models of classical and robust portfolio optimization presented in the previous chapters to a portfolio of 5 assets: Nasdaq Composite, S&P 500, FTSE 100, Dax and Nikkei 225. The test bed and the analysis is similar to the computational study of Tütüncü and Koenig [10]. They also compare the classical mean-variance optimization problems with their robust counterparts.

For the estimation of the expected returns and covariances, we apply a bootstrapping approach similarly as Tütüncü and Koenig do. We use the historical data for bootstrapping and compute the sampled means and sampled covariances for the time series. We repeat this procedure 3000 times, compute the 50%, 25% and 75% quantiles and get the estimations of expected returns and covariance matrix as well as the corresponding lower and upper bounds for the uncertainty sets 4.6 and 4.12.

When comparing the efficient frontiers of the robust and classical optimization problems, we observe that the robust efficient frontier lies below the efficient frontier of the classical Markowitz optimization. The reason is clear: the robust optimization model takes into account the worst case realization of expected returns and covariance matrix. The terms of the composition of optimal robust and classical portfolios we observe that in both models most of the capital is investd in Nasdaq, if the upper bound on the variance or the lower bound on the asset return is high. Especially in the optimization problem of minimizing the risk, the classical mean-variance optimal portfolio is more diversified than the optimal robust portfolio.

Another result is that the robust optimal portfolio is more stable in the following sense. We compare the reward and the variance of two optimal robust portfolios and a classical optimal portfolio for different realizations of return and covariance from the corresponding uncertainty sets. In 60% of the considered cases one of the robust portfolios performs better than the classical portfolio in terms of reward and/or variance.

Summarizing, our tests confirm that the robust portfolio optimization is a valuable alternative to the classical mean-variance optimization at least for risk-aware investors.

Bibliography

[1] Marc C. Steinbach:
Markowitz Revisited: Single-Period and Multi-Period Mean-Variance Models,
Society for Industrial and Applied Mathematics,
Siam Review, 2001, vol. 43 (1), pp. 31-85

[2] Andreas Niedermayer, Daniel Niedermayer:
Applying Markowitz's Critical Line Algorithm,
Discussion Paper, University of Bern, Department of Economics, 2006, dp0602

[3] Rainer E. Burkard:
Einführung in die mathematische Optimierung
Lecture notes, summer term 2007, TU Graz

[4] G. Cornuejols, R. Tütüncü:
Optimization Methods in Finance.
Cambridge University Press, 2007

[5] Frank J. Fabozzi, Petter N. Kolm, Dessislava Pachamanova:
Robust Portfolio Optimization and Management
John Wiley and Sons Inc, 2007

[6] Wei Jiao:
Portfolio Resampling and Efficiency Issues
Master thesis, Institute of Statistics and Economics, 2003

[7] Amit K. Sanyal, Teeyoung Lee, Melvin Leok, N. Harris McClamroch:
Global optimum attitude estimation using uncertainty ellipsoids,
Systems and Control Letters, 2008, vol. 57, pp. 236-245

[8] Lieven Vandenberghe, Stephen Boyd:
Semidefinite Programming,
Siam Rewiew, 1996, vol. 38, pp. 49-95

[9] Dimitris Bertsimas, Dessislava Pachamanova:
Robust multiperiod portfolio management in the presence of transaction costs,
Computers and Operations Research, 2008, vol. 35, pp. 3-17

[10] R.H.Tütüncü, M. Koenig:
Robust Asset Allocation,
Annals of Operations Research, Kluwer Academic Publisher, 2004,
vol. 132, pp. 157-187