FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

MASTER THESIS

Bc. Karel Kozmík

Robust approaches in portfolio optimization with stochastic dominance

Department of Probability and Mathematical Statistics

Supervisor of the master thesis:	doc. RNDr. Ing. Miloš Kopa, Ph.D.
Study programme:	Mathematics
Study branch:	Probability, Mathematical Statistics and Econometrics

Prague 2019

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In Prague on .... Karel Kozmík

I would like to thank doc. RNDr. Ing. Miloš Kopa, Ph.D. for introducing me the topic of portfolio optimization with stochastic dominance constraints and for always being ready to help me whenever I was stuck and providing ideas for possible ways to solve the problem.

Title: Robust approaches in portfolio optimization with stochastic dominance

Author: Bc. Karel Kozmík

Department: Department of Probability and Mathematical Statistics

Supervisor: doc. RNDr. Ing. Miloš Kopa, Ph.D., Department of Probability and Mathematical Statistics

Abstract: We use modern approach of stochastic dominance in portfolio optimization, where we want the portfolio to dominate a benchmark. Since the distribution of returns is often just estimated from data, we look for the worst distribution that differs from empirical distribution at maximum by a predefined value. First, we define in what sense the distribution is the worst for the first and second order stochastic dominance. For the second order stochastic dominance, we use two different formulations for the worst case. We derive the robust stochastic dominance test for all the mentioned approaches and find the worst case distribution as the optimal solution of a non-linear maximization problem. Then we derive programs to maximize an objective function over the weights of the portfolio with robust stochastic dominance in constraints. We consider robustness either in returns or in probabilities for both the first and the second order stochastic dominance. To the best of our knowledge nobody was able to derive such program before. We apply all the derived optimization programs to real life data, specifically to returns of assets captured by Dow Jones Industrial Average, and we analyze the problems in detail using optimal solutions of the optimization programs with multiple setups. The portfolios calculated using robustness in returns turned out to outperform the classical approach without robustness in an out-of-sample analysis.

Keywords: stochastic dominance, robustness, portfolio optimization

Contents

Introduction

The problem of portfolio optimization is a typical problem in economics and finance. The classical approach is based on models considering the trade-off between mean and risk . The mean-risk models (defined and analyzed for example in Kozmík [2010]) try to find efficient portfolios that maximize mean given a maximum risk or minimize risk given a minimum mean. Some of the models also combine both mean and risk into performance ratios, which help to select one portfolio out of the set of efficient portfolios.

Modern methods that deal with portfolio optimization include stochastic dominance. The concept of stochastic dominance allows us to compare two random variables, which in this case represent the return of our final portfolio and a benchmark portfolio. In this work we explore the resistance of the optimal portfolio to the changes of distribution of the returns. To get the optimal portfolio, historical observations are usually used, which represent the empirical distribution. We analyze, what is the worst distribution in the neighborhood of the empirical distribution for the optimal portfolio achieved by portfolio optimization with first and second stochastic dominance constraints. We use the Wasserstein distance (defined and used for optimization purposes for example in Pflug and Pichler [2014]) to measure the distance between two distributions and we also define in what sense the distribution is the worst in two different ways for the second order stochastic dominance and compare the results. It is important to know which distribution violates the stochastic dominance constraints the most, because if that would be the true distribution it might require re-balancing of the portfolio. We also derive ways how to achieve portfolios that maximize an objective function and robustly dominate a benchmark portfolio when considering robustness either in returns or in probabilities for both the first order stochastic dominance and the second order stochastic dominance.

We follow Kuosmanen [2004], where the basics of portfolio optimization with stochastic dominance constraints are stated for the empirical distribution. Then we also use the work of Luedtke [2008], where the problem was generalized for distributions with atoms that do not have the same probability. The topic of stochastic dominance was also explored in Dentcheva and Ruszczyński [2006], from which we use especially the formulation for portfolio optimization program with second order stochastic dominance in constraints. Robust optimization with second order stochastic dominance constraints is then discussed in Dentcheva and Ruszczyński [2010]. Further, the relation between first and second order stochastic dominance was explored in Dentcheva and Ruszczynski [2004] and portfolio optimization with third order stochastic dominance was considered in Post and Kopa [2017].

In chapter one, we introduce risk and the concept of stochastic dominance and state the programs for portfolio optimization with stochastic dominance of the first, the second and the third order in constraints. In the second chapter, we introduce the distributionally robust optimization and the Wasserstein distance. We define the worst case distribution for the first order stochastic dominance and we define the worst case distribution for the second order stochastic dominance in two different ways. We derive tests for robust stochastic dominance using the definitions of the worst case distribution and find the worst case distributions as a solution of an optimization problem. Finally, we solve distributionally robust problems with robust stochastic dominance in constraints. We derive programs for the first and the second order when considering robustness either in returns or in probabilities. In chapter three, we apply the derived optimization programs on stock returns data from Dow Jones Industrial Average and discuss the results. We present multiple graphs to provide insight to the behavior of the worst case distributions and we present portfolio weights for the robust stochastic dominance problems. The portfolios are tested on out-of-sample observations.

1. Risk and stochastic dominance

1.1 Risk, risk measures and mean-risk models

1.1.1 Definition of risk

Let us have a portfolio of assets, according to Levy [2015] we define risky position or risky asset as a situation in which there is more than one financial outcome. That means that for at least one of the assets, the distribution of outcome is not degenerated. On the contrary to uncertainty, when we do not know the full distribution, risk is defined as the situation in which we know all the possible outcomes and the distribution of the outcomes.

1.1.2 Mean-risk models

Let us have random variable R denoting the returns of an investment. In mean-risk models, we take into consideration both the mean ($\mathsf{E}[R]$) and risk, let us denote it $r(R)$, which denotes some risk measure (we list some of the possible choices in the next section). We define the classical mean-risk model in accordance with Kozmík [2010]. Let us have a portfolio of N stocks with weights $\boldsymbol{w} = (w_1, .., w_N)$, where $\sum_{i=1}^{N} w_i = 1$. Let us have random variable stating the simple return of each stock $R_1, .., R_N$. The random return of the whole portfolio is then:

$$R(\boldsymbol{w}) = \sum_{i=1}^{N} w_i R_i.$$

Let us denote $u_{\boldsymbol{w}} = \mathsf{E}\,(R(\boldsymbol{w}))$ and $r_{\boldsymbol{w}} = r(R(\boldsymbol{w}))$. Then we can define an optimization problem to find an efficient portfolio:

$$\begin{aligned} &\min_{\boldsymbol{w}} \; r_{\boldsymbol{w}} \\ \text{subject to } \; & u_{\boldsymbol{w}} \geq u_e \\ & \sum_{i=1}^{N} w_i = 1 \\ & w_i \in \mathbb{R}, \; i = 1, .., N, \end{aligned} \tag{1.1}$$

where u_e represents benchmark return or any pre-specified return selected by the investor.

Sometimes we add another restrictions that do not allow short selling

$$w_1, .., w_N \geq 0.$$

The restrictions can be modified for example to

$$w_1, .., w_N \geq -1$$

to limit the exposure in every asset (or bound sums of groups of w_i to limit exposure to a group of assets) and also to make the problem bounded.

1.1.3 Measures of risk

To measure risk the investor is exposed to, several risk measures were introduced, we mention some of the most used (that can be used in the optimization model presented above). Let us have a portfolio $\boldsymbol{w}$ and let $R(\boldsymbol{w})$ be a random variable denoting its return, $L(\boldsymbol{w}) = -R(\boldsymbol{w})$ will denote the loss of the portfolio.

One of the first, which was used by Markowitz (Markowitz [1952]), and most straightforward measure of risk is the variance.

Definition 1. *Let $R(\boldsymbol{w})$ be a random variable denoting the return of a portfolio, then:*

$$var(R(\boldsymbol{w})) = \mathsf{E}\left(R(\boldsymbol{w}) - \mathsf{E}[R(\boldsymbol{w})]\right)^2.$$

If we do not want to give higher weight to observations further from the expected value, we can use the absolute value instead of the square.

Definition 2. *Let $L(\boldsymbol{w})$ be a random variable denoting the loss of a portfolio, we define absolute deviation as:*

$$r_m(L(\boldsymbol{w})) = \mathsf{E}\left|L(\boldsymbol{w}) - \mathsf{E}\, L(\boldsymbol{w})\right|.$$

Because investor is usually worried only about the loss and unexpected gains are welcome, to reflect this attribute, we define semivariance.

Definition 3. *Let $L(\boldsymbol{w})$ be a random variable denoting the loss of a portfolio, we define semivariance as:*

$$r_s(L(\boldsymbol{w})) = \mathsf{E}\left[\max\left(0, L(\boldsymbol{w}) - \mathsf{E}\, L(\boldsymbol{w})\right)^2\right].$$

Investors and financial institutions often care about the left tail of the distribution of returns, the highest losses. For this purpose, more complex risk measures VaR (Value at Risk) and $CVaR$ (Conditional Value at Risk) were introduced.

Definition 4. *Let $\alpha \in (0,1)$ and let $L(\boldsymbol{w})$ be a random variable denoting the loss of a portfolio. Then we define VaR_α as:*

$$VaR_\alpha(L(\boldsymbol{w})) = \inf\left\{l \in \mathbb{R}, \mathsf{P}\left(L(\boldsymbol{w}) > l\right) \leq 1 - \alpha\right\}$$

In the definition α represents the confidence level, $VaR_\alpha(L(\boldsymbol{w}))$ represent the maximum loss on confidence level α. We can see that this definition does not take into account the rightmost tail of the distribution. To include also the information about the right tail of losses (highest losses), $CVaR$ was introduced. We use the properties of $CVar$ proven in Rockafellar and Uryasev [2002].

Definition 5. *Let $\alpha \in (0,1)$ and let $L(\boldsymbol{w})$ be a random variable denoting the loss of a portfolio. Then we define $CVaR_\alpha$ as:*

$$CVaR_\alpha(L(\boldsymbol{w})) = min\left\{a \in \mathbb{R}, a + \frac{1}{1-\alpha}\,\mathsf{E}\left[\max\left(0, L(\boldsymbol{w}) - a\right)\right]\right\}.$$

For continuous distributions, according to , $CVaR$ can also be equivalently defined as :

$$\begin{aligned} CVaR_\alpha(L(\boldsymbol{w})) &= \mathsf{E}\left[L(\boldsymbol{w})\,|L(\boldsymbol{w}) > VaR_\alpha(L(\boldsymbol{w}))\right] \\ &= \mathsf{E}\left[L(\boldsymbol{w})\,|L(\boldsymbol{w}) \geq VaR_\alpha(L(\boldsymbol{w}))\right]. \end{aligned} \tag{1.2}$$

The latter definition shows us that $CVaR$ is actually the expected loss in the case that the loss is higher than VaR.

If there is a probability atom at $VaR_\alpha(L(\boldsymbol{w}))$ (which is true for example for discrete distributions), then

$$\mathsf{E}\left[L(\boldsymbol{w})\,|L(\boldsymbol{w}) > VaR_\alpha(L(\boldsymbol{w}))\right] \neq \mathsf{E}\left[L(\boldsymbol{w})\,|L(\boldsymbol{w}) \geq VaR_\alpha(L(\boldsymbol{w}))\right].$$

In this case we can use the weighted average formula for $CVaR$. Let $F(L(\boldsymbol{w}), x)$ denote the distribution function of $L(\boldsymbol{w})$ in point x. Let us define $\lambda_\alpha(L(\boldsymbol{w})) \in [0,1]$ as:

$$\lambda_\alpha(L(\boldsymbol{w})) = \left[F(L(\boldsymbol{w}), VaR_\alpha(L(\boldsymbol{w}))) - \alpha\right]/[1-\alpha].$$

If $F(L(\boldsymbol{w}), VaR_\alpha(L(\boldsymbol{w})) < 1$ (there can be also greater loss than $VaR_\alpha(L(\boldsymbol{w}))$), then

$$\begin{aligned} &CVaR_\alpha(L(\boldsymbol{w})) = \\ &\quad \lambda_\alpha(L(\boldsymbol{w}))VaR_\alpha(L(\boldsymbol{w})) + (1-\lambda_\alpha(L(\boldsymbol{w})))\,\mathsf{E}\left[L(\boldsymbol{w})\,|L(\boldsymbol{w}) > VaR_\alpha(L(\boldsymbol{w}))\right] \end{aligned} \tag{1.3}$$

and if $F(L(\boldsymbol{w}), VaR_\alpha(L(\boldsymbol{w})) = 1$ (no higher loss can be realized), then

$$CVaR_\alpha(L(\boldsymbol{w})) = VaR_\alpha(L(\boldsymbol{w})).$$

1.2 Stochastic dominance

1.2.1 Motivation

Let $u : S \to \mathbb{R}, S \subseteq \mathbb{R}$ be a utility function, we suppose the utility function is continuous and non-decreasing. Let us suppose that investor's preferences can be represented by a utility function u and he makes choices based on the expected utility from the investment $\mathsf{E}\,u(R)$, where random variable R denotes the returns of the investment. Then for a client with a given utility function, we are able to order possible portfolios based on their expected utility. But when we do not know the utility function or we want the portfolio to be acceptable for a large amount of people (for example pension funds etc.), we can use the concept of stochastic dominance.

In this work, we often talk about random returns, which are often represented by a random variable and about observed returns, which are represented by scenarios. It should be clear from the context, which one we mean and we try to emphasize the difference using random when talking about random variable. Most of the work uses scenarios so we use just simply returns when we talk about observed returns or scenarios.

1.2.2 Definition of stochastic dominance

The fact that utility function is non-decreasing represent our assumption that every investor wants more money rather than less. Further, we distinguish investors based on their attitude to risk.

Definition 6. *Let u be the utility function of the investor and let W represent current wealth of the investor. If:*

- $\mathsf{E}\, u(W+R) < u(W+\mathsf{E}\, R)$ *for all suitable random variables R, we say that the investor is risk averse,*
- $\mathsf{E}\, u(W+R) = u(W+\mathsf{E}\, R)$ *for all suitable random variables R, we say that the investor is risk neutral,*
- $\mathsf{E}\, u(W+R) > u(W+\mathsf{E}\, R)$ *for all suitable random variables R, we say that the investor is risk seeking,*

where R is a suitable random variable if both expectations exist.

R in the previous definition usually represents random returns from an investment opportunity.

Most of the investors are risk averse, which corresponds to the second order stochastic dominance, which we define later. The utility functions of risk averse investors are concave, which means that every additional unit of wealth gives less utility than the previous one, which is supported by empirical evidence.

Further on, we will assume function u to be sufficiently smooth and random variable R to have finite moments to a sufficient order.

Now we define the first order stochastic dominance (FSD) and the second order stochastic dominance (SSD) in accordance with Levy [2015].

Definition 7. *First we define sets $\mathcal{U}$.*

- *We define $\mathcal{U}_1$ as the set of all utility functions*
- *We define $\mathcal{U}_2$ as the set of all concave utility functions*
- *We define higher orders $\mathcal{U}_n, n \geq 2$ as*

$$\mathcal{U}_n = \left\{ u \in \mathcal{U}_1 : u^{(i)} \cdot (-1)^{i+1} \geq 0, i = 1, ..., n \right\}.$$

Notice that the restrictions for higher order $\mathcal{U}$ contain all the restrictions of the lower order and there are some additional restrictions for higher order derivatives, which implies $... \subset \mathcal{U}_3 \subset \mathcal{U}_2 \subset \mathcal{U}_1$.

Definition 8. *Let X and Y be random variables, we say that X first order stochastically dominates Y ($X \geq_{FSD} Y$) if*

$$\mathsf{E}[u(X)] \geq \mathsf{E}[u(Y)]\ \forall u \in \mathcal{U}_1.$$

From the definition, by choosing identity as our utility function, we can see that $\mathsf{E}\ X \geq \mathsf{E}\ Y$ is a necessary condition (which holds for higher order dominance, too).

Definition 9. *Let X and Y be random variables, we say that X second order stochastically dominates Y ($X \geq_{SSD} Y$) if*

$$\mathsf{E}[u(X)] \geq \mathsf{E}[u(Y)]\ \forall u \in \mathcal{U}_2.$$

We define higher orders of stochastic dominance in the same way, we just assume $u \in \mathcal{U}_n$ for order n. Note that $\mathcal{U}_1$ and $\mathcal{U}_2$ translates to non-decreasing and concave non-decreasing functions for the FSD and the SSD.

From the definition, we can see that lower order stochastic dominance implies higher orders of stochastic dominance, for example $X \geq_{FSD} Y \Rightarrow X \geq_{SSD} Y$, which follows from the fact that $\mathcal{U}_2 \subset \mathcal{U}_1$.

For strict stochastic dominance of order n ($X >_{nSD} Y$), we require existence of a utility function v satisfying the restrictions stated above, such that:

$$\mathsf{E}\left[v(X)\right] > \mathsf{E}\left[v(Y)\right].$$

We also state the equivalent conditions for the FSD and the SSD, which can be found in Levy [2015].

Definition 10. *Let X be a random variable and $F(x)$ its distribution function, the we define integrated distribution function as:*

$$F_X^{(2)}(x) = \int_{-\infty}^{x} F(t)dt.$$

Theorem 1. *Let X_1 and X_2 be a random variable and F_{X_1} and F_{X_2} their distribution functions, then:*

(i)

$$X_1 \geq_{FSD} X_2 \iff F_{X_1}(x) \leq F_{X_2}(x), \ \forall x \in \mathbb{R}$$

(ii)

$$X_1 \geq_{SSD} X_2 \iff F_{X_1}^{(2)}(x) \leq F_{X_2}^{(2)}(x), \ \forall x \in \mathbb{R}$$

(iii)

$$X_1 >_{FSD} X_2 \iff F_{X_1}(x) \leq F_{X_2}(x), \ \forall x \in \mathbb{R}$$

and there exists at least one $x \in \mathbb{R}$ for which the inequality is sharp.

(iv)

$$X_1 >_{SSD} X_2 \iff F_{X_1}^{(2)}(x) \leq F_{X_2}^{(2)}(x), \ \forall x \in \mathbb{R}$$

and there exists at least one $x \in \mathbb{R}$ for which the inequality is sharp.

Proof can be found for example in Hanoch and Levy [1975].

1.2.3 Special cases of stochastic dominance

For some commonly used parametric distributions, the stochastic dominance constraints translate to constraints for the parameters of the distributions. For example for normal distribution:

Theorem 2. *Let $X_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$ and $X_2 \sim \mathcal{N}(\mu_2, \sigma_2^2)$, then:*

(i)

$$X_1 >_{FSD} X_2 \iff \mu_1 > \mu_2 \ \wedge \ \sigma_1^2 = \sigma_2^2,$$

(ii)

$$X_1 >_{SSD} X_2 \iff \mu_1 \geq \mu_2 \ \wedge \ \sigma_1^2 \leq \sigma_2^2$$

and at least one inequality is sharp.

Proof can be found for example in Levy [2015].
We can see that the condition for the SSD corresponds to the classical Markowitz criterion (dominant variable has higher mean and lower variance).

For discrete distributions with equally probable atoms, the stochastic dominance constraints do also have a simple formulation for the first and the second order.

Theorem 3. *Let X attains values $x_1 \leq ... \leq x_T$ with probability $1/T$ and let Y attains values $y_1 \leq ... \leq y_T$ with probability $1/T$. Then*

$$X \geq_{FSD} Y \iff x_t \geq y_t, \ t = 1, ..., T, \tag{1.4}$$

$$X \geq_{SSD} Y \iff \sum_{s=1}^{t} x_s \geq \sum_{s=1}^{t} y_s, \ t = 1, ..., T. \tag{1.5}$$

Note that the sum for $t = 1$ is the same as the FSD criterion for the first pair of the ordered values, which means that the worst scenario value of the dominant variable must be greater or equal to the worst scenario value of the dominated variable (lowest return of the dominant portfolio must be at least as high as the lowest return of the dominated portfolio).

1.2.4 Models with stochastic dominance constraints

Now we want to use the stochastic dominance approach to create a portfolio that is better than some benchmark portfolio. We can use for example index portfolio as a benchmark or we can use our current portfolio and find out, whether we should re-balance the portfolio. The stochastic dominance ensures that the expected return must be at least the same, which agrees with the constraint for the minimal expected return from classical models.

To obtain such portfolio, we use the historical observations (scenarios) and we use static (one period) models.

Let us define the optimization problem to find a dominating portfolio:

Definition 11. *Let us have a portfolio of N stocks with weights $\boldsymbol{w} = (w_1, .., w_N)$, where $\sum_{i=1}^{N} w_i = 1$. Let Y be a random variable denoting the returns of our benchmark portfolio, then we define the optimization problem:*

$$\begin{gathered} \max_{\boldsymbol{w}} \ f(\boldsymbol{w}) \\ \textit{subject to } R(\boldsymbol{w}) \geq_{nSD} Y \\ \boldsymbol{w} \in W, \end{gathered} \tag{1.6}$$

where $f(\boldsymbol{w})$ is a concave continuous function and

$$W = \left\{ \boldsymbol{w} \in \mathbb{R}^N : \sum_{i=1}^{N} w_i = 1, w_i \geq 0, i = 1, .., N \right\}.$$

The objective function $f(\boldsymbol{w})$ can be chosen for example to be the expected return $f(\boldsymbol{w}) = \mathsf{E}[R(\boldsymbol{w})]$.

Usual assumption of no short selling can be easily replaced by bounded exposure (bounding of the weights from both sides or at least from below).

The first and second order stochastic dominance is discussed in Kuosmanen [2004], where permutation and doubly stochastic matrices are used. It follows from the special case for discrete distributions (1.4) and (1.5), the permutation matrix is used to order the observations. By a permutation matrix we understand

$$\boldsymbol{\pi} \in \{0,1\}^{T\times T} : \sum_{s=1}^{T} \pi_{ts} = 1, t = 1, ..., T; \sum_{t=1}^{T} \pi_{ts} = 1, s = 1, ..., T.$$

The permutation is used to create pairs of observations of our and benchmark portfolio which are compared.

By a doubly stochastic matrix we understand

$$\boldsymbol{\pi} \in \mathbb{R}_{+}^{T\times T} : \sum_{s=1}^{T} \pi_{ts} = 1, t = 1, ..., T; \sum_{t=1}^{T} \pi_{ts} = 1, s = 1, ..., T.$$

Doubly stochastic matrix can also transfer the extra value of dominant variable to the next sum of the constraints as t goes from 1 to T in (1.5) (the extra returns on the left tail can compensate for lower returns on the right tail of the returns distribution). The problem was then reformulated in Luedtke [2008] for distributions with different numbers of atoms and with different probabilities.

First order stochastic dominance

Let random return rates have a discrete joint distribution with realizations $r_{it}, t = 1, ..., T,\ i = 1, ..., N$, attained with probabilities $p_t = {}^1\!/_T, t = 1, 2, ..., T$ and let $y_t, t = 1, ..., T$ again denote the returns of benchmark portfolio again with the same probabilities. The program with the FSD constraints (from Kuosmanen [2004]) is formulated in the following way:

$$\begin{aligned} \max_{\boldsymbol{w},\pi_{ts}} \ & f(\boldsymbol{w}) & & \\ \text{subject to } & \sum_{i=1}^{N} w_i r_{it} \geq \sum_{s=1}^{T} \pi_{ts} y_s, & & t = 1, ..., T \\ & \sum_{s=1}^{T} \pi_{ts} = 1, & & t = 1, ..., T \\ & \sum_{t=1}^{T} \pi_{ts} = 1, & & s = 1, ..., T \\ & \pi_{ts} \in \{0,1\}, & & t = 1, ..., T; s = 1, ..., T \\ & \boldsymbol{w} \in W, & & \end{aligned} \tag{1.7}$$

where π_{st} represent the elements of a permutation matrix.

The program orders the observations and check whether the return for our portfolio is greater than or equal to the corresponding return of the benchmark.

We also state the more general formulation of the constraints. Let the benchmark portfolio attains returns $y_1 < ... < y_D$ with probabilities $q_k, k = 1, .., D$ and the available stocks returns again attain values $r_{it}, t = 1, ..., T; i = 1, ..., N$, with probabilities $p_t, t = 1, 2, ..., T$. Then the program with the FSD constraints (from Luedtke [2008]) is formulated as:

$$
\begin{aligned}
\max_{\boldsymbol{w}, \pi_{ts}, v_k} \quad & f(\boldsymbol{w}) & & \\
\text{subject to} \quad & \sum_{s=1}^{D} \pi_{ts} = 1, & & t = 1, ..., T \\
& \sum_{i=1}^{N} w_i r_{it} \geq \sum_{s=1}^{D} y_s \pi_{ts}, & & t = 1, ..., T \\
& v_s - \sum_{t=1}^{T} p_t \pi_{ts} = 0, & & s = 1, ..., D \qquad (1.8) \\
& \sum_{s=1}^{k-1} v_s \leq \sum_{s=1}^{k-1} q_s, & & k = 1, ..., D \qquad (1.9) \\
& v_k \in \mathbb{R}, & & k = 1, ..., D \\
& \pi_{ts} \in \{0, 1\}, & & t = 1, ..., T; s = 1, ..., D \\
& \boldsymbol{w} \in W. & &
\end{aligned}
$$

In this more general case, we again order the returns, but now more observations can correspond to only one observation from the benchmark and vice-versa. Then there is a constraint for the sum of probabilities. The return of our portfolio we choose must have sufficient probability to cover the corresponding return of the benchmark.

In the previous formulation variables v_s are presented for computational purposes to break down one more complex constraint combining (1.8) and (1.9).

Second order stochastic dominance

When we focus on the second order stochastic dominance, the problem corresponds to risk aversion. It was also shown in Dentcheva and Ruszczynski [2004] that the second order stochastic dominance is just convexification of the problem with the first order stochastic dominance for random variables on probability space $(\Omega, \mathcal{F}, \mathsf{P})$, where $\Omega = \{\omega_1, ..., \omega_T\}, T \in \mathbb{N}$.

The assumption corresponds to the fact that we use the empirical distribution function as the true distribution, this allows us to reformulate the problem. Let random return rates have a discrete joint distribution with realizations $r_{it}, t = 1, ..., T; i = 1, ..., N$, attained with probabilities $p_t, t = 1, 2, ..., T$. This means we have T possible realizations and N stocks.

The problem was reformulated using $CVaR$ definition of stochastic dominance in Dentcheva and Ruszczyński [2006] introducing variables s_{ts} representing shortfall of $R(\boldsymbol{w})$ below y_t (realization of benchmark portfolio) in realization

$t, s = 1, ..., T$ as:

$$
\begin{aligned}
\max_{\boldsymbol{w}, s_{ts}} \ & f(\boldsymbol{w}) \\
\text{subject to} \ & \sum_{i=1}^{N} w_i r_{is} + s_{ts} \geq y_t, \quad && t = 1, ..., T; s = 1, ..., T \\
& \sum_{s=1}^{T} p_s s_{ts} \leq F_Y^{(2)}(y_t), \quad && t = 1, ..., T \\
& s_{ts} \geq 0, && t = 1, ..., T; s = 1, ..., T \\
& \boldsymbol{w} \in W.
\end{aligned}
\tag{1.10}
$$

In this approach, the integrated distribution functions are compared.

For expected return $f(\boldsymbol{w}) = \mathsf{E}\left[R(\boldsymbol{w})\right] = \sum_{t=1}^{T} p_t \sum_{i=1}^{N} w_i r_{it}$ this is a linear programming problem.

Similarly to the case of the FSD, the more general case can be found in Luedtke [2008]. Let the benchmark portfolio attains returns $y_1 < ... < y_D$ with probabilities $q_k, k = 1, .., D$ and let again stock returns attain values $r_{it}, t = 1, ..., T; i = 1, ..., N$, with probabilities $p_t, t = 1, 2, ..., T$. The SSD constraint is then formulated as:

$$
\begin{aligned}
\max_{\boldsymbol{w}, \pi_{ts}, v_k} \ & f(\boldsymbol{w}) \\
\text{subject to} \ & \sum_{s=1}^{D} \pi_{ts} = 1, && t = 1, ..., T \\
& \sum_{i=1}^{N} w_i r_{it} \geq \sum_{s=1}^{D} y_s \pi_{ts}, && t = 1, ..., T \\
& v_s - \sum_{t=1}^{T} p_t \pi_{ts} = 0, && s = 1, ..., D \\
& \sum_{s=1}^{k-1} (y_k - y_s) v_s \leq \sum_{s=1}^{k-1} (y_k - y_s) q_s, \quad && k = 1, ..., D \\
& v_k \in \mathbb{R}, && k = 1, ..., D \\
& \pi_{ts} \geq 0, && t = 1, ..., T; s = 1, ..., D \\
& \boldsymbol{w} \in W.
\end{aligned}
\tag{1.11}
$$

In this approach, we do not use the integrated distribution functions, but we compare the returns themselves after the ordering and ”transfer of extra returns”. We again must keep track of the probabilities for which there are the constraints with v_s.

Third order stochastic dominance

Portfolio optimization with the third order stochastic dominance in constraints is discussed in Post and Kopa [2017]. Authors define the super-convex third order stochastic dominance (SCTSD), which is proven to be sufficient condition for the third order stochastic dominance (TSD), and take it as an approximation of the TSD. Let us formulate the definition of SCTSD. We assume that the returns of benchmark portfolio are ordered, i.e. $y_1 \leq ... \leq y_T$.

Definition 12. *Let $\epsilon_s \geq 0, s = 1, ..., T$ a series of data-dependent tolerance parameters that are defined as $\epsilon_1, \epsilon_2 = 0$ and*

$$\epsilon_s = \left(\frac{\mathcal{S}^2_{\boldsymbol{\tau}}(y_s)}{\mathcal{S}^2_{\boldsymbol{\tau}}(y_{s-1}) + 2\mathcal{E}_{\boldsymbol{\tau}}(y_{s-1})(y_s - y_{s-1})} - 1 \right), s = 3, ..., T.$$

Portfolio $\boldsymbol{w} \in W$ dominates benchmark portfolio $\boldsymbol{\tau} \in W$ by the super-convex third order stochastic dominance (SCTSD), or $\boldsymbol{w} \geq_{SCTSD} \boldsymbol{\tau}$, if:

$$(1 + \epsilon_s)\mathcal{S}^2_{\boldsymbol{w}}(y_s) \leq \mathcal{S}^2_{\boldsymbol{\tau}}(y_s), s = 1, ...T, \tag{1.12}$$

$$\sum_{t=1}^{T} p_t \sum_{i=1}^{N} w_i r_{it} \geq \sum_{t=1}^{T} p_t y_t, \tag{1.13}$$

where $y_t = \sum_{i=1}^{N} \tau_i r_{it}$ denotes the return of benchmark portfolio and $\mathcal{S}^2_{\boldsymbol{\tau}}(y_s)$ denotes semivariance:

$$\mathcal{S}^2_{\boldsymbol{w}}(x) = \sum_{t=1}^{T} p_t (x - \sum_{i=1}^{N} w_i r_{it})^2 D_{\boldsymbol{w},t}(x),$$

where $D_{\boldsymbol{w},t}(x), t = 1, ..., T$ is a binary variable that takes value 1 if $\sum_{i=1}^{N} w_i r_{it} \leq x$ and 0 otherwise. $\mathcal{E}_{\boldsymbol{\tau}}(y_s)$ denotes expected shortfall:

$$\mathcal{E}_{\boldsymbol{w}}(x) = \sum_{t=1}^{T} p_t (x - \sum_{i=1}^{N} w_i r_{it}) D_{\boldsymbol{w},t}(x).$$

Returns, probabilities and weights are denoted the same way as in the previous sections.

Further in the paper, the SCTSD constraint is reformulated as system of linear and convex quadratic constraints, which leads to the optimization problem:

$$\begin{aligned}
\max_{\boldsymbol{w}, \theta_{st}} \; & f(\boldsymbol{w}) \\
\text{subject to } & (1 + \epsilon_s) \sum_{t=1}^{T} p_t \theta_{st}^2 \leq \mathcal{S}^2_{\boldsymbol{\tau}}(y_s), && s = 1, ..., T \\
& - \theta_{st} - \sum_{i=1}^{N} w_i r_{it} \leq -y_s, && t, s = 1, ..., T \\
& - \sum_{t=1}^{T} p_t \sum_{i=1}^{N} w_i r_{it} \leq - \sum_{t=1}^{T} p_t y_t \\
& \theta_{st} \geq 0, && s, t = 1, ..., T \\
& \boldsymbol{w} \in W.
\end{aligned} \tag{1.14}$$

For computation purposes, the number of y_s can be reduced to approximate the result and with more than 100 values between $\min_s(y_s)$ and $\max_s(y_s)$ it is usually a good approximation. This leads to a reduction of the number of variables and also to a reduction of the number of constraints.

2. Robust optimization

As we saw in the previous part, we are using historical observations as the true distribution of the random returns. To deal with the fact that it is just an estimate of the real distribution, robustness is added to the optimization problem. We want our optimal portfolio to stochastically dominate the benchmark portfolio even when the input data slightly change. This is called distributional robustness. When we talk about robustness in this work, we mean distributional robustness.

In this chapter we often state variables below the max or min to denote which variables we optimize over, the indexes just refer to the sets that define the dimension of the problem and we try to use always the same indexes no to confuse the reader (i for stock/asset and t for scenario). Those indexes do not refer and are not connected to anything else.

2.1 Robust optimization with stochastic dominance constraints

We define the robust stochastic dominance in accordance with Dentcheva and Ruszczyński [2010].

Definition 13. *We say that a random variable X dominates robustly a random variable Y in the n-th order over a set of probability measures Q ($X \geq_{nSD}^{Q} Y$) if*

$$\mathsf{E}_P[u(X)] \geq \mathsf{E}_P[u(Y)] \; \forall u \in \mathcal{U}_n \; \forall P \in Q. \tag{2.1}$$

We understand X and Y as random variables denoting return of some portfolios, which consist of stocks, and that the joint distribution of the random returns of the underlying stocks is defined by the distribution P, which we allow to change slightly.

The set Q was not specified and as we mentioned at the beginning, we want our portfolio to be prepared for slight changes in the distribution. For this purpose, we select a suitable measure of distance between two distributions and we bound the change of the distribution by a constant, which defines the set Q.

2.2 The Wasserstein distance

We use the definition and computation procedures from Pflug and Pichler [2014]. We want to define distance between two probability distributions on $\mathbb{R}^N$ (N is the number of stocks).

Definition 14. *Let there be two probability spaces $(\Omega_1, \mathcal{F}_1, P_1)$ and $(\Omega_2, \mathcal{F}_2, P_2)$. The Wasserstein distance of order r ($r \geq 1$) is defined as:*

$$d_r(P_1, P_2) = \left(\inf_{\pi} \iint_{\Omega_1 \times \Omega_2} d(\omega_1, \omega_2)^r \pi(d\omega_1, d\omega_2) \right)^{\frac{1}{r}}, \tag{2.2}$$

where the infimum is among all joint probability measures π on $\Omega_1 \times \Omega_2$ which satisfy

$$\pi(A \times \Omega_2) = P_1(A) \wedge \pi(\Omega_2 \times B) = P_2(B) \; \forall A \in \mathcal{F}_1 \; \forall B \in \mathcal{F}_2 \textit{ measurable}$$

and $d(\omega_1, \omega_2)$ is inherited distance between elements of Ω_1 and Ω_2. Let ξ_1 be $\mathbb{R}^N$-valued random variable on Ω_1 and ξ_2 be $\mathbb{R}^N$-valued random variable on Ω_2 then:

$$d(\omega_1, \omega_2) = D\left(\xi_1(\omega_1), \xi_2(\omega_2)\right)$$

for some distance D in $\mathbb{R}^N$. For example a norm in $\mathbb{R}^N$.

The Wasserstein distance is a distance, i.e. it satisfies triangle inequality. It is also r-convex: for $0 \leq \lambda \leq 1$:

$$d_r(P, (1-\lambda)P_0 + \lambda P_1)^r \leq (1-\lambda) d_r(P, P_0)^r + \lambda d_r(P, P_1)^r \tag{2.3}$$

For proof see Pflug and Pichler [2014].

2.2.1 Convexity of Q

Now let us have a look at the set Q defined as a closed neighborhood of the empirical distribution using the Wasserstein distance.

Theorem 4. *Let Q be a closed neighborhood of a probability distribution defined by the Wasserstein distance, then Q is convex.*

Proof. Q is defined as $Q = \{P : d_r(P, P_0) \leq \epsilon\}$. Let $P_1, P_2 \in Q$, that is $d_r(P_1, P_0) \leq \epsilon, d_r(P_2, P_0) \leq \epsilon$. From convexity (2.3) we have:

$$\begin{aligned} d_r(P_0, (1-\lambda)P_1 + \lambda P_2)^r &\leq (1-\lambda) d_r(P_0, P_1)^r + \lambda d_r(P_0, P_2)^r \\ &\leq (1-\lambda)\epsilon^r + \lambda \epsilon^r = \epsilon^r \end{aligned} \tag{2.4}$$

Now by taking $1/r$ power of each side gives us the result. □

Note that we can use any fixed observed distribution, it does not have to be empirical distribution and the convexity still holds.

2.2.2 Wasserstein distance in a discrete framework

We adjust definition 14 to our case, where both distributions have the same number of atoms.

Definition 15. *Let us have two discrete distributions with finite support. P_1 attains values $x_1, ..., x_T$ with probabilities $p_1, ..., p_T$ and P_2 attains values $y_1, ..., y_T$ with probabilities $q_1, ..., q_T$. The Wasserstein distance of order r ($r \geq 1$) corresponds to solving the following linear program:*

$$\begin{aligned} \min_{\xi_{ts}} \quad & \sum_{t=1}^{T} \sum_{s=1}^{T} \xi_{ts} d_{ts}^r \\ \textit{subject to} \quad & \sum_{s=1}^{T} \xi_{ts} = p_t, \quad t = 1, ..., T \\ & \sum_{t=1}^{T} \xi_{ts} = q_s, \quad s = 1, ..., T \\ & \xi_{ts} \geq 0, \qquad t, s = 1, ..., T. \end{aligned} \tag{2.5}$$

We can also use the dual formulation:

$$\begin{aligned}
\max_{\lambda_t,\mu_s} \quad & \sum_{t=1}^{T} p_t\lambda_t + \sum_{s=1}^{T} q_s\mu_s \\
\textit{subject to} \quad & \lambda_t + \mu_s \le d_{ts}^r && t,s = 1,...,T \\
& \lambda_t \in \mathbb{R}, && t = 1,...,T \\
& \mu_s \in \mathbb{R}, && s = 1,...,T,
\end{aligned} \tag{2.6}$$

where $d_{ts} = d(x_t, y_s)$ denotes a distance between the points x_t and y_s.

For our purposes, we need the flexibility both in values and in probabilities. The first distribution represents our estimate of the distribution based on the observations $r_{it}^0, i = 1,...,N; t = 1,...,T$ and $p_t = 1/T, t = 1,..,T$, and the second represents the changed distribution for robustness purposes. As for the value of r (order of the Wasserstein distance), $r = 1$ or $r = 2$ is usually chosen.

2.3 Worst case distribution

2.3.1 Second order stochastic dominance - first formulation

In Dentcheva and Ruszczyński [2010], we can find an equivalent formulation of robust stochastic dominance of the second order $X \ge_{SSD}^{Q} Y$:

$$\sup_{P\in Q,\eta\in\mathbb{R}} \mathsf{E}_P\left[(\eta - X)_+ - (\eta - Y)_+\right] \le 0. \tag{2.7}$$

So we can think of the worst case distribution for given η as the one in which the supremum is attained (or a close distribution if it is not attained). Let us define the worst case distribution.

Definition 16. *We say that distribution P is the worst-case distribution for the SSD from Q if the supremum in problem* (2.7) *is attained for this P for some $\eta \in \mathbb{R}$.*

Now let us have N stocks/assets and let the random return rates have a discrete joint distribution P with realizations $r_{it}, t = 1,...,T; i = 1,...,N$, attained with probabilities $p_t, t = 1,2,...,T$. Let our portfolio have weights $\boldsymbol{w}$ and our benchmark portfolio has weights $\boldsymbol{\tau}$. Then we can write the supremum as:

$$\sup_{r_{it},p_t} \sum_{t=1}^{T} \left((\eta - \sum_{i=1}^{N} w_i r_{it})_+ - (\eta - \sum_{i=1}^{N} \tau_i r_{it})_+ \right) p_t \le 0 \;\forall \eta \in \mathbb{R}, \tag{2.8}$$

where the constraints for r_{it} and p_t are defined later.

We also know from Dentcheva and Ruszczyński [2006] that we do not need to consider all possible η, it is enough to require it only for the realizations of the benchmark portfolio. We also choose Q to be closed so the supremum is attained and we can write max instead of sup. So we get:

$$\max_{r_{it},p_t,k} \sum_{t=1}^{T} \left((\eta_k - \sum_{i=1}^{N} w_i r_{it})_+ - (\eta_k - \sum_{i=1}^{N} \tau_i r_{it})_+ \right) p_t \le 0, \eta_k = \sum_{i=1}^{N} \tau_i r_{ik}, k = 1,...,T. \tag{2.9}$$

Further, let us denote our observed values by 0, which means r_{jt}^0 and p_t^0. Moreover let us define the set Q as a closed neighborhood defined by the Wasserstein distance and suppose that the number of atoms remains unchanged. We work with the returns as simple returns in terms of joint return of a portfolio being just a weighted average of returns of the single stocks. To make sure that the returns make sense, we must take returns greater than -1, which represents losing the whole investment. Now let us not think of the equation (2.9) as a constraint but as a maximization problem. We can always check whether the value of the objective function is smaller than or equal to 0. We use the definition (2.5) with $r = 1$. Since the empirical distribution has probabilities of all scenarios $1/T$, we get:

$$
\begin{aligned}
\max_{\xi_{ts}, r_{it}, p_t, k} \quad & \sum_{t=1}^{T} \left((\eta_k - \sum_{i=1}^{N} w_i r_{it})_+ - (\eta_k - \sum_{i=1}^{N} \tau_i r_{it})_+ \right) p_t \\
\text{subject to } & \eta_k = \sum_{i=1}^{N} \tau_i r_{ik}, && k = 1, ..., T \\
& \min \sum_{t=1}^{T} \sum_{s=1}^{T} \xi_{ts} \left\| \boldsymbol{r}_t^0 - \boldsymbol{r}_s \right\| \leq \epsilon \\
& \sum_{s=1}^{T} \xi_{ts} = \frac{1}{T}, && t = 1, ..., T \\
& \sum_{t=1}^{T} \xi_{ts} = p_s, && s = 1, ..., T \\
& \xi_{ts} \geq 0, && t, s = 1, ..., T \\
& r_{it} \geq -1, && i = 1, ..., N; t = 1, ..., T \\
& p_t \geq 0, && t = 1, ..., T.
\end{aligned}
\tag{2.10}
$$

where $\boldsymbol{r}_t = (r_{1t}, ..., r_{Nt})$ denotes the vector of returns in scenario t and $\|\cdot\|$ denotes some norm, we can use for example L_1 norm or Euclidean norm.

Now we can also remove the minimum, because we are fine if we find just one solution for which the value is smaller than or equal to ϵ, because the minimum of the LHS of the inequality must be smaller than or equal to the value of the

LHS for the found solution. So we get:

$$
\begin{aligned}
\max_{\xi_{ts}, r_{it}, p_t, k} \quad & \sum_{t=1}^{T} \left((\eta_k - \sum_{i=1}^{N} w_i r_{it})_+ - (\eta_k - \sum_{i=1}^{N} \tau_i r_{it})_+ \right) p_t \\
\text{subject to } \eta_k = & \sum_{i=1}^{N} \tau_i r_{ik}, && k = 1, ..., T \\
& \sum_{t=1}^{T} \sum_{s=1}^{T} \xi_{ts} \left\| \boldsymbol{r}_t^0 - \boldsymbol{r}_s \right\| \leq \epsilon \\
& \sum_{s=1}^{T} \xi_{ts} = \frac{1}{T}, && t = 1, ..., T \\
& \sum_{t=1}^{T} \xi_{ts} = p_s, && s = 1, ..., T \\
& \xi_{ts} \geq 0, && t, s = 1, ..., T \\
& r_{it} \geq -1, && i = 1, ..., N; t = 1, ..., T \\
& p_t \geq 0, && t = 1, ..., T.
\end{aligned}
\tag{2.11}
$$

Now we remove the positive parts. We replace $(\eta_k - \sum_{i=1}^{N} \tau_i r_{it})_+$ by the optimization problem:

$$
\begin{aligned}
\min \ & \delta_{tk} \\
\text{subject to } & \delta_{tk} \geq 0 \\
& \delta_{tk} \geq \eta_k - \sum_{i=1}^{N} \tau_i r_{it}.
\end{aligned}
\tag{2.12}
$$

We can also merge the minimum with the outer maximum, because we have a maximization problem and there is a minus in front of the second term. We do the same with the first term, but we must use the dual formulation:

$$
\begin{aligned}
\max \ & y_{tk} \left(\eta_k - \sum_{i=1}^{N} w_i r_{it} \right) \\
\text{subject to } & y_{tk} \geq 0 \\
& y_{tk} \leq 1.
\end{aligned}
\tag{2.13}
$$

We can again merge the two problems because they are both maximization.

To implement in software, one may use binary variables to get only one objective function and do not optimize over the value of index k, then the objective function is in form of:

$$
\sum_{k=1}^{T} \left(\sum_{t=1}^{T} \left(y_{tk} (\eta_k - \sum_{i=1}^{N} w_i r_{it}) - \delta_{tk} \right) p_t \right) \cdot b_k,
$$

where b_k are binary variables and $\sum_{k=1}^{T} b_k = 1$. Moreover we can relax the binary restriction of b_k and take just non-negative b_k, because the maximization program takes b_k equal to 1 for the term that is the largest.

Now we have the final formulation with the use of Euclidean norm without the square root (the square root was removed for computational purposes):

$$\begin{aligned}
\max_{y_{tk},\delta_{tk},\xi_{ts},r_{it},p_t,b_k} \quad & \sum_{k=1}^{T}\left(\sum_{t=1}^{T}\left(y_{tk}(\eta_k-\sum_{i=1}^{N}w_ir_{it})-\delta_{tk}\right)p_t\right)\cdot b_k && \\
\text{subject to } \eta_k &= \sum_{i=1}^{N}\tau_ir_{ik}, && k=1,...,T \\
& \sum_{t=1}^{T}\sum_{s=1}^{T}\xi_{ts}\sum_{i=1}^{N}(r_{it}^0-r_{is})^2 \le \epsilon && \\
& \sum_{s=1}^{T}\xi_{ts}=\frac{1}{T}, && t=1,...,T \\
& \sum_{t=1}^{T}\xi_{ts}=p_s, && s=1,...,T \\
& y_{tk}\ge 0, && t,k=1,...,T \\
& y_{tk}\le 1, && t,k=1,...,T \\
& \delta_{tk}\ge 0, && t,k=1,...,T \\
& \delta_{tk}\ge \eta_k-\sum_{i=1}^{N}\tau_ir_{it}, && t,k=1,...,T \\
& \xi_{ts}\ge 0, && t,s=1,...,T \\
& r_{it}\ge -1, && i=1,...,N \\
& && t=1,...,T \\
& p_t\ge 0, && t=1,...,T \\
& b_k\ge 0, && k=1,...,T \\
& \sum_{k=1}^{T}b_k=1.
\end{aligned} \tag{2.14}$$

The resulting distribution defined by r_{it} and p_t can be considered as the worst case distribution by our previous considerations. Also notice that we do not have to add a constraint for the probabilities to sum up to 1, this follows from the Wasserstein distance constraints, specifically from that $\sum_{s=1}^{T}\xi_{ts}=\frac{1}{T}, t=1,...,T$ by summing over index t.

The selection of the metrics is not very important, it does affect the feasibility of specific distributions in the neighborhood slightly, but the concept remains the same and the choice highly affects the computational tractability.

One can also penalize more the returns that are further by replacing the second power by fourth, sixth, ..., basically by any even positive integer to keep the value non-negative.

Let us formulate the findings in the following theorem.

Theorem 5. *Let X and Y be random variables denoting returns of a portfolio defined by weights $\boldsymbol{w}$ and $\boldsymbol{\tau}$. Let us have observed historical returns $r_{it}^0, i=1,...,N; t=1,...,T$. Let us have Q a set of probability measures defined on $\mathbb{R}^N$ with T atoms: determined by returns $r_{it}, i=1,...,N; t=1,...,T$ and probabilities $p_t, t=1,...,T$ defined as a neighborhood of the empirical distribution. Let* *the neighborhood be defined with the use of the Wasserstein distance and let the distance on $\mathbb{R}^N$ be defined as Euclidean norm squared, i.e. let $x, y \in \mathbb{R}^N$, then $d(x,y) = \sum_{i=1}^{N}(x_i - y_i)^2$. Then X dominates robustly Y in the second order over the set of probability measures Q ($X \geq_{SSD}^{Q} Y$) if and only if the optimal objective value of the problem* (2.14) *is less than or equal to zero.*

2.3.2 Second order stochastic dominance - alternative formulation

To derive another formulation of the robust SSD test, we use one of the formulations stated and proved in Luedtke [2008]. We state the formulation as the following theorem.

Theorem 6. *Let us have X and Y discrete random variables attaining values $x_1, ..., x_T$ and $y_1, ..., y_T$ respectively with probabilities $p_1, ..., p_T$. Then $X \geq_{SSD} Y$ if and only if there exists $\pi \in \mathbb{R}_+^{T \times T}$ such that*

$$\begin{aligned} \sum_{s=1}^{T} y_s \pi_{ts} - x_t \leq 0, &\quad t = 1, ..., T \\ \sum_{s=1}^{T} \pi_{ts} = 1, &\quad t = 1, ..., T \\ \sum_{t=1}^{T} p_t \pi_{ts} = p_s, &\quad s = 1, ..., T. \end{aligned} \tag{2.15}$$

We again understand that x_t are returns (scenarios) of our portfolio and that y_t are returns (scenarios) of the benchmark portfolio. Both portfolios consist of stocks with random returns defined by a probability distribution P. We assume that the distribution is discrete with T probability atoms. We understand the set Q (from which we take the possible distributions for the stocks) as a set of probability distributions with T probability atoms on $\mathbb{R}^N$, where in our case the set is defined as the neighborhood of the empirical distribution defined by the Wasserstein distance. We would like to use this equivalent condition to define the worst case distribution, but there are also variables π_{ts}, which are in the definition and we do not know which ones to take for different values of returns. For this reasons, we define the worst case distribution later on.

We could also define the worst case in another way: not as maximum over the inequalities but as a sum of positive parts, i.e.

$$\sum_{t=1}^{T} \sum_{s=1}^{T} (y_s \pi_{ts} - x_t)_+ .$$

Let us reformulate the conditions: $X \geq_{SSD} Y$ if and only if there exists $z \in \mathbb{R}$

and $\pi \in \mathbb{R}_+^{T\times T}$ such that

$$\begin{aligned} z \leq& 0 \\ \sum_{s=1}^{T} y_s \pi_{ts} - x_t \leq& z, \quad t = 1, ..., T \\ \sum_{s=1}^{T} \pi_{ts} =& 1, \quad t = 1, ..., T \\ \sum_{t=1}^{T} p_t \pi_{ts} =& p_s, \quad s = 1, ..., T. \end{aligned} \tag{2.16}$$

This means that if we manage to find $z \leq 0$, then $X \geq_{SSD} Y$, so if the minimum (over π_{ts} and z) of z given the other constraints of (2.16) is non-positive then $X \geq_{SSD} Y$. Notice that z represents the fulfilling of the SSD criterion and that it also represents the maximum of the LHS over the index t.

Now we want to let the distribution to be able to vary from the empirical distribution. Again let us have N stocks/assets and let the random return rates have a discrete joint distribution P with realizations $r_{it}, t = 1, ..., T; i = 1, ..., N$, attained with probabilities $p_t, t = 1, ..., T$. Let our portfolio have weights $\boldsymbol{w}$ and our benchmark portfolio has weights $\boldsymbol{\tau}$. Further, let us denote our observed values by 0, which means r_{it}^0 and p_t^0. Moreover let us define the set of possible distributions as a closed neighborhood defined by the Wasserstein distance and suppose that the number of atoms remains unchanged. When z represents the fulfilling of the SSD criterion, then the worst case distribution can be understood as the one that maximizes the value of z over the possible distributions (when lower values mean that the criterion is more fulfilled). Again, let us not think of it as an constraint, but as an optimization problem. For easier notation, we now use y_t and x_t to denote returns of benchmark and our portfolio in scenario t

respectively:

$$
\begin{aligned}
&\max_{r_{it},p_t,\xi_{ts}} \min_{z,\pi_{ts}} z \\
&\text{subject to } \sum_{s=1}^{T} y_s \pi_{ts} - x_t \le z, && t = 1,...,T \\
&\sum_{s=1}^{T} \pi_{ts} = 1, && t = 1,...,T \\
&\sum_{t=1}^{T} p_t \pi_{ts} = p_s, && s = 1,...,T \\
&\pi_{ts} \ge 0, && t,s = 1,...,T \\
&\sum_{t=1}^{T}\sum_{s=1}^{T} \xi_{ts} \sum_{i=1}^{N} (r_{it}^0 - r_{is})^2 \le \epsilon \\
&\sum_{s=1}^{T} \xi_{ts} = \frac{1}{T}, && t = 1,...,T \\
&\sum_{t=1}^{T} \xi_{ts} = p_s, && s = 1,...,T \\
&\xi_{ts} \ge 0, && t,s = 1,...,T \\
&r_{it} \ge -1, && i = 1,...,N; t = 1,...,T \\
&p_t \ge 0, && t = 1,...,T \\
&x_t = \sum_{i=1}^{N} w_i r_{it}, && t = 1,...,T \\
&y_t = \sum_{i=1}^{N} \tau_i r_{it}, && t = 1,...,T.
\end{aligned} \tag{2.17}
$$

Definition 17. *We say that distribution P (defined by r_{it} and p_t)is the worst-case distribution for the SSD from Q (defined by the Wasserstein distance in our case) if it is solution to the optimization problem* (2.17)*.*

Now if we look at the inner minimization problem, we can see that it is linear in z and π_{ts}, the other variables are parameters for the inner problem. We can use the dual maximization formulation, it affects only the first 4 constraints. The dual formulation is:

$$
\begin{aligned}
&\max_{u_{1t},u_{2t},u_{3t}} \sum_{t=1}^{T} u_{1t} x_t + \sum_{t=1}^{T} u_{2t} + \sum_{t=1}^{T} u_{3t} p_t \\
&\text{subject to } \sum_{t=1}^{T} -u_{1t} = 1 \\
&y_s u_{1t} + u_{2t} + u_{3s} p_t \le 0, && t,s = 1,...,T \\
&u_{1t} \le 0, && t = 1,...,T \\
&u_{2t}, u_{3t} \in \mathbb{R}, && t = 1,...,T.
\end{aligned} \tag{2.18}
$$

Note that when the solution would give us $p_t = 0$ for some t, the program would still consider the return in this scenario, even though it does not make sense, therefore we add a constraint for $p_t > 0$. The problem is that for software implementation, sharp inequality cannot be used, so we approximate it by adding a very small term, for example 10^{-5} (we represent it as $p_t \geq m$, where m (margin) is a small positive constant).

The fact we want to add a constraint $p_t > 0$ is very complicated. Using this constraint, we remove part of the border of the set of possible distributions. The set is no longer closed and the supremum might not be achieved. This is not a problem when we use ϵ such that no probability can be 0 or we set a lower bound for all the probabilities of scenarios ($p_t \geq m$). In the program we use this lower bound, but one could easily leave the $p_t \geq 0$ and just keep in mind that this distribution is just a limit version of the worst distributions with T probability atoms as $m \to 0+$. One must also keep in mind that the returns in this scenarios with 0 probability are also considered by the program and should understand is as they had some very small probability.

Now plugging it into the whole program and plugging the returns for our and the benchmark portfolio gives us the final formulation:

$$\begin{aligned}
\max_{r_{it},p_t,u_{1t},u_{2t},u_{3t},\xi_{ts}} \quad & \sum_{t=1}^{T} u_{1t} \sum_{i=1}^{N} w_i r_{it} + \sum_{t=1}^{T} u_{2t} + \sum_{t=1}^{T} u_{3t} p_t \\
\text{subject to} \quad & \sum_{s=1}^{T} -u_{1s} = 1 \\
& \sum_{i=1}^{N} \tau_i r_{is} u_{1t} + u_{2t} + u_{3s} p_t \leq 0, && t, s = 1, ..., T \\
& \sum_{t=1}^{T} \sum_{s=1}^{T} \xi_{ts} \sum_{i=1}^{N} (r_{it}^0 - r_{is})^2 \leq \epsilon \\
& \sum_{s=1}^{T} \xi_{ts} = \frac{1}{T}, && t = 1, ..., T \\
& \sum_{t=1}^{T} \xi_{ts} = p_s, && s = 1, ..., T \\
& u_{1t} \leq 0, && t = 1, ..., T \\
& u_{2t}, u_{3t} \in \mathbb{R}, && t = 1, ..., T \\
& \xi_{ts} \geq 0, && t, s = 1, ..., T \\
& r_{it} \geq -1, && i = 1, ..., N; t = 1, ..., T \\
& p_t \geq m, && t = 1, ..., T.
\end{aligned} \tag{2.19}$$

Again we can say that the resulting distribution given by optimal values of r_{it} and p_t can be considered as the worst case distribution in the sense of previous considerations. Moreover we derived a robust SSD test, we formulate our findings in the following theorem.

Theorem 7. *Let X and Y be random variables denoting returns of a portfolio defined by weights $\boldsymbol{w}$ and $\boldsymbol{\tau}$. Let us have observed historical returns $r_{it}^0, i = 1, ..., N; t = 1, ..., T$. Let us have Q a set of probability measures defined on $\mathbb{R}^N$ with T atoms: determined by returns $r_{it}, i = 1, ..., N; t = 1, ..., T$ and probabilities $p_t, t = 1, ..., T$ defined as a neighborhood of the empirical distribution. Let* *the neighborhood be defined with the use of the Wasserstein distance and let the distance on $\mathbb{R}^N$ be defined as Euclidean norm squared, i.e. let $x, y \in \mathbb{R}^N$, then $d(x,y) = \sum_{i=1}^{N}(x_i - y_i)^2$. Then X dominates robustly Y in the second order over the set of probability measures Q ($X \geq_{SSD}^{Q} Y$) if and only if there exists a right open neighborhood of 0 such that for each value m from this neighborhood the optimal value of the problem* (2.19) *is less than or equal to zero.*

Note that we can also use the program with $m = 0$ and then optimal value being less than or equal to zero is a sufficient condition for the robust stochastic dominance. By setting $m = 0$ we allow for more distributions, so the condition holds for a larger set of possible distributions. Also note that if we find m for which there is a feasible solution and the objective value is positive, then the robust stochastic dominance cannot hold.

As was already mentioned when we set a lower bound for all the probabilities of scenarios ($p_t \geq m, m > 0$) then it is enough to run the program for this value of m. Notice that we restrict the set of possible distributions by setting the lower bound.

In comparison with the first robust SSD test, this one does have a simpler formulation, which could be used later in robustification of the programs with the SSD constraints.

2.3.3 First order stochastic dominance

In the first chapter in theorem 1 we did state equivalent conditions for the first order stochastic dominance. We can extend this for the first order robust stochastic dominance $X \geq_{FSD}^{Q} Y$ similarly to the case of second order stochastic dominance:

$$\sup_{P \in Q, x \in \mathbb{R}} F_{X,P}(x) - F_{Y,P}(x) \leq 0. \tag{2.20}$$

Again the condition is understood in the way that the distribution functions of both portfolios depend on a distribution of underlying assets which can vary and is defined by P (we stress this dependence in the subscript of F).

Now we can think of the worst case distribution for a given x as the one when in which the supremum is attained or some limit probability distribution if it is not attained.

Definition 18. *We say that distribution P is the worst-case distribution for the FSD from Q if the supremum of the LHS in problem* (2.20) *is attained for this P for some $x \in \mathbb{R}$.*

Now let us have N stocks/assets and let the random return rates have a discrete joint distribution P with realizations $r_{jt}, t = 1, ..., T; j = 1, ..., N$, attained with probabilities $p_t, t = 1, 2, ..., T$. Let our portfolio have weights $\boldsymbol{w}$ and our benchmark portfolio has weights $\boldsymbol{\tau}$. Then we can rewrite the difference of the distribution functions in x as:

$$\sum_{t=1}^{T} p_t \mathbb{I}_{[\sum_{i=1}^{N} w_i r_{it} \leq x]} - \sum_{t=1}^{T} p_t \mathbb{I}_{[\sum_{i=1}^{N} \tau_i r_{it} \leq x]},$$

where $\mathbb{I}$ represent indicator whether the return of portfolio is smaller than x or not. We also know that a distribution function is non-decreasing which means it is enough to require it only for the points where our portfolio (not the benchmark as it was for the SSD), defined by $\boldsymbol{w}$, has probability atoms. Using again the Wasserstein distance we try to evaluate the supremum by maximizing:

$$
\begin{aligned}
\max_{\xi_{ts}, r_{it}, p_t, k} \quad & \sum_{t=1}^{T} p_t \mathbb{I}_{\left[\sum_{i=1}^{N} w_i r_{it} \leq x_k\right]} - \sum_{t=1}^{T} p_t \mathbb{I}_{\left[\sum_{i=1}^{N} \tau_i r_{it} \leq x_k\right]} & \\
\text{subject to } & x_k = \sum_{i=1}^{N} w_i r_{ik}, & k = 1, ..., T \\
& \sum_{t=1}^{T} \sum_{s=1}^{T} \xi_{ts} \left\| \boldsymbol{r}_t^0 - \boldsymbol{r}_s \right\| \leq \epsilon & \\
& \sum_{t=1}^{T} \xi_{ts} = \frac{1}{T}, & t = 1, ..., T \\
& \sum_{t=1}^{T} \xi_{ts} = p_s, & s = 1, ..., T \\
& \xi_{ts} \geq 0, & t, s = 1, ..., T \\
& r_{it} \geq -1, & i = 1, ..., N; t = 1, ..., T \\
& p_t \geq 0, & t = 1, ..., T.
\end{aligned}
\tag{2.21}
$$

Now we use the big M to rewrite the indicators using binary variables. We deal with the problem $\max \mathbb{I}_{\left[\sum_{i=1}^{N} w_i r_{it} \leq x_k\right]}$, we reformulate it using u_{tk} representing the indicator:

$$
\begin{aligned}
\max \ & u_{tk} & \\
\text{subject to } & \sum_{i=1}^{N} w_i r_{it} \leq x_k + (1 - u_{tk})M, & t, k = 1, ..., T \\
& u_{tk} \in \{0, 1\}, & t, k = 1, ..., T,
\end{aligned}
\tag{2.22}
$$

where M is sufficiently large constant. For the case of the second indicator, there is minus in front of it, which makes it much more difficult to handle. We need to use the inverse inequality, which in this case is sharp inequality. We get a reformulation for $\max -\mathbb{I}_{\left[\sum_{i=1}^{N} \tau_i r_{it} \leq x_k\right]}$:

$$
\begin{aligned}
\max \ & -v_{tk} & \\
\text{subject to } & \sum_{i=1}^{N} \tau_i r_{it} > x_k - v_{tk}M, & t, k = 1, ..., T \\
& v_{tk} \in \{0, 1\}, & t, k = 1, ..., T.
\end{aligned}
\tag{2.23}
$$

So we get a reformulated constraint $\sum_{i=1}^{N} \tau_i r_{it} \geq m + x_k - v_{tk}M, \ t, k = 1, ..., T$.

We use again the small positive constant m. We also use the same norm as in the case of the SSD. Objective function and the first constraint are again merged for implementation purposes. This of use the m is similar to the case of the SSD. By using a sharp inequality, then maximum might not be attained, so we approximate it. This time we cannot avoid it by setting a lower bound for probabilities, but the m basically sets the minimal recognizable difference between the returns of our and the benchmark portfolios. By setting m to zero, we would consider the same returns (imagine $x_t = y_s$ for some s, t) higher for the benchmark (even though they are the same), this is the limit version for $m \to 0+$. We can do this and bear in mind that the returns of benchmark that are the same as some returns of our portfolio actually mean that they are infinitesimally higher for the actual worst case distribution. By setting $m = 0$ enlarge the set of feasible distributions so if robust FSD holds for even larger set, then it certainly holds for the smaller set.

$$
\begin{aligned}
\max_{\xi_{ts}, r_{it}, p_t, u_{tk}, v_{tk}, b_k} \quad & \sum_{k=1}^{T} \left(\sum_{t=1}^{T} p_t (u_{tk} - v_{tk}) \right) \cdot b_k && \\
\text{subject to } x_k = & \sum_{i=1}^{N} w_i r_{ik}, && k = 1, ..., T \\
& \sum_{t=1}^{T} \sum_{s=1}^{T} \xi_{ts} \sum_{i=1}^{N} (r_{it}^0 - r_{is})^2 \le \epsilon && \\
& \sum_{s=1}^{T} \xi_{ts} = \frac{1}{T}, && t = 1, ..., T \\
& \sum_{t=1}^{T} \xi_{ts} = p_s, && s = 1, ..., T \\
& \sum_{i=1}^{N} w_i r_{it} \le x_k + (1 - u_{tk}) M, && t, k = 1, ..., T \\
& \sum_{i=1}^{N} \tau_i r_{it} \ge m + x_k - v_{tk} M, && t, k = 1, ..., T \\
& \xi_{ts} \ge 0, && t, s = 1, ..., T \\
& u_{tk} \in \{0, 1\}, && t, k = 1, ..., T \\
& v_{tk} \in \{0, 1\}, && t, k = 1, ..., T \\
& r_{it} \ge -1, && i = 1, ..., N; t = 1, ..., T \\
& b_k \ge 0, && k = 1, ..., T \\
& \sum_{k=1}^{T} b_k = 1 && \\
& p_t \ge 0, && t = 1, ..., T.
\end{aligned}
\tag{2.24}
$$

As for the SSD, we can understand the optimal values of r_{it} and p_t defining a distribution as the worst case distribution in the sense of definition 18. We formulate the derived results in the following theorem.

Theorem 8. *Let X and Y be random variables denoting returns of a portfolio defined by weights $\boldsymbol{w}$ and $\boldsymbol{\tau}$. Let us have observed historical returns $r_{it}^0, i = 1, ..., N; t = 1, ..., T$. Let us have Q a set of probability measures defined on $\mathbb{R}^N$ with T atoms: determined by returns $r_{it}, i = 1, ..., N; t = 1, ..., T$ and probabilities $p_t, t = 1, ..., T$ defined as a neighborhood of the empirical distribution. Let the neighborhood be defined with the use of the Wasserstein distance and let the distance on $\mathbb{R}^N$ be defined as Euclidean norm squared, i.e. let $x, y \in \mathbb{R}^N$, then $d(x, y) = \sum_{i=1}^{N} (x_i - y_i)^2$. Then X dominates robustly Y in the first order over the* *set of probability measures Q ($X \geq_{FSD}^{Q} Y$) if and only if there exists a right open neighborhood of 0 such that for each value m from this neighborhood the optimal value of the problem* (2.24) *is less than or equal to zero.*

As in the previous case, we can also use the program with $m = 0$ and then optimal value being less than or equal to zero is a sufficient condition for the robust stochastic dominance. Also if we find m for which there is a feasible solution and the objective value is positive, then the robust stochastic dominance cannot hold.

Note that we cannot derive an alternative formulation in the same way as we did for the SSD, because π_{ts} in the formulation from Luedtke [2008] are in that case binary and we cannot use the dual formulation for linear programming.

For all the robust tests we derived, we can generalize the test for probabilities different from $1/T$. Let us have general probabilities $p_t^0, t = 1, ..., T$ satisfying $\sum_{t=1}^{T} p_t^0 = 1$, then if we look at the program, the only place we use the observed distribution is in the Wasserstein distance, so we can easily generalize the program by replacing the constraint $\sum_{s=1}^{T} \xi_{ts} = \frac{1}{T}, t = 1, ..., T$ by the constraint $\sum_{s=1}^{T} \xi_{ts} = p_t^0, t = 1, ..., T$.

2.4 Robustification of programs with stochastic dominance constraints

In this section, we want to change the programs stated in section 1.2.4 so they would find distributionally robust solutions.

We could easily add the derived tests to the constraints. Let us denote the value of objective function of the test (optimal value) as a function of $\boldsymbol{w}$ as $g(\boldsymbol{w})$. We can formulate the optimization program as:

$$\begin{aligned} &\max_{w} f(\boldsymbol{w}) \\ &\text{subject to } g(\boldsymbol{w}) \leq 0 \\ &\boldsymbol{w} \in W. \end{aligned} \tag{2.25}$$

The problem is that this is a bilevel program, because we have already an optimization program in the constraint and there is no easy way to solve such problem. For this reason, we tried to derive computationally tractable programs directly from the formulations for non-robust stochastic dominance.

We would like to find at least some distributionally robust portfolios and we do not consider right away the most general case, where we define the neighborhood of distribution using the Wasserstein distance. Such problem is again very complicated to solve, so we consider only changes in returns or only in probabilities.

2.4.1 First order stochastic dominance

We start by the cases where we allow only the returns $r_{it}, i = 1, ..., N; t = 1, ..., T$ or only the probabilities $p_t, t = 1, ..., T$ defining the distribution of returns of the assets to change. For these changes, it is not necessary to use the Wasserstein metrics and we can use just the sum of absolute values of differences (from the observed values). Sum of absolute values is chosen because it can be reformulated as a linear constraint.

Robustness in returns

We use the formulation stated in problem (1.7). The key part is the fact that the probabilities remain $1/T$, which simplifies the problem. So now just the returns of both our and benchmark portfolio can change. We use the same formulation but we add new variables $r_{it}, i = 1, ..., N; t = 1, ..., T$ and again denote the observed values by $r_{it}^0, i = 1, ..., N; t = 1, ..., T$. Now the benchmark portfolio returns in scenarios are changing with the change of the returns of the underlying assets, let it have weights $\boldsymbol{\tau}$, so the returns are $y_t = \sum_{i=1}^N \tau_i r_{it}, t = 1, ..., T$.

We use the conditions in problem (1.7):

$$\begin{aligned}
&\sum_{s=1}^T \pi_{ts} \sum_{i=1}^N \tau_i r_{is} - \sum_{i=1}^N w_i r_{it} \leq 0, && t = 1, ..., T \\
&\sum_{s=1}^T \pi_{ts} = 1, && t = 1, ..., T \\
&\sum_{t=1}^T \pi_{ts} = 1, && s = 1, ..., T \\
&\pi_{ts} \in \{0, 1\}, && t = 1, ..., T; s = 1, ..., T \\
&\boldsymbol{w} \in W,
\end{aligned} \tag{2.26}$$

We use the fact that the inequality must hold for all r_{it}, if the inequality is fulfilled for maximum then it is certainly fulfilled for every possible choice. This must be true for all values of $t \in \{1, ..., T\}$. We state the general problem of robust stochastic dominance in constraints. The minimization is there because we use the permutation to achieve the fulfillment of the constraints. If the condition holds for minimum then there exists one permutation for which the conditions hold.

$$\begin{aligned}
\max_{r_{it}} \min_{\pi_{ts}} \max_{t} &\left(\sum_{s=1}^T \pi_{ts} \sum_{i=1}^N \tau_i r_{is} - \sum_{i=1}^N w_i r_{it} \right) \leq 0 && \\
\text{subject to } &\sum_{i=1}^N \sum_{t=1}^T |r_{it}^0 - r_{it}| \leq \epsilon && \\
&\sum_{s=1}^T \pi_{ts} = 1, && t = 1, ..., T \\
&\sum_{t=1}^T \pi_{ts} = 1, && s = 1, ..., T \\
&r_{it} \geq -1, && i = 1, ..., N; t = 1, ..., T \\
&\pi_{ts} \in \{0, 1\}, && t, s = 1, ..., T.
\end{aligned} \tag{2.27}$$

We want to find such weights $\boldsymbol{w}$ that maximize $f(\boldsymbol{w})$ (we define the possible choices of the function later) and for each set of $r_{it}, i = 1, ..., N; t = 1, ..., T$ such that $\sum_{i=1}^N \sum_{t=1}^T |r_{it}^0 - r_{it}| \leq \epsilon$ there exist $\pi_{ts}^r \in \{0, 1\}, t = 1, ..., T; s = 1, ..., T$

such that $\sum_{i=1}^{N} w_i r_{it} \geq \sum_{s=1}^{T} \pi_{ts}^{r} \sum_{i=1}^{N} \tau_i r_{is}, t = 1, ..., T$, where we use ^{r} to denote the dependence of elements of the permutation matrix π_{ts} on return values r_{it}. This presents a bilevel optimization problem, which is generally very hard to solve. To approximate the problem, we remove the dependence of permutation matrix on the value of return. If we choose ϵ sufficiently small the permutation needed for the rearranging of the returns does not change (which is connected to the fact that the sorting of the returns does not change). By removing the dependence of π_{ts}, we might lose some of the possible solutions, because for some weights $\boldsymbol{w}$ and returns r_{it} a different permutation is needed to ensure that the constraint is satisfied (and there must also exist such permutation). To present a case described above, it would mean that one scenario of benchmark portfolio (y_1) exceeds its assigned counterpart (x_1) of our portfolio and at the same time, there exists reordering such that the return of our portfolio exceeds the returns of the new corresponding return of the benchmark, i.e. the FSD condition holds. In other words we neglect some possible changes in ordering of the returns. The solution achieved by solving the program will certainly be robust in the FSD, but might not be optimal in the value of the objective function.

Using the approximation described above. It basically means that we change the order of outer minimum and maximum, it always holds that (which can be proved by starting with $(\cdot) \leq \max_{r_{it}}(\cdot)$ and applying min and max).

$$\max_{r_{it}} \min_{\pi_{ts}}(\cdot) \leq \min_{\pi_{ts}} \max_{r_{it}}(\cdot).$$

So when we check the constraint for the greater value that it most certainly holds for the smaller value. After changing the order, we can remove the minimum because it is enough that just one such set of π_{ts} exists. Then we check the condition for each t, because it must hold for maximum.

Note that we made the constraint stricter, which means we made the set of possible weights smaller. This is a conservative approach as it is stricter than robust stochastic dominance. This mathematical approximation corresponds to the consideration described above. The optimal value will be less than or equal to the one without approximation as we have a maximization problem and a smaller set of feasible solutions (for example smaller expected return), but the constraint is stronger than robust stochastic dominance. Also notice that the approximation is better for smaller values of ϵ because than the returns do not differ much and there is no need to change the ordering. So we now get:

$$
\begin{aligned}
&\max_{w_i, \pi_{ts}} \ f(\boldsymbol{w}) \\
&\text{subject to} \ \max_{r_{it}} \sum_{s=1}^{T} \pi_{ts} \sum_{i=1}^{N} \tau_i r_{is} - \sum_{i=1}^{N} w_i r_{it} \leq 0, \quad t = 1, ..., T \\
&\qquad \sum_{i=1}^{N} \sum_{t=1}^{T} |r_{it}^0 - r_{it}| \leq \epsilon \\
&\qquad \sum_{s=1}^{T} \pi_{ts} = 1, \qquad t = 1, ..., T \\
&\qquad \sum_{t=1}^{T} \pi_{ts} = 1, \qquad s = 1, ..., T \\
&\qquad \pi_{ts} \in \{0, 1\}, \qquad t = 1, ..., T; s = 1, ..., T \\
&\qquad r_{it} \geq -1, \qquad i = 1, ..., N; t = 1, ..., T \\
&\qquad \boldsymbol{w} \in W.
\end{aligned}
\tag{2.28}
$$

For a specific value of t, we look into the inner optimization problem (we now use index s for scenarios in the list of the variables under max so we avoid any confusion):

$$
\begin{aligned}
&\max_{r_{is}} \ \sum_{s=1}^{T} \pi_{ts} \sum_{i=1}^{N} \tau_i r_{is} - \sum_{i=1}^{N} w_i r_{it} \\
&\text{subject to} \ \sum_{i=1}^{N} \sum_{s=1}^{T} |r_{is}^0 - r_{is}| \leq \epsilon \\
&\qquad r_{is} \geq -1, \qquad i = 1, ..., N; s = 1, ..., T.
\end{aligned}
\tag{2.29}
$$

Note that for each value of t (different constraint), we get a different optimal solution (r_{is}), but this does not matter as we do not need the optimal value, we just want it to be smaller than a predefined value. We reformulate the problem:

$$
\begin{aligned}
&\max_{r_{is}} \ \sum_{s=1}^{T} \pi_{ts} \sum_{i=1}^{N} \tau_i (r_{is} - r_{is}^0) + \sum_{s=1}^{T} \pi_{ts} \sum_{i=1}^{N} \tau_i r_{is}^0 \\
&\qquad - \sum_{i=1}^{N} w_i (r_{it} - r_{it}^0) - \sum_{i=1}^{N} w_i r_{it}^0 \\
&\text{subject to} \ \sum_{i=1}^{N} \sum_{s=1}^{T} |r_{is}^0 - r_{is}| \leq \epsilon \\
&\qquad r_{is} \geq -1, \qquad i = 1, ..., N; s = 1, ..., T.
\end{aligned}
\tag{2.30}
$$

First, we remove the absolute values. Let us define new variable as $a_{it} = r_{it} - r_{it}^0$. The constraint is now $a_{it} \geq -1 - r_{it}^0$. Now we can plug in the a_{it}. We can rewrite the inner optimization problem as:

$$
\begin{aligned}
\max_{a_{is}} \quad & \sum_{s=1}^{T} \pi_{ts} \sum_{i=1}^{N} \tau_i a_{is} + \sum_{s=1}^{T} \pi_{ts} \sum_{i=1}^{N} \tau_i r_{is}^0 - \sum_{i=1}^{N} w_i a_{it} - \sum_{i=1}^{N} w_i r_{it}^0 \\
\text{subject to} \quad & \sum_{i=1}^{N} \sum_{s=1}^{T} |a_{is}| \leq \epsilon \\
& a_{is} \geq -1 - r_{is}^0, \quad i = 1, ..., N; s = 1, ..., T.
\end{aligned}
\tag{2.31}
$$

Now we have a_{it} directly in the optimization problem, moreover we disintegrate a_{it} to deal with the absolute value: $a_{it} = a_{it}^+ - a_{it}^-, a_{it}^+, a_{it}^- \geq 0$ and we replace $|a_{it}|$ by $a_{it}^+ + a_{it}^-$. The absolute value is only in the constraint so we do not mind the problem with $a_{it} = a_{it}^+ - a_{it}^-, a_{it}^+, a_{it}^- \geq 0$ not being unique. When a set of a_{it}^+, a_{it}^- fulfills the constraint, then also a_{it} must fulfill the constraint because minimum of the absolute value is achieved when one of the variables a_{it}^+, a_{it}^- is zero and this case corresponds to the case $|a_{it}| = a_{it}^+ + a_{it}^-$. Otherwise $|a_{it}| < a_{it}^+ + a_{it}^-$ so the constraints is also fulfilled. On the other hand, for every a_{it} there exist a_{it}^+, a_{it}^- such that $a_{it} = a_{it}^+ - a_{it}^-$ and $|a_{it}| = a_{it}^+ + a_{it}^-$, this corresponds with the case when one of the variables is zero. Moreover a part of the objective function now does not depend on a_{it} so we remove it from the problem and add it to the RHS of the corresponding constraint in formulation (2.28). We get a linear programming formulation of the problem:

$$
\begin{aligned}
\max_{a_{is}^+, a_{is}^-} \quad & \sum_{s=1}^{T} \pi_{ts} \sum_{i=1}^{N} \tau_i (a_{is}^+ - a_{is}^-) - \sum_{i=1}^{N} w_i (a_{it}^+ - a_{it}^-) \\
\text{subject to} \quad & \sum_{i=1}^{N} \sum_{s=1}^{T} (a_{is}^+ + a_{is}^-) \leq \epsilon \\
& a_{is}^+ - a_{is}^- \geq -1 - r_{is}^0, \quad i = 1, ..., N; s = 1, ..., T \\
& a_{is}^+, a_{is}^-, \geq 0 \qquad\qquad\quad i = 1, ..., N; s = 1, ..., T.
\end{aligned}
\tag{2.32}
$$

We get the dual formulation for a fixed $t \in \{1, ..., T\}$:

$$
\begin{aligned}
\min_{u_{ist}, a_t} \quad & a_t \epsilon + \sum_{i=1}^{N} \sum_{s=1}^{T} u_{ist} (1 + r_{is}^0) \\
\text{subject to} \quad & a_t - u_{ist} \geq \pi_{ts} \tau_i - w_i, \quad && t = s; s = 1, ..., T; i = 1, ..., N \\
& a_t + u_{ist} \geq -\pi_{ts} \tau_i + w_i, && t = s; s = 1, ..., T; i = 1, ..., N \\
& a_t - u_{ist} \geq \pi_{ts} \tau_i, && t \neq s; s = 1, ..., T; i = 1, ..., N \\
& a_t + u_{ist} \geq -\pi_{ts} \tau_i, && t \neq s; s = 1, ..., T; i = 1, ..., N \\
& a_t \geq 0 \\
& u_{ist} \geq 0 && i = 1, ..., N; s = 1, ..., T.
\end{aligned}
\tag{2.33}
$$

Now we plug the dual formulation into the main program and we can remove the minimum as we did many times before. We use the corresponding dual formulation for each value of $t \in \{1, ..., T\}$ (and also we plug in the removed parts):

$$
\begin{aligned}
\max_{w_i,\pi_{ts},a_t,u_{its}} \quad & f(\boldsymbol{w}) \\
\text{subject to} \quad & a_t\epsilon + \sum_{i=1}^{N}\sum_{s=1}^{T} u_{ist}(1+r_{is}^0) \leq -\sum_{s=1}^{T}\pi_{ts}\sum_{i=1}^{N}\tau_i r_{is}^0 + \sum_{i=1}^{N} w_i r_{it}^0, \\
& \qquad\qquad\qquad\qquad\qquad t = 1,...,T \\
& a_t - u_{ist} \geq \pi_{ts}\tau_i - w_i, \qquad t = s; t,s = 1,...,T; i = 1,...,N \\
& a_t + u_{ist} \geq -\pi_{ts}\tau_i + w_i, \quad t = s; t,s = 1,...,T; i = 1,...,N \\
& a_t - u_{ist} \geq \pi_{ts}\tau_i, \qquad t \neq s; t,s = 1,...,T; i = 1,...,N \\
& a_t + u_{ist} \geq -\pi_{ts}\tau_i, \qquad t \neq s; t,s = 1,...,T; i = 1,...,N \\
& \sum_{s=1}^{T}\pi_{ts} = 1, \qquad t = 1,...,T \\
& \sum_{t=1}^{T}\pi_{ts} = 1, \qquad s = 1,...,T \\
& \pi_{ts} \in \{0,1\}, \qquad t = 1,...,T; s = 1,...,T \\
& a_t \geq 0, \qquad t = 1,...,T \\
& u_{ist} \geq 0 \qquad i = 1,...,N; t,s = 1,...,T \\
& \boldsymbol{w} \in W.
\end{aligned}
\tag{2.34}
$$

Now to determine what to use as the function $f(\boldsymbol{w})$, first we can consider just the expected return with respect to the original distribution with returns r_{it}^0:

$$f(\boldsymbol{w}) = \sum_{t=1}^{T} p_t \sum_{i=1}^{N} w_i r_{it}^0.$$

We could also consider robust approach in the objective function and use the worst expected return from the set of probability distributions that differ from the empirical one in the same way as we defined in the constraints, i.e. $\sum_{i=1}^{N}\sum_{t=1}^{T}|r_{it}^0 - r_{it}| \leq \epsilon$. We want to be prepared for the worst distribution from the considered set and maximize our expected return under such distribution.

$$
\begin{aligned}
\min_{r_{it}} \quad & \sum_{t=1}^{T} p_t \sum_{i=1}^{N} w_i r_{it} \\
\text{subject to} \quad & \sum_{i=1}^{N}\sum_{t=1}^{T}|r_{it}^0 - r_{it}| \leq \epsilon \\
& r_{it} \geq -1, \qquad i = 1,...,N; t = 1,...,T.
\end{aligned}
\tag{2.35}
$$

Now we must use the same procedure as above to obtain dual formulation.

We rewrite the objective function to obtain difference from the original value.

$$\begin{aligned}
\min_{r_{it}} \quad & \sum_{t=1}^{T} p_t \sum_{i=1}^{N} w_i (r_{it} - r_{it}^0) + \sum_{t=1}^{T} p_t \sum_{i=1}^{N} w_i r_{it}^0 \\
\text{subject to} \quad & \sum_{i=1}^{N} \sum_{t=1}^{T} |r_{it}^0 - r_{it}| \leq \epsilon \\
& r_{it} \geq -1, \qquad\qquad i = 1, ..., N; t = 1, ..., T.
\end{aligned} \tag{2.36}$$

We plug in b_{it} for the difference and use negative and positive parts to obtain linear programming formulation.

$$\begin{aligned}
\min_{b_{it}^+, b_{it}^-} \quad & \sum_{t=1}^{T} p_t \sum_{i=1}^{N} w_i (b_{it}^+ - b_{it}^-) + \sum_{t=1}^{T} p_t \sum_{i=1}^{N} w_i r_{it}^0 \\
\text{subject to} \quad & \sum_{i=1}^{N} \sum_{t=1}^{T} (b_{it}^+ + b_{it}^-) \leq \epsilon \\
& b_{it}^+ - b_{it}^- \geq -1 - r_{it}^0, \qquad\qquad i = 1, ..., N; t = 1, ..., T \\
& b_{it}^+, b_{it}^-, \geq 0 \qquad\qquad i = 1, ..., N; t = 1, ..., T.
\end{aligned} \tag{2.37}$$

We get a dual formulation (not considering the part that does not depend on the variables):

$$\begin{aligned}
\max_{c, d_{it}} \quad & c\epsilon + \sum_{i=1}^{N} \sum_{t=1}^{T} d_{it} (1 + r_{it}^0) \\
\text{subject to} \quad & c - d_{it} \leq p_t w_i, \qquad t = 1, ..., T; i = 1, ..., N \\
& c + d_{it} \leq -p_t w_i, \qquad t = 1, ..., T; i = 1, ..., N \\
& c \leq 0 \\
& d_{it} \leq 0 \qquad i = 1, ..., N; t = 1, ..., T.
\end{aligned} \tag{2.38}$$

We can now join the maximum with the one in the original problem and receive a program robust in both the objective function and the constraints. We also plug $1/T$ for p_t.

$$
\begin{aligned}
\max_{w_i,\pi_{ts},a_t,u_{its},c,d_{it}} \quad & c\epsilon + \sum_{i=1}^{N}\sum_{t=1}^{T} d_{it}(1+r_{it}^0) + \sum_{t=1}^{T}\frac{1}{T}\sum_{i=1}^{N} w_i r_{it}^0 \\
\text{subject to} \quad & a_t\epsilon + \sum_{i=1}^{N}\sum_{s=1}^{T} u_{ist}(1+r_{is}^0) \le -\sum_{s=1}^{T}\pi_{ts}\sum_{i=1}^{N}\tau_i r_{is}^0 + \sum_{i=1}^{N} w_i r_{it}^0, \\
& \qquad\qquad\qquad\qquad\qquad\qquad t = 1,...,T \\
& a_t - u_{ist} \ge \pi_{ts}\tau_i - w_i, \qquad t = s; t,s = 1,...,T; i = 1,...,N \\
& a_t + u_{ist} \ge -\pi_{ts}\tau_i + w_i, \quad t = s; t,s = 1,...,T; i = 1,...,N \\
& a_t - u_{ist} \ge \pi_{ts}\tau_i, \qquad\qquad t \ne s; t,s = 1,...,T; i = 1,...,N \\
& a_t + u_{ist} \ge -\pi_{ts}\tau_i, \qquad\quad t \ne s; t,s = 1,...,T; i = 1,...,N \\
& c - d_{it} \le \frac{1}{T}w_i, \qquad\qquad t = 1,...,T; i = 1,...,N \\
& c + d_{it} \le -\frac{1}{T}w_i, \qquad\quad t = 1,...,T; i = 1,...,N \\
& \sum_{s=1}^{T}\pi_{ts} = 1, \qquad\qquad t = 1,...,T \\
& \sum_{t=1}^{T}\pi_{ts} = 1, \qquad\qquad s = 1,...,T \\
& \pi_{ts} \in \{0,1\}, \qquad\qquad t = 1,...,T; s = 1,...,T \\
& a_t \ge 0, \qquad\qquad\qquad t = 1,...,T \\
& u_{ist} \ge 0 \qquad\qquad\qquad i = 1,...,N; t,s = 1,...,T \\
& c \le 0 \\
& d_{it} \le 0 \qquad\qquad\qquad i = 1,...,N; t = 1,...,T \\
& \boldsymbol{w} \in W.
\end{aligned}
\tag{2.39}
$$

Again if one would like to consider general probabilities p_t instead of the usual $1/T$, we could use the more general formulation (2.40), which we use in the next section. When we look at the difference for the stochastic dominance constraints we can see that just the constraint $\sum_{t=1}^{T}\pi_{ts} = 1, s = 1,...,T$ is replaced by the constraint $\sum_{t=1}^{T} p_t \sum_{s=1}^{k-1}\pi_{ts} \le \sum_{s=1}^{k-1} p_s, k = 2,...,T$. But this formulation can be used only for ordered observations. We would have to add another permutation matrix to order the returns which would greatly increase the complexity of the problem. For the robustness in the objective function and general probabilities, we would not plug in the $1/T$ as we did in the last step which means replacing $c - d_{it} \le \frac{1}{T}w_i, t = 1,...,T; i = 1,...,N$ by $c - d_{it} \le p_t w_i, t = 1,...,T; i = 1,...,N$ and changing the other inequality in the same way.

Robustness in probabilities

Now we want to derive a program that would be robust in probabilities p_t while the returns r_{it} are fixed. We use one of the formulations of equivalent FSD conditions from Luedtke [2008]. It is very similar to the formulation (1.8), but the two of the conditions are merged and probabilities are the same for both random variables. We denote the observed returns r_{it} as there is now no need to use the ^{0} notation, those are not variables to the problem but parameters as they are fixed. Let us now use the observed probabilities p_t^0 that can differ from $1/T$, it would be a little harder to adjust the final program for this case (but still very straightforward). In this formulation, the benchmark returns must be ordered, i.e. $y_1 < ... < y_T$

$$
\begin{aligned}
\max \;& f(\boldsymbol{w}) && \\
\text{subject to } & \sum_{s=1}^{T} \pi_{ts} = 1, && t = 1, ..., T \\
& \sum_{i=1}^{N} w_i r_{it} \geq \sum_{s=1}^{T} y_s \pi_{ts}, && t = 1, ..., T \\
& \sum_{t=1}^{T} p_t \sum_{s=1}^{k-1} \pi_{ts} \leq \sum_{t=1}^{k-1} p_t, && k = 2, ..., T \\
& \pi_{ts} \in \{0, 1\}, && t = 1, ..., T; s = 1, ..., T \\
& \boldsymbol{w} \in W.
\end{aligned}
\tag{2.40}
$$

We can notice that only changes in probabilities cannot influence the ordering of scenarios. This is crucial, because we are now able to use this formulation. Moreover it does not influence the returns, which are compared in the second constraint as we can see it does not depend on p_t. So the only constraint we have to deal with when we want to allow probabilities p_t to vary (add robustness) is the last constraint with inequality. Even though the ordering does not change, with different probabilities π_{ts} is no longer a permutation matrix, it can assign multiple scenarios of our portfolio to just one scenario of the benchmark. This means the π_{ts} can depend on the specific values of probabilities p_t (different assignments might be needed). We use the same approach as above when considering robustness in returns and interchange the order of max (over p_t) and min (over π_{ts}), which removes the dependence on the values π_{ts}. We use again an approximation but this time it is different, even for very small differences in probabilities, the return of our portfolio in some of the scenarios might have to be drastically changed in order to satisfy the dominance. But it does hold that for smaller values of ϵ the approximation is better, there just might be some discontinuities in the behavior.

$$
\begin{aligned}
\max_{w_i,\pi_{ts}} \ & f(\boldsymbol{w}) \\
\text{subject to } & \max_{p_t} \sum_{t=1}^{T} p_t \sum_{s=1}^{k-1} \pi_{ts} - \sum_{t=1}^{k-1} p_t \leq 0, \quad k = 2, ..., T \\
& \sum_{t=1}^{T} \left| p_t - p_t^0 \right| \leq \epsilon \\
& \sum_{t=1}^{T} p_t = 1 \\
& \sum_{i=1}^{N} w_i r_{it} \geq \sum_{s=1}^{T} y_s \pi_{ts}, \qquad t = 1, ..., T \\
& \sum_{s=1}^{T} \pi_{ts} = 1, \qquad t = 1, ..., T \\
& \pi_{ts} \in \{0, 1\}, \qquad t = 1, ..., T; s = 1, ..., T \\
& p_t \geq m, \qquad t = 1, ..., T \\
& \boldsymbol{w} \in W.
\end{aligned} \tag{2.41}
$$

We restrain the sum of changes by ϵ. Again we must have all probabilities positive so the conditions make sense. We use m (margin) as a small positive constant. We must also make sure that p_t are probabilities so that they sum to 1. Again we understand the constraint with maximum that it must hold for all $k, k = 2, ..., T$.

Let us have a look at the inner optimization program for a fixed k:

$$
\begin{aligned}
& \max_{p_t} \sum_{t=1}^{T} p_t \sum_{s=1}^{k-1} \pi_{ts} - \sum_{t=1}^{k-1} p_t \\
\text{subject to } & \sum_{t=1}^{T} \left| p_t - p_t^0 \right| \leq \epsilon \\
& \sum_{t=1}^{T} p_t = 1 \\
& p_t \geq m, \qquad t = 1, ..., T.
\end{aligned} \tag{2.42}
$$

We again want to shift the variables to the originally observed point, $q_t = p_t - p_t^0$. We rewrite the problem as:

$$
\begin{aligned}
& \max_{p_t} \sum_{t=1}^{T} \left(p_t - p_t^0\right) \sum_{s=1}^{k-1} \pi_{ts} + \sum_{t=1}^{T} p_t^0 \sum_{s=1}^{k-1} \pi_{ts} - \sum_{t=1}^{k-1} \left(p_t - p_t^0\right) - \sum_{t=1}^{k-1} p_t^0 \\
\text{subject to } & \sum_{t=1}^{T} \left| p_t - p_t^0 \right| \leq \epsilon \\
& \sum_{t=1}^{T} \left(p_t - p_t^0\right) = 0 \\
& p_t - p_t^0 \geq m - p_t^0, \quad t = 1, ..., T.
\end{aligned} \tag{2.43}
$$

Now we plug in the q_t and remove the parts which do not depend on it, which will be added back later.

$$\begin{aligned}
\max_{q_t} \quad & \sum_{t=1}^{T} q_t \sum_{s=1}^{k-1} \pi_{ts} - \sum_{t=1}^{k-1} q_t \\
\text{subject to} \quad & \sum_{t=1}^{T} |q_t| \leq \epsilon \\
& \sum_{t=1}^{T} q_t = 0 \\
& q_t \geq m - p_t^0 \qquad\qquad t = 1, ..., T.
\end{aligned} \tag{2.44}$$

For the same reasons as above we can use positive and negative parts q_t^+, q_t^- to get a linear programming formulation.

$$\begin{aligned}
\max_{q_t^+, q_t^-} \quad & \sum_{t=1}^{T} (q_t^+ - q_t^-) \sum_{s=1}^{k-1} \pi_{ts} - \sum_{t=1}^{k-1} (q_t^+ - q_t^-) \\
\text{subject to} \quad & \sum_{t=1}^{T} (q_t^+ + q_t^-) \leq \epsilon \\
& \sum_{t=1}^{T} (q_t^+ - q_t^-) = 0 \\
& (q_t^+ - q_t^-) \geq m - p_t^0, \qquad\qquad t = 1, ..., T \\
& q_t^+, q_t^- \geq 0, \qquad\qquad t = 1, ..., T.
\end{aligned} \tag{2.45}$$

The dual formulation is

$$\begin{aligned}
\min_{a_k, b_k, c_{tk}} \quad & a_k \epsilon + \sum_{t=1}^{T} \left(p_t^0 - m\right) c_{tk} \\
\text{subject to} \quad & a_k + b_k - c_{tk} \geq \sum_{s=1}^{k-1} \pi_{ts} - 1, \quad 1 \leq t \leq k-1 \\
& a_k - b_k + c_{tk} \geq \sum_{s=1}^{k-1} -\pi_{ts} + 1, \quad 1 \leq t \leq k-1 \\
& a_k + b_k - c_{tk} \geq \sum_{s=1}^{k-1} \pi_{ts}, \quad T \geq t > k-1 \\
& a_k - b_k + c_{tk} \geq \sum_{s=1}^{k-1} -\pi_{ts}, \quad T \geq t > k-1 \\
& c_{tk} \geq 0, \quad t = 1, ..., T \\
& a_k \geq 0 \\
& b_k \in \mathbb{R}.
\end{aligned} \tag{2.46}$$

Now we plug in the dual formulation for each k as we did the the case of returns and get the final formulation (we also remove the minimum and add the terms we skipped when deriving the dual formulation):

$$
\begin{aligned}
&\max_{w_i,\pi_{ts},a_k,b_k,c_{tk}} f(\boldsymbol{w}) \\
&\text{s.t.} a_k\epsilon + \sum_{t=1}^{T}\left(p_t^0 - m\right)c_{tk} \le -\sum_{t=1}^{T}p_t^0\sum_{s=1}^{k-1}\pi_{ts} + \sum_{s=1}^{k-1}p_s^0,\ k=2,...,T \\
&a_k + b_k - c_{tk} \ge \sum_{s=1}^{k-1}\pi_{ts} - 1, \quad 1 \le t \le k-1; k=2,...,T \\
&a_k - b_k + c_{tk} \ge \sum_{s=1}^{k-1} -\pi_{ts} + 1, \quad 1 \le t \le k-1; k=2,...,T \\
&a_k + b_k - c_{tk} \ge \sum_{s=1}^{k-1}\pi_{ts}, \quad T \ge t > k-1; k=2,...,T \\
&a_k - b_k + c_{tk} \ge \sum_{s=1}^{k-1} -\pi_{ts}, \quad T \ge t > k-1; k=2,...,T \\
&\sum_{i=1}^{N} w_i r_{it} \ge \sum_{s=1}^{T} y_s\pi_{ts}, \quad t=1,...,T \\
&\sum_{s=1}^{T}\pi_{ts} = 1, \quad t=1,...,T \\
&\pi_{ts} \in \{0,1\}, \quad t=1,...,T; s=1,...,T \\
&c_{tk} \ge 0, \quad k=2,...,T; t=1,...,T \\
&a_k \ge 0, \quad k=2,...,T \\
&b_k \in \mathbb{R}, \quad k=2,...,T \\
&\boldsymbol{w} \in W.
\end{aligned}
\tag{2.47}
$$

Notice that the fact we want our probabilities to be strictly positive creates a very large number of additional variables. We could just remove all of them and bear in mind that even though some of the probabilities are set to zero, the returns in those scenarios are still considered. This is the same problem as we discussed in the chapter of worst case distributions. Using just non-negative probabilities would make the constraint a little stricter and would prepare our portfolio even for those cases. One should consider it as a very small probability scenario, not with zero probability. Moreover this formulation allows us to set m as a lower bound, which could be convenient.

Also notice that we must use ordered returns of benchmark, as it is stated in the assumptions of the formulation from Luedtke [2008]. This is not a problem, because the change in probabilities does not affect the ordering and it can be done prior to the computation.

For the objective function, now we can also add robustness to it. We could in general use any combination of robustness in returns and in probabilities in both objective function and in constraints, but this does not make much sense. We could also choose different values of ϵ for objective function and for the constraint, which means consider different radius of neighborhoods.

Let us state the problem for the objective function, we again want to be prepared for the worst case.

$$
\begin{aligned}
\min_{p_t} \quad & \sum_{t=1}^{T} p_t \sum_{i=1}^{N} w_i r_{it} \\
\text{subject to} \quad & \sum_{t=1}^{T} \left| p_t - p_t^0 \right| \leq \epsilon \\
& \sum_{t=1}^{T} p_t = 1 \\
& p_t \geq m, \qquad\qquad t = 1, ..., T.
\end{aligned} \tag{2.48}
$$

We again shift the probabilities by p_t^0 and get the linear programming formulation and then use the dual formulation:

$$
\begin{aligned}
\min_{q_t^+, q_t^-} \quad & \sum_{t=1}^{T} (q_t^+ - q_t^-) \sum_{i=1}^{N} w_i r_{it} + \sum_{t=1}^{T} p_t^0 \sum_{i=1}^{N} w_i r_{it} \\
\text{subject to} \quad & \sum_{t=1}^{T} (q_t^+ + q_t^-) \leq \epsilon \\
& \sum_{t=1}^{T} (q_t^+ - q_t^-) = 0 \\
& (q_t^+ - q_t^-) \geq m - p_t^0, \qquad\qquad t = 1, ..., T \\
& q_t^+, q_t^- \geq 0, \qquad\qquad t = 1, ..., T.
\end{aligned} \tag{2.49}
$$

The dual formulation (after removing the constants) is:

$$
\begin{aligned}
\max_{d,e,f_t} \quad & d\epsilon + \left(p_t^0 - m\right) \sum_{t=1}^{T} f_t \\
\text{subject to} \quad & d + e - f_t \leq \sum_{i=1}^{N} w_i r_{it}, \quad t = 1, ..., T \\
& d - e + f_t \leq - \sum_{i=1}^{N} w_i r_{it}, \quad t = 1, ..., T \\
& d \leq 0 \\
& f_t \leq 0, \qquad\qquad t = 1, ..., T.
\end{aligned} \tag{2.50}
$$

Now by merging the maximum with the one in the original problem and adding the removed part, we get the final formulation:

$$
\begin{aligned}
\max_{w_i,\pi_{ts},a_k,b_k,c_{tk},d,e,f_t} \quad & d\epsilon + \left(p_t^0 - m\right)\sum_{t=1}^{T} f_t + \sum_{t=1}^{T} p_t^0 \sum_{i=1}^{N} w_i r_{it} \\
\text{s.t.} a_k\epsilon + \sum_{t=1}^{T} \left(p_t^0 - m\right) c_{tk} \le & -\sum_{t=1}^{T} p_t^0 \sum_{s=1}^{k-1} \pi_{ts} + \sum_{s=1}^{k-1} p_s^0, \;\; k = 2, ..., T \\
& a_k + b_k - c_{tk} \ge \sum_{s=1}^{k-1} \pi_{ts} - 1, \quad 1 \le t \le k-1; k = 2, ..., T \\
& a_k - b_k + c_{tk} \ge \sum_{s=1}^{k-1} -\pi_{ts} + 1, \quad 1 \le t \le k-1; k = 2, ..., T \\
& a_k + b_k - c_{tk} \ge \sum_{s=1}^{k-1} \pi_{ts}, \quad T \ge t > k-1; k = 2, ..., T \\
& a_k - b_k + c_{tk} \ge \sum_{s=1}^{k-1} -\pi_{ts}, \quad T \ge t > k-1; k = 2, ..., T \\
& \sum_{i=1}^{N} w_i r_{it} \ge \sum_{s=1}^{T} y_s \pi_{ts}, \quad t = 1, ..., T \\
& \sum_{s=1}^{T} \pi_{ts} = 1, \quad t = 1, ..., T \\
& d + e - f_t \le \sum_{i=1}^{N} w_i r_{it}, \quad t = 1, ..., T \\
& d - e + f_t \le -\sum_{i=1}^{N} w_i r_{it}, \quad t = 1, ..., T \\
& \pi_{ts} \in \{0, 1\}, \quad t = 1, ..., T; s = 1, ..., T \\
& c_{tk} \ge 0, \quad k = 2, ..., T; t = 1, ..., T \\
& a_k \ge 0, \quad k = 2, ..., T \\
& b_k \in \mathbb{R}, \quad k = 2, ..., T \\
& d \le 0 \\
& f_t \le 0, \quad t = 1, ..., T \\
& \boldsymbol{w} \in W.
\end{aligned}
\tag{2.51}
$$

For our main use, which is the empirical distribution, we just plug $1/T$ for p_t^0 and we also use this choice in the empirical study.

2.4.2 Second order stochastic dominance

Let us again consider the robustness either in returns or in probabilities, which allows us to formulate the optimization problem in a computationally tractable way.

Robustness in returns

Let us use again a formulation from Luedtke [2008], which we already used in the section of worst case distributions. Now we use it for fixed probabilities $1/T$. We state the formulation to remind it.

Let us have X and Y discrete random variables attaining values $x_1, ..., x_T$ and $y_1, ..., y_T$ with probabilities $1/T$. Then $X \geq_{SSD} Y$ if and only if there exists $\pi \in \mathbb{R}_+^{T\times T}$ such that

$$\begin{aligned}\sum_{s=1}^{T} y_s \pi_{ts} - x_t \leq& 0, \quad t = 1, ..., T\\ \sum_{s=1}^{T} \pi_{ts} =& 1, \quad t = 1, ..., T\\ \sum_{t=1}^{T} \pi_{ts} =& 1, \quad s = 1, ..., T.\end{aligned} \tag{2.52}$$

Again as it was in the case of the FSD, we want that for every set of r_{it} satisfying $\sum_{i=1}^{N}\sum_{t=1}^{T} |r_{it}^0 - r_{it}| \leq \epsilon$ there exist π_{ts} satisfying (2.52). Similar to the case of the SSD worst case distribution, we use π_{ts} to minimize the difference in order to check whether there exists a set of π_{ts} that the condition is met. So we want the maximum over r_{it} of this minimum over π_{ts} to satisfy the inequality condition for each $t \in \{1, ..., T\}$. This means we want maximum over t to satisfy it.

$$\begin{aligned}&\max_{r_{it}} \min_{\pi_{ts}} \max_{t} \left(\sum_{s=1}^{T} \pi_{ts} \sum_{i=1}^{N} \tau_i r_{is} - \sum_{i=1}^{N} w_i r_{it} \right) \leq 0\\ &\quad \text{subject to } \sum_{i=1}^{N}\sum_{t=1}^{T} |r_{it}^0 - r_{it}| \leq \epsilon\\ &\qquad \sum_{s=1}^{T} \pi_{ts} = 1, && t = 1, ..., T\\ &\qquad \sum_{t=1}^{T} \pi_{ts} = 1, && s = 1, ..., T\\ &\qquad r_{it} \geq -1, && i = 1, ..., N; t = 1, ..., T\\ &\qquad \pi_{ts} \geq 0, && t, s = 1, ..., T.\end{aligned} \tag{2.53}$$

The formulation is identical to the robust FSD, but the π_{ts} do not have to be binary, just non-negative. This greatly reduces the complexity of the problem, but still we are not able to solve it. It leads to a bilevel problem again. We can now use the same approximation as above and all the steps would be the same as there is no difference. Just in the final program, the set from which we take π_{ts} is changed, i.e. $\pi_{ts} \geq 0$. On one hand, the problem is now less complex, but on the other hand, the approximation is worse. As the matrix π_{ts} now does not only contain information about the ordering but also about the transferring of extra returns from lower to higher ordered scenarios, we lose more feasible portfolios by neglecting all of this. This is connected to the fact that the set of possible values for π_{ts} is not only $\{0, 1\}$, but the whole interval $[0, 1]$, so the adjustments to the values of π_{ts} have much wider range and by changing the order of min and max we do not allow for any of those changes. It still holds that the approximation is better for smaller values of ϵ.

We do not state again the program, it is enough to adjust domain for π_{ts} in (2.39).

For the case of general probabilities, one would use a more general formulation (2.16). In this case, $\sum_{t=1}^{T} \pi_{ts} = 1, s = 1, ..., T$ is just replaced by the constraint $\sum_{t=1}^{T} p_t \pi_{ts} = p_s, s = 1, ..., T$. Thanks to this nice formulation, it is much easier than it was for the case of the FSD.

Robustness in probabilities

As it was for the case of the robust FSD, the ordering does not change when the returns do not change, which allows us to use the formulation (1.11). We use again the approximation to remove the dependence on π_{ts}. It now appears in two inequalities but it does not matter. We want that for each set of probabilities p_t there exist π_{ts} such that the inequalities hold. If we find π_{ts} such that it holds for every possible choice of p_t then the constraints are satisfied. We can select this π_{ts} for every choice of p_t. So we again use a conservative approach. After the approximation, we deal with the problem:

$$\begin{aligned}
&\max_{w_i, \pi_{ts}} f(\boldsymbol{w}) \\
&\text{subject to } \max_{p_t} \sum_{t=1}^{T} p_t \sum_{s=1}^{k-1} (y_k - y_s)\pi_{ts} - \sum_{s=1}^{k-1} (y_k - y_s) p_s \leq 0, \quad && k = 2, ..., T \\
&\sum_{t=1}^{T} \left| p_t - p_t^0 \right| \leq \epsilon \\
&\sum_{t=1}^{T} p_t = 1 \\
&\sum_{s=1}^{T} \pi_{ts} = 1, && t = 1, ..., T \\
&\sum_{i=1}^{N} w_i r_{it} \geq \sum_{s=1}^{T} y_s \pi_{ts}, && t = 1, ..., T \\
&\pi_{ts} \geq 0, && t, s = 1, ..., T \\
&p_t \geq m, && t = 1, ..., T \\
&\boldsymbol{w} \in W.
\end{aligned} \tag{2.54}$$

We restrain the sum of changes by ϵ. Again we must have all probabilities positive so the conditions make sense. We use m (margin) as a small positive constant. We must also make sure that p_t are probabilities so that they sum to 1. Again we understand the constraint with maximum that it must hold for all $k, k = 2, ..., T$.

Let us have a look at the inner optimization program for a fixed k:

$$
\begin{aligned}
&\max_{p_t} \sum_{t=1}^{T} p_t \sum_{s=1}^{k-1} (y_k - y_s)\pi_{ts} - \sum_{s=1}^{k-1} (y_k - y_s) p_s \\
\text{subject to } &\sum_{t=1}^{T} \left| p_t - p_t^0 \right| \leq \epsilon \\
&\sum_{t=1}^{T} p_t = 1 \\
&p_t \geq m, \qquad\qquad t = 1, ..., T.
\end{aligned}
\tag{2.55}
$$

We again shift the variables to the originally observed point, $q_t = p_t - p_t^0$. As the returns of benchmark y_t are constants, let us denote $(y_k - y_s) = g_{ks}$. We rewrite the problem as:

$$
\begin{aligned}
&\max_{p_t} \sum_{t=1}^{T} \left(p_t - p_t^0\right) \sum_{s=1}^{k-1} g_{ks}\pi_{ts} + \sum_{t=1}^{T} p_t^0 \sum_{s=1}^{k-1} g_{ks}\pi_{ts} \\
&\quad - \sum_{s=1}^{k-1} g_{ks} \left(p_s - p_s^0\right) - \sum_{s=1}^{k-1} g_{ks} p_s^0 \\
\text{subject to } &\sum_{t=1}^{T} \left| p_t - p_t^0 \right| \leq \epsilon \\
&\sum_{t=1}^{T} \left(p_t - p_t^0\right) = 0 \\
&p_t - p_t^0 \geq m - p_t^0, \quad t = 1, ..., T.
\end{aligned}
\tag{2.56}
$$

Now we plug in the q_t and remove the parts which do not depend on it, which will be added back later.

$$
\begin{aligned}
&\max_{q_t} \sum_{t=1}^{T} q_t \sum_{s=1}^{k-1} g_{ks}\pi_{ts} - \sum_{s=1}^{k-1} g_{ks} q_s \\
\text{subject to } &\sum_{t=1}^{T} |q_t| \leq \epsilon \\
&\sum_{t=1}^{T} q_t = 0 \\
&q_t \geq m - p_t^0 \qquad\qquad t = 1, ..., T.
\end{aligned}
\tag{2.57}
$$

For the same reasons as above we can use positive and negative parts q_t^+, q_t^-

to get a linear programming formulation.

$$
\begin{aligned}
\max_{q_t^+, q_t^-} \quad & \sum_{t=1}^{T} (q_t^+ - q_t^-) \sum_{s=1}^{k-1} g_{ks} \pi_{ts} - \sum_{s=1}^{k-1} g_{ks} (q_s^+ - q_s^-) \\
\text{subject to} \quad & \sum_{t=1}^{T} (q_t^+ + q_t^-) \leq \epsilon \\
& \sum_{t=1}^{T} (q_t^+ - q_t^-) = 0 \\
& (q_t^+ - q_t^-) \geq m - p_t^0, \qquad\qquad t = 1, ..., T \\
& q_t^+, q_t^- \geq 0, \qquad\qquad t = 1, ..., T.
\end{aligned}
\tag{2.58}
$$

The dual formulation is

$$
\begin{aligned}
\min_{a_k, b_k, c_{tk}} \quad & a_k \epsilon + \sum_{t=1}^{T} \left(p_t^0 - m \right) c_{tk} \\
\text{subject to} \quad & a_k + b_k - c_{tk} \geq \sum_{s=1}^{k-1} g_{ks} \pi_{ts} - g_{kt}, \qquad 1 \leq t \leq k-1 \\
& a_k - b_k + c_{tk} \geq \sum_{s=1}^{k-1} -g_{ks} \pi_{ts} + g_{kt}, \qquad 1 \leq t \leq k-1 \\
& a_k + b_k - c_{tk} \geq \sum_{s=1}^{k-1} g_{ks} \pi_{ts}, \qquad T \geq t > k-1 \\
& a_k - b_k + c_{tk} \geq \sum_{s=1}^{k-1} -g_{ks} \pi_{ts}, \qquad T \geq t > k-1 \\
& c_{tk} \geq 0, \qquad t = 1, ..., T \\
& a_k \geq 0 \\
& b_k \in \mathbb{R}.
\end{aligned}
\tag{2.59}
$$

Now we plug in the dual formulation for each k as we did the the case of returns and get the final formulation (we also remove the minimum and add the terms we skipped when deriving the dual formulation):

$$
\begin{aligned}
&\max_{w_i,\pi_{ts},a_k,b_k,c_{tk}} f(\boldsymbol{w}) \\
s.t.\ & a_k\epsilon + \sum_{t=1}^{T}\left(p_t^0 - m\right)c_{tk} \le -\sum_{t=1}^{T} p_t^0 \sum_{s=1}^{k-1} g_{ks}\pi_{ts} + \sum_{s=1}^{k-1} g_{ks}p_s^0,\ k = 2, ..., T \\
& a_k + b_k - c_{tk} \ge \sum_{s=1}^{k-1} g_{ks}\pi_{ts} - g_{kt}, \quad 1 \le t \le k-1; k = 2, ..., T \\
& a_k - b_k + c_{tk} \ge \sum_{s=1}^{k-1} -g_{ks}\pi_{ts} + g_{kt}, \quad 1 \le t \le k-1; k = 2, ..., T \\
& a_k + b_k - c_{tk} \ge \sum_{s=1}^{k-1} g_{ks}\pi_{ts}, \quad T \ge t > k-1; k = 2, ..., T \\
& a_k - b_k + c_{tk} \ge \sum_{s=1}^{k-1} -g_{ks}\pi_{ts}, \quad T \ge t > k-1; k = 2, ..., T \\
& \sum_{i=1}^{N} w_i r_{it} \ge \sum_{s=1}^{T} y_s\pi_{ts}, \quad t = 1, ..., T \\
& \sum_{s=1}^{T} \pi_{ts} = 1, \quad t = 1, ..., T \\
& \pi_{ts} \ge 0, \quad t = 1, ..., T; s = 1, ..., T \\
& c_{tk} \ge 0, \quad k = 2, ..., T; t = 1, ..., T \\
& a_k \ge 0, \quad k = 2, ..., T \\
& b_k \in \mathbb{R}, \quad k = 2, ..., T \\
& \boldsymbol{w} \in W.
\end{aligned}
\tag{2.60}
$$

For the robustness in the objective function (robust expected return), we can use derivation we already did for the FSD, there are no changes.

For higher order stochastic dominance, we do not have a nice formulation to use, we would have to use the twice integrated distribution functions, which would be very hard to solve. We could use the approximation using SCTSD, but the formulation is already a quadratically-constrained program. It also turns out that when we use SCTSD, the portfolio might have lower expected return than when we use the SSD, so in this case we take the portfolio from the SSD.

3. Application on financial data

3.1 Data

We used stock prices of assets covered by the Dow Jones Industrial Average (DJIA) index, which consists of 30 American companies, which belong to the largest and most traded ones. We obtained the dataset from Kaggle (kag [2018]), which included data of more than 2000 stocks and the ones covered by DJIA were among them. R software (The R Project for Statistical Computing) was used to prepare the dataset for further use and also for the visualization of the results. Adjusted close prices were transformed them to simple returns, which were then used in GAMS ([GAMS Development Corporation, 2016]). The dataset used consists of observations from April 2008 to July 2018. We selected the start date this way, because this is the month when Visa entered the stock market and we do not have to exclude any stocks or deal with the problem of missing data. Two main datasets were created, one with monthly returns (124 observations of returns) and one with quarterly returns (41 observations of returns). For purposes where we needed even smaller dataset, because the complexity of the problem was very high, we used a subset of the quarterly returns. For practical use, when one wants to cover larger periods with smaller number of observations, longer term such as half-year or even year can be chosen for return observations.

In table 3.1 we can find all the tickers (of the considered stocks) with some basic characteristics and information. We can find there the full name of the company, mean (expected return), minimal and maximal return, standard deviation and median, all computed from the quarterly returns. We can see that Apple has the highest expected return and also has higher SD than most of the other assets, but there are some stocks which are dominated by Apple in Markowitz sense (lower return and higher variance). We can also see that Home Depot dominates many other stocks in the Markowitz sense.

To state the computation capabilities, we used a PC with 16 GB RAM and Intel core i5 6500 Skylake, we state the computation times for better orientation in the complexity of the problems. Non-commented scripts in GAMS and R used for the computation are in the attachments.

3.2 Assumptions and benchmark portfolio

In all the applications, we need a benchmark portfolio with fixed weights, so we cannot use the index, because its weights change with the prices of stocks. For this reasons we use $1/N$ strategy, which means that we take as the benchmark a portfolio that invests the same amount into every available asset. For our case of the Dow Jones 30 it means that $1/30$ is invested in every stock (in our notation $\tau_i = 1/30, i = 1, ..., 30$).

For the form of the set W of possible weights, for simplicity we use the constraints that $w_i \geq 0, i = 1, ..., N; \sum_{i=1}^{N} w_i = 1$, i.e. no short selling. To complete our set of assumptions, we do not consider any transactions costs and we assume that the stocks are perfectly divisible.

Table 3.1: Basic characteristics calculated from quarterly data

Ticker	Name	Mean	Min	Max	SD	Median
AAPL	Apple	0.08	-0.38	0.50	0.16	0.09
V	Visa	0.07	-0.24	0.32	0.12	0.06
UNH	UnitedHealth	0.06	-0.29	0.28	0.13	0.06
HD	Home Depot	0.06	-0.21	0.34	0.10	0.07
NKE	Nike	0.05	-0.18	0.23	0.12	0.08
BA	Boeing	0.05	-0.21	0.31	0.14	0.05
DWDP	DowDuPont	0.05	-0.50	0.83	0.23	0.05
MSFT	Microsoft	0.04	-0.23	0.25	0.11	0.04
DIS	The Walt Disney Company	0.04	-0.27	0.34	0.13	0.07
JPM	JPMorgan Chase & Co.	0.04	-0.36	0.46	0.16	0.05
AXP	American Express	0.04	-0.45	0.60	0.17	0.04
CAT	Caterpillar	0.04	-0.38	0.51	0.18	0.06
MCD	McDonald's	0.04	-0.12	0.20	0.07	0.02
MMM	3M	0.03	-0.26	0.21	0.10	0.04
INTC	Intel	0.03	-0.17	0.21	0.11	0.02
TRV	The Travelers Companies	0.03	-0.20	0.27	0.09	0.03
PFE	Pfizer	0.03	-0.22	0.28	0.10	0.04
WBA	Walgreens Boots Alliance	0.03	-0.29	0.31	0.14	0.02
JNJ	Johnson & Johnson	0.02	-0.12	0.17	0.07	0.03
MRK	Merck & Co.	0.02	-0.15	0.23	0.09	0.02
UTX	United Technologies Corp.	0.02	-0.22	0.20	0.10	0.05
VZ	Verizon Communications	0.02	-0.11	0.27	0.09	0.01
CVX	Chevron	0.02	-0.17	0.24	0.11	-0.00
CSCO	Cisco Systems	0.02	-0.23	0.23	0.10	0.03
GS	Goldman Sachs Group Inc	0.02	-0.35	0.50	0.17	0.01
KO	The Coca-Cola Company	0.02	-0.16	0.20	0.07	0.02
WMT	Walmart	0.02	-0.13	0.26	0.08	0.02
IBM	IBM	0.01	-0.20	0.13	0.08	0.02
PG	Procter & Gamble	0.01	-0.23	0.16	0.08	0.02
XOM	ExxonMobil	0.01	-0.16	0.22	0.09	0.01

As our objective function, we used the expected return if not said otherwise: $f(\boldsymbol{w}) = \sum_{t=1}^{T} p_t \sum_{i=1}^{N} w_i r_{it}$.

3.3 Portfolio with SD constraints

In this section, we will briefly comment on the results of using optimization programs for the first, the second and the third order stochastic dominance without robustness.

3.3.1 First order stochastic dominance

We used the Cplex solver for our MIP (mixed integer programming) problem using the formulation (1.7). For the monthly data, the computation did not finish in reasonable time because of the large amount of binary variables, which increase the complexity of the problem. For the quarterly data, the computation took about 1 second and the results can be found in table 3.2.

Table 3.2: Weights of FSD portfolio

ticker	AAPL	CAT	DIS	HD	JPM	MCD	TRV	UNH	V
weight	0.441	0.001	0.015	0.321	0.029	0.037	0.030	0.002	0.124

Note that due to rounding, weights might not sum precisely to 1. As we can see, the portfolio consists of stocks with the highest expected returns, which is not surprising as our objective is to maximize the expected return. Because we used the FSD, the portfolio could not consist only of Apple as the best asset in terms of expected return and there are also many other assets in the portfolio.

3.3.2 Second order stochastic dominance

We used Cplex solver for LP (linear programming), we were able to solve the problem for the monthly data as well, the computation took less than 1 second. We used the formulation (1.11). The problem is formulated as a linear programming program, so even much larger dataset could be used. The optimal portfolio weights for monthly data can be found in table 3.3 and for quarterly data in table 3.4.

Table 3.3: Weights of SSD portfolio for monthly data

ticker	AAPL	NKE	V
weight	0.310	0.339	0.351

Table 3.4: Weights of SSD portfolio for quarterly data

ticker	AAPL	HD
weight	0.528	0.472

First thing to notice is that for quarterly data, the portfolio is more greedy, i.e. the SSD constraint is not so strict as in the case of monthly data. This corresponds to the fact that if we had only one observation, we would invest all in the asset with highest return, when we use more observations within the same period, the SD constraint becomes more restrictive. When we compare the results with the FSD using the quarterly data, we can see that the optimal portfolio consists of less assets and the weights are higher for the most profitable stocks. This corresponds to the fact that the FSD implies the SSD so the optimal portfolio with the SSD constraint has at least the same expected return as the optimal portfolio with the FSD constraint (we used expected returns as the objective functions).

To compare the expected returns, we transformed the SSD portfolio calculated using monthly data to quarterly returns. The FSD for quarterly data has quarterly expected return 0.065, the SSD for monthly data has quarterly expected return also 0.065 and the SSD for quarterly data has quarterly expected return 0.068.

3.3.3 Third order stochastic dominance

We used Cplex solver for QCP (quadratically-constrained programming), we were able to get solution for the monthly dataset, the computation took about 15 seconds. We used the formulation (1.14), where the concept of SCTSD is used. The weights for monthly data can be found in table 3.5 and for the quarterly data in table 3.6.

Table 3.5: Weights of SCTSD portfolio for monthly data

ticker	AAPL	HD	NKE	V
weight	0.252	0.048	0.336	0.364

Table 3.6: Weights of SCTSD portfolio for quarterly data

ticker	AAPL	HD
weight	0.510	0.490

Notice that the portfolio does look similar to the one we achieved with the SSD constraint, but we would expect just an increase of weights for the most profitable stocks, which did not happen. In fact, the expected return is little bit smaller for quarterly data (0.0679 for the SSD and 0.0676 for the SCTSD). This is caused by the fact that the SSD does not imply the SCTSD, so even though the portfolio we found as optimal for the SSD would be feasible for the TSD, it is not for the SCTSD. In this case, trying to use less strict constraint did not bring higher expected return because of the approximation by the SCTSD, but the difference is small from the SSD. Similar phenomenon can be observed for the quarterly data. When this happens we usually take the portfolio received from the SSD.

3.4 Worst case distribution

In this section, we analyzed the worst case distribution, which is obtained by (2.14), (2.19) and (2.24) for the optimal portfolios found in the previous section and the benchmark portfolio. We analyzed the development of the worst case distribution in dependence on the value of ϵ, which defines the radius of the neighborhood of the empirical distribution. Investor with such portfolio wants to know which is the worst possible distribution for him, what events would make him re-balance etc.

3.4.1 Robust SSD

We applied the robust SSD test ((2.14)) to the portfolio we achieved using the non-robust algorithm for the SSD. We used the CONOPT 3 solver for NLP (non-linear programming), the computation took less than 20 seconds for all values of ϵ (14 values which are defined later) for quarterly data, but the computation showed to be very unstable as we will see on the resulting worst distributions. The problem is non-linear and non-convex, which greatly increases the instability.

The solver was not even able to find a starting point, so a subprogram, which fixed observed values $r_{it} = r_{it}^0$ and $p_t = {}^1/_T$, was used to get a feasible starting point. Other solvers available for us were not able to solve the problem in reasonable time or converged to a point of infeasibility. Also as was commented in the previous chapter, the simplest form of distance was used in order to make the computation stable - the Euclidean norm without the square root. We also tried the absolute value, but the nonexistence of derivation disrupted the computation.

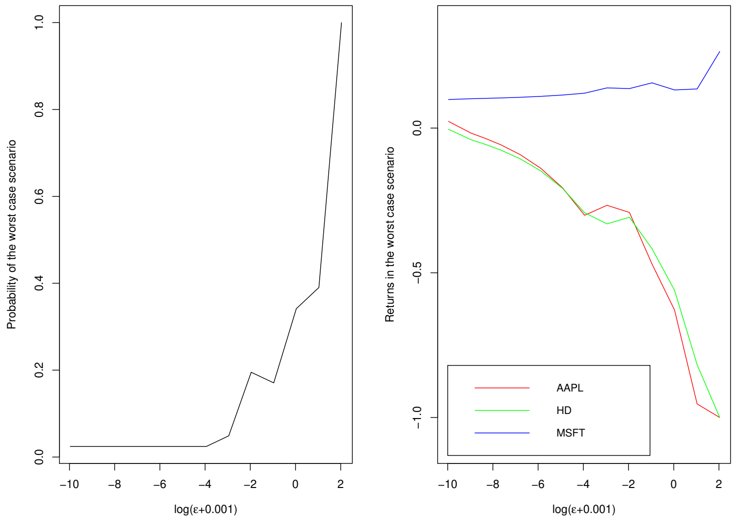
To comment further on the instability, the value of objective function (of (2.14)) in optimum for $\epsilon = 0$ should also be exactly 0 (as we selected the portfolio to be the optimal portfolio with the SSD constraint), which was not true and was around 0.013, which shows great numerical difficulties. For the starting point ($\epsilon = 0$), IPOPT solver was also able to find a solution and gave the correct solution of 0 ($\epsilon = 0$ is equivalent to $r_{it} = r_{it}^0$ and $p_t = {}^1/_T$), but did not converge for the actual program. Even though there are numerical challenges and instability, the results given by the solver do make good sense and provide us valuable information, but we have to keep in mind that the found solution is probably only a local optimum and that there might be some inaccuracies in the results.

We chose the values of ϵ as 0 for the first case and then starting with 0.001, we doubled the value of epsilon, ending with 4.096, when the probability of one of the scenarios was 1 (the worst case scenario).

For quarterly data, the portfolio consists only of two assets, AAPL and HD. The behavior of the worst case distribution is almost as expected, but the solver had problem with converging to an infeasible solution for smaller values of ϵ. The constraint for the distance of resulting and empirical distribution is always fulfilled as an equality, which means the program used the possibility to use the furthest distribution possible, which was expected. The probability of a specific scenario (usually the scenario with the worst return in the empirical distribution) is inflated and the returns in this scenario are pushed down for assets in the portfolio (AAPL, HD). The returns of other assets, which are not in the portfolio, but are in the benchmark, are pushed up much less, because the exposition in the benchmark portfolio for every asset is relatively smaller in comparison to the exposition of the two assets in our portfolio. It is also hard to see precisely how the Wasserstein distance is formed, given that it is the result of an optimization program, which affects our expectations of the development of the worst case distribution. Some of the basic development of the probability of the worst case scenario and returns in this scenario for some of the stocks is captured in fig. 3.1. We can see that the development is not really smooth, it jumps up and down. This is most likely the result of the numerical instability. For the smaller values of ϵ, there are also some other scenarios for which the probability is higher than ${}^1/_T$ and for which the return of stocks included in our portfolio is decreased and the return of other stocks is increased, however only by a smaller amount.

We were also able to achieve results for monthly data, the computation took less than 10 minutes for all ϵ considered. The behavior of explored variables seemed to be more stable than it was for the quarterly data, but we can see that the solver most likely found only local optimum from the way the values rise and fall with increasing ϵ (but the bumps are smaller than for the quarterly data). We can see the development of probability and returns of some stocks for the worst case scenario in fig. 3.2 and in fig. 3.3 for another scenario (which has higher

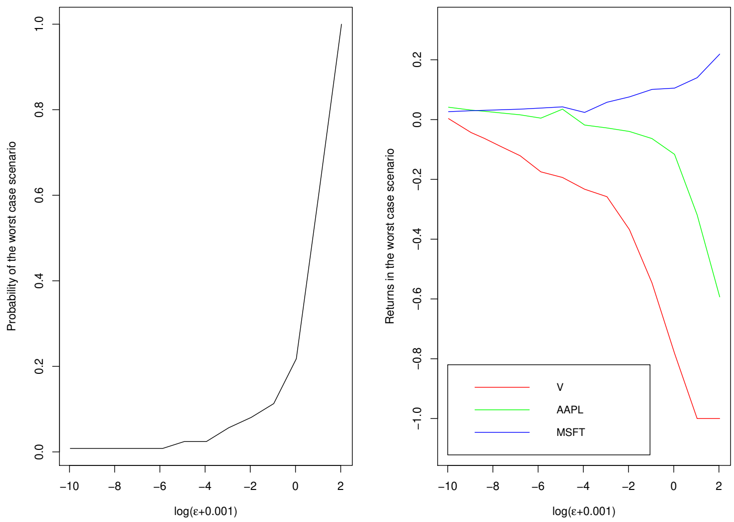
Figure 3.1: Development of probability and returns of the worst case scenario with increasing ϵ for quarterly data for the SSD



probability than $1/T$ until the probability of the worst case scenario is 1). We can see that increase in probability is connected with decrease in returns of stocks in our portfolio.

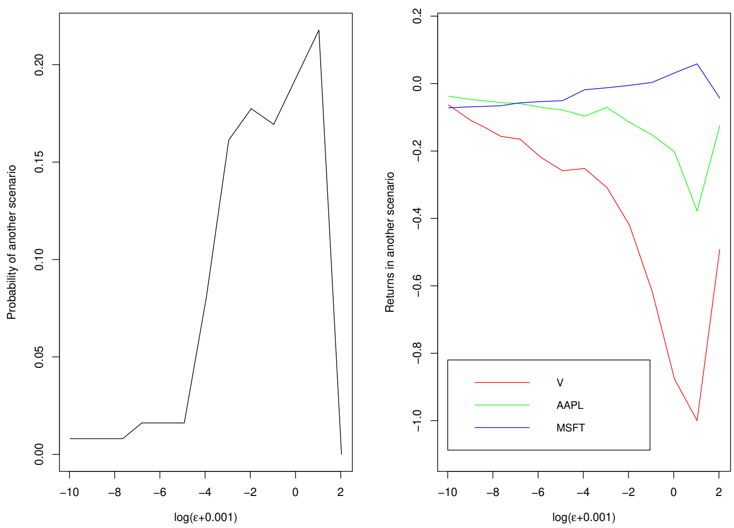
The optimal solution is also "path-dependent". This means a different scenario is chosen based on the initial values saved in the variables than it would have been chosen if there were no initial values of the variables. This shows that the optimal solution is only local.

Figure 3.2: Development of probability and returns of the worst case scenario with increasing ϵ for monthly data for the SSD



To analyze the worst case distribution even further, we plot cumulative distribution functions for both our and benchmark portfolios for empirical distribution and for the worst case distribution for multiple values of ϵ for quarterly data. We

Figure 3.3: Development of probability and returns of another scenario with increasing ϵ for monthly data for the SSD



can see in fig. 3.4 how the left tail (lowest returns) is amplified for our portfolio. We can see how the worst case distribution evolves (expressed as distribution function of returns of our and benchmark portfolios) with increasing ϵ. It corresponds to the fact that the formulation from which the test was derived compares in fact the integrated distribution functions, which means the areas under the cumulative distribution function. This can be clearly seen in the picture, the worst case distribution maximizes the difference in the areas for some specific $x \in \mathbb{R}$. The x is the one where the distribution functions intersect in the pictures (they might not truly intersect as they are not continuous). We can see that returns for our portfolio are lowered and also that the probabilities of these scenarios with low returns for our portfolio are increased. This is particularly visible for the highest visualized ϵ. We can also see that the returns higher than the specific x are almost not changed, this is clear because the values that are higher do not contribute towards the objective value. Changing the returns for scenarios with higher return that the specific x would only increase the distance of distributions but would not increase the objective value.

We also analyzed the integrated distribution functions themselves. We can see the differences in integrated distribution functions in fig. 3.5. We can see that the first difference ($\epsilon = 0$) is below zero and this corresponds to the fact that the portfolio was constructed to dominate benchmark in the SSD. We can also see how the differences change with increasing ϵ. The x for which the maximum was attained for $\epsilon = 0.001$ was marked by the black dashed vertical line. We can see that the point almost does not change and that there was already a local maximum for the difference using the empirical distribution of returns ($\epsilon = 0$).

Figure 3.4: Cumulative distribution functions for empirical distribution and for the worst case distribution for different values of ϵ for quarterly data for the SSD

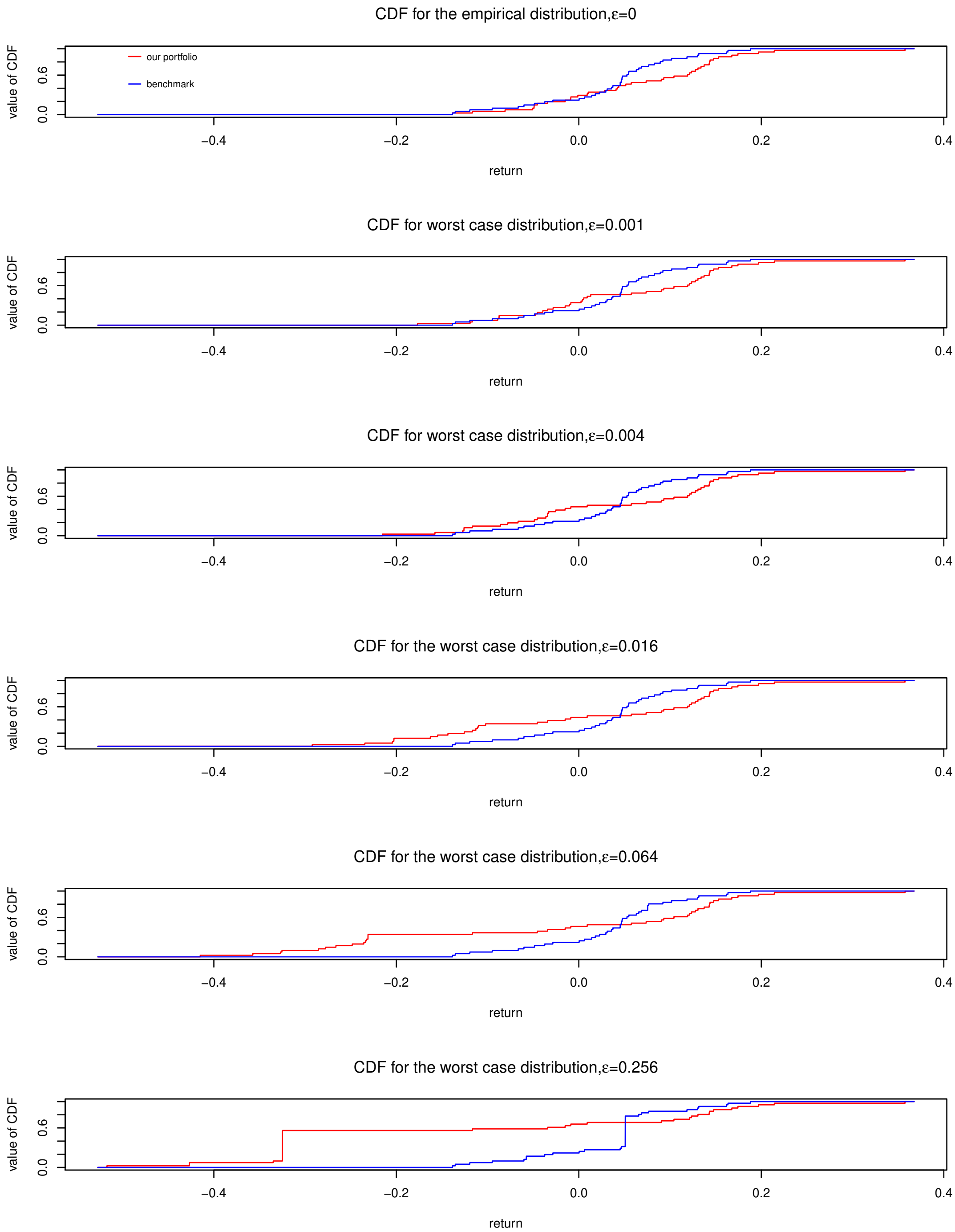
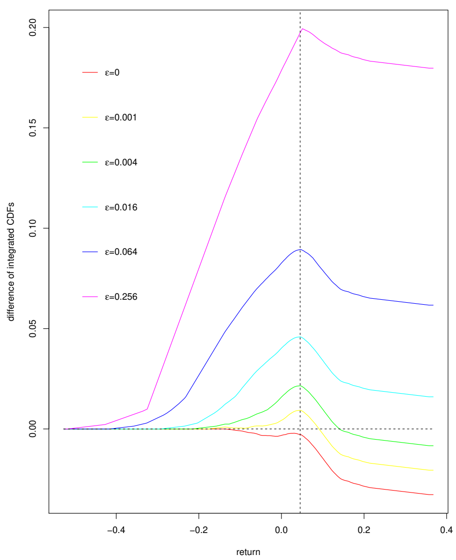


Figure 3.5: Differences of integrated distribution functions with increasing ϵ for quarterly data for the SSD

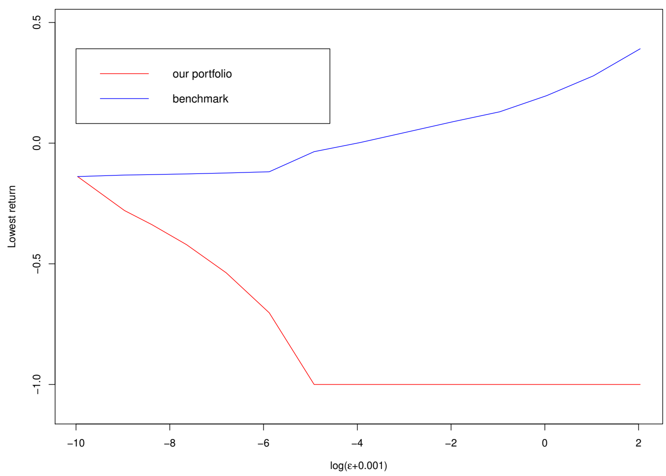


3.4.2 Robust SSD - alternative formulation

Under the same assumptions as above, we use formulation (2.19) to determine the worst case distribution and compare it with the one we obtained by the first approach. We used the same values of ϵ and the same portfolios.

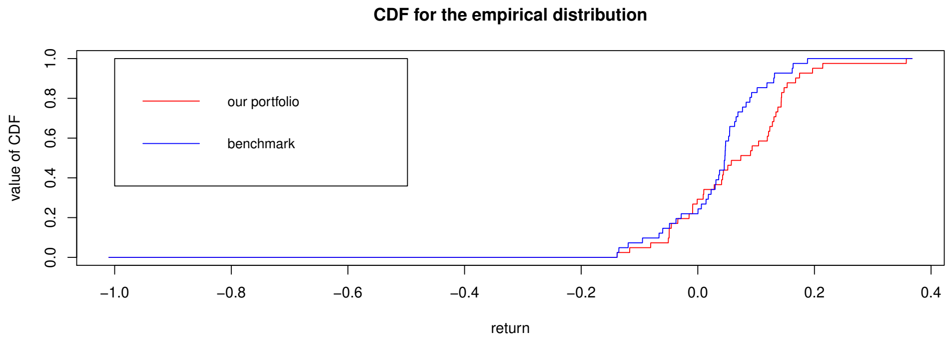
We used CONOPT 3 solver for NLP on both monthly and quarterly data. Computation took 35 minutes for the monthly data for all considered values of ϵ. The constraint for the distance of distributions was always fulfilled as an equality. The optimal solution with increasing ϵ changed in the way that it has chosen one scenario (the worst one), for which the returns were decreased for our portfolio (which means decreased return of assets in our portfolio) and it increased returns of other assets in this scenario. It also increased returns of all assets in all other scenarios (benchmark contains all assets). This follows from the conditions in formulation (1.5), the condition for the lowest return is the same as it is for the FSD, because the sum has only one element. The worst case distribution uses this fact and chooses one scenario for which the return is low and then tries to make the minimum return of benchmark over the scenarios to be as high as possible. This comes from the formulation where the differences in the corresponding returns are considered, not integrated distribution functions as it was in the first approach. This approach in some sense transforms the problem to FSD, as was explained. The resulting distribution is not intuitive as it increases the return of highly exposed assets that are in our portfolio in all but one scenario. Basic illustration of this behavior can be found in fig. 3.6 for quarterly data and was almost the same for the monthly data.

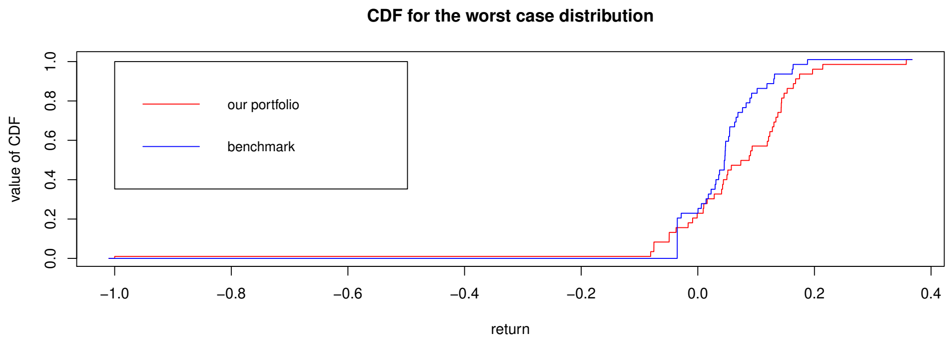
Figure 3.6: Development of lowest returns of our and benchmark portfolio in the worst case distribution with increasing ϵ for the SSD alternative approach for quarterly data



We use $\epsilon = 0.032$ for quarterly data to plot the CDFs. We can see in fig. 3.7 that there is one scenario (the probability was originally 10^{-5} (margin m) but we changed it so it is visible in the plot) for which the return is pushed to -1 for our portfolio. Also there is a big jump for the benchmark, because many scenarios have the same return, which corresponds to the previous description to maximize the difference between the lowest returns. This approach does not compare integrated distribution functions so inspecting them would not provide any additional information.

Figure 3.7: Cumulative distribution functions for empirical distribution and for the worst case distribution for $\epsilon = 0.032$ for quarterly data for the SSD alternative approach





To compare with the first approach, the resulting portfolio is not intuitive. The expected value in the first approach gives it nice interpretation, this test just checks the definition and the constraint is strictest in the lowest return, which is used by the worst distribution. On the other hand the computation was more stable.

3.4.3 Robust FSD

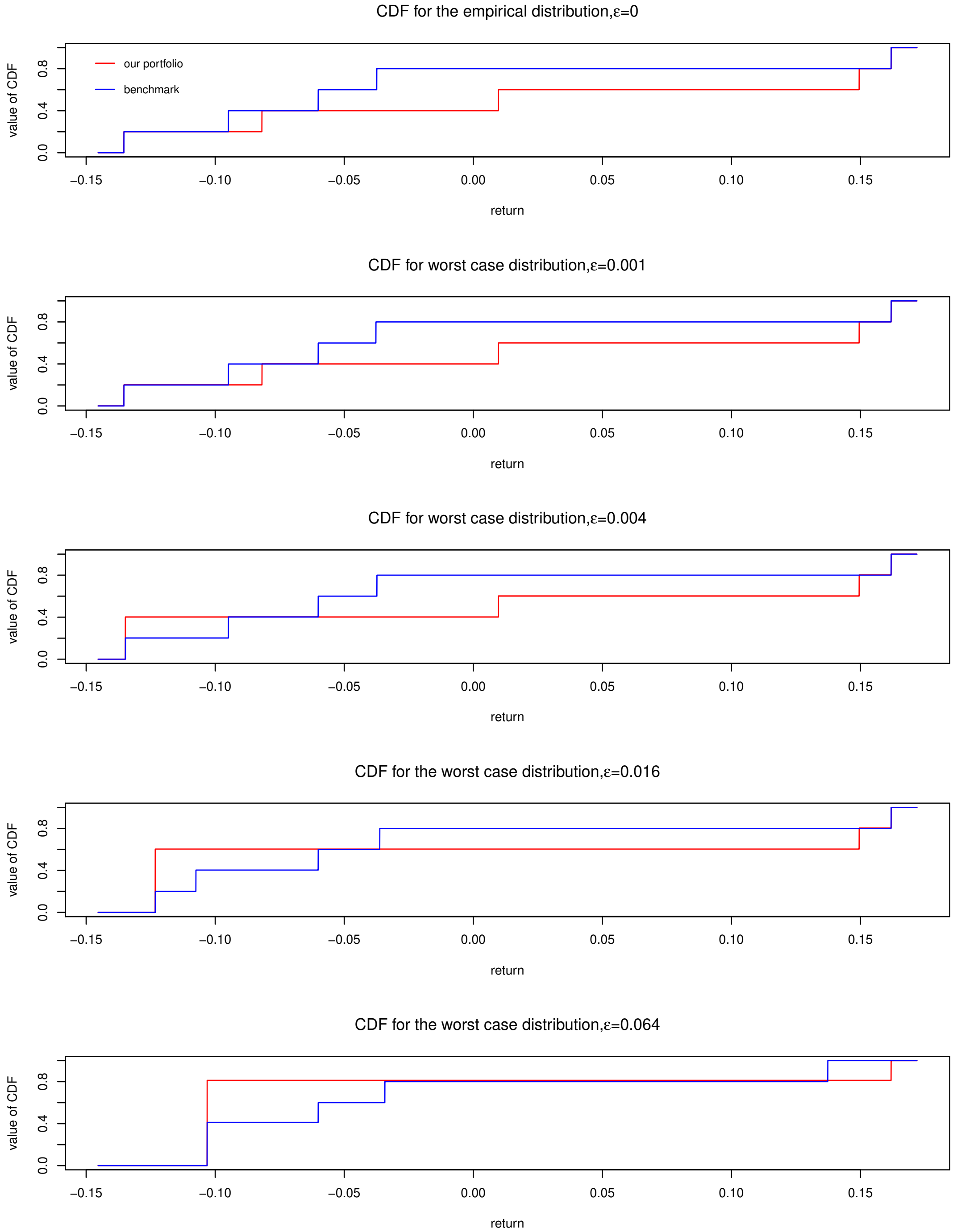
The robust FSD test given by (2.24) faced even greater challenges given that the problem is not only non-linear, but also mixed integer. We used BONMIN solver for MINLP (mixed integer non-linear programming). We used a sub-sample of the quarterly data, precisely the first 5 observations (more caused computational problems, solver converging to an infeasible solution). We considered the same values of ϵ ending with 0.128 when the difference of the distribution functions was maximal (value 1). The computation took around 2 minutes for all the considered values of ϵ. We used again as our portfolio the portfolio obtained by optimizing expected return with FSD constraints, the portfolio contained V (0.354), MCD (0.391), AAPL (0.255), where the numbers in brackets represent weights.

The constraint for the distance of distributions was fulfilled as equality as is was for the second order case, the value was slightly lower on the left side of the equality for the last value of ϵ ($0.127 < 0.128$). This does makes sense, there was no option for the objective to further improve as it was already 1. The probabilities were changed in the worst case scenario for the lower values of ϵ, because the change of probability could make the difference between cumulative distribution functions larger. As the program was able to make the difference close to 1, there was no need to adjust the probabilities as all the values of benchmark had to be larger than the values of our portfolio. We cannot really see much from the development of a single scenario as it was for the case of the second order stochastic dominance, so we use the cumulative distribution functions for visualization.

We can see multiple cumulative distribution functions plotted in fig. 3.8. We can see the rise of the CDF for our portfolio in one part of the graphs, because the FSD condition is strictly the difference between CDFs, it might not be on the left tail as it was for the SSD. The fact that the lowest return is actually where the difference is the highest is connected to the fact that the portfolio was created such that the conditions are fulfilled as an equality on the left tail (lower returns) so it is easier (in terms of distance of distributions) to violate those conditions.

Even though it is not visible from the graph, the FSD condition is violated for $\epsilon = 0.001$, the difference in the returns is so small it cannot be captured by the graph, the critical points are return -0.13538 for our portfolio and -0.13536 for the benchmark. This is the m (margin) we used for the computation, this difference can be made arbitrarily small and the FSD conditions would be violated. The situation is the same for other depicted ϵ, for example for $\epsilon = 0.064$ the returns are -0.10312 for our portfolio and -0.10311 for the benchmark.

Figure 3.8: Cumulative distribution functions for empirical distribution and for the worst case distribution for different values of ϵ for FSD



3.5 Portfolios with robust stochastic dominance

We present portfolio weights for different values of ϵ. For higher values of ϵ the weights are the same as for the last stated value.

3.5.1 Robustness in returns for the FSD

We inspected the optimal portfolios given by solving the problem (2.39) for different values of ϵ. We started again with $\epsilon = 0$ and we expected that the solver would give us the same solution as if we would have used the non-robust program. This was true for some of the solvers, some gave slightly different solutions. We used the one that gave the same solution, BONMIN. For example DICOPT gave slightly different results. We used the first 10 observations from quarterly data, for 20 the computation did not finish in reasonable time for some values of ϵ (the ones in the middle, the computation was fast for very high or low values). For 10 observations, the computation took less than a minute for all of the values of ϵ.

We can see the results in table 3.7. The values start on the values that gave the non-robust program and as the ϵ increases, the weights converge to the weights of the benchmark portfolio, which is an expected outcome. When the returns can change too much, the benchmark is the only portfolio that can dominate the benchmark (whatever may happen, they are identical). There is no visible change for the smallest value of ϵ, but it is only due to rounding, there are some smaller changes in the weights.

The program without robustness in the objective function gave slightly different results (for values of ϵ different from 0 and 0.512) as it was taking into consideration the original values, not maximizing for the worst case.

3.5.2 Robustness in probabilities for the FSD

We test the program (2.51) (with observed probabilities $1/T$) on real data to inspect its behavior. We again used the same dataset as for the robustness in returns and the same solver, the results are in table 3.8. The values are again the same for $\epsilon = 0$, because this is the empirical distribution. The weights do not change when increasing ϵ even more, the final portfolio is not the benchmark portfolio now, which could be surprising. In fact, the portfolio does have better return in each of the considered scenarios, so the probabilities does not matter anymore. This is also caused by the fact that we have only 10 scenarios, for higher number of scenarios, it is harder to find a portfolio that does have better return in each of them. For higher number of scenarios, the weights should be drawn closer to the benchmark portfolio.

For the program with non-robust objective function, we received slightly different results but it is the same case as it was for the robustness in returns. We consider the original observations or we want to prepare for the worst.

3.5.3 Robustness in returns for the SSD

We use the formulation (2.39) with changed domain for the π_{ts} to non-negative from binary, the results can be found in table 3.9. We used CONOPT solver for

Table 3.7: Weights of each stock in the optimal portfolio for different values of ϵ for FSD robust in returns

$\epsilon \cdot 1000$	0	1	2	4	8	16	32	64	128	256	512
MMM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.18	0.09	0.03
AXP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.03
AAPL	0.52	0.52	0.52	0.52	0.51	0.49	0.44	0.36	0.24	0.09	0.03
BA	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
CAT	0.00	0.00	0.00	0.00	0.00	0.02	0.08	0.14	0.10	0.03	0.03
CVX	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
CSCO	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
KO	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
DIS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
DWDP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.08	0.03
XOM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
GS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
HD	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.09	0.03
IBM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.09	0.03
INTC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
JNJ	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
JPM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.03
MCD	0.29	0.29	0.29	0.30	0.31	0.33	0.36	0.36	0.24	0.09	0.03
MRK	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
MSFT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
NKE	0.19	0.19	0.19	0.18	0.18	0.16	0.12	0.09	0.11	0.09	0.03
PFE	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
PG	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
TRV	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.08	0.03
UTX	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
UNH	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.04	0.08	0.03
VZ	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.03
V	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.09	0.03
WMT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.03
WBA	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03

NLP and the computation took less than a minute for all considered values of ϵ. We used again the same dataset as for the FSD so we can compare the results.

The portfolio is actually identical to the FSD for $\epsilon = 0$ (which was not the case when we used all the scenarios as we saw in the first part of the empirical study), so we have the same starting point. Compared to the FSD, we can see that the portfolios do have higher weights in the more profitable portfolios as we expected because we maximize (robust) expected return and the constraints are not so strict as they are for the FSD. The portfolios again converge to the benchmark portfolio as it is the only feasible portfolio when we allow huge changes in returns.

Table 3.8: Weights of each stock in the optimal portfolio for different values of ϵ for FSD robust in probabilities

$\epsilon \cdot 1000$	0	1	2	4	8	16	32	64	128	256
MMM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
AXP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
AAPL	0.52	0.39	0.39	0.39	0.44	0.36	0.43	0.39	0.45	0.35
BA	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
CAT	0.00	0.07	0.06	0.08	0.00	0.00	0.06	0.07	0.00	0.00
CVX	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
CSCO	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
KO	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DIS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DWDP	0.00	0.05	0.05	0.05	0.00	0.14	0.00	0.05	0.00	0.06
XOM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
HD	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
IBM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
INTC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
JNJ	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
JPM	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.13	0.08
MCD	0.29	0.38	0.36	0.42	0.16	0.32	0.32	0.39	0.32	0.38
MRK	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
MSFT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
NKE	0.19	0.02	0.04	0.00	0.37	0.08	0.13	0.03	0.10	0.13
PFE	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PG	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
TRV	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
UTX	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
UNH	0.00	0.08	0.09	0.06	0.04	0.10	0.07	0.08	0.00	0.00
VZ	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
V	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
WMT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
WBA	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

3.5.4 Robustness in probabilities for the SSD

We use the formulation (2.60) with robustness in the objective function (with observed probabilities $1/T$), the results can be found in table 3.10. We used CONOPT solver for NLP and the computation took less than a minute for all considered values of ϵ. We used again the same dataset as for the FSD so we can compare the results.

The results are similar to the FSD, the portfolio again converges to one that has higher returns in every scenario. This time because the constraint is not so strict, the portfolio takes its final form (it does not change for higher values of ϵ) for $\epsilon = 0.512$ and it was only $\epsilon = 0.256$ for the FSD. We can again see that the weights are slightly higher than they were for the FSD for the more profitable stocks, which was expected.

Table 3.9: Weights of each stock in the optimal portfolio for different values of ϵ for SSD robust in returns

$\epsilon \cdot 1000$	0	1	2	4	8	16	32	64	128	256	512
MMM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.10	0.03
AXP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
AAPL	0.52	0.52	0.52	0.52	0.51	0.50	0.49	0.46	0.29	0.10	0.03
BA	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
CAT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.03
CVX	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
CSCO	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
KO	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.03
DIS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
DWDP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.03
XOM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
GS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
HD	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.10	0.03
IBM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.10	0.03
INTC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
JNJ	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
JPM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
MCD	0.29	0.29	0.29	0.30	0.31	0.33	0.37	0.45	0.29	0.10	0.03
MRK	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
MSFT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
NKE	0.19	0.19	0.19	0.18	0.18	0.16	0.14	0.10	0.23	0.10	0.03
PFE	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
PG	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
TRV	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.03
UTX	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
UNH	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.06	0.03
VZ	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.03
V	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.15	0.10	0.03
WMT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
WBA	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03

Table 3.10: Weights of each stock in the optimal portfolio for different values of ϵ for SSD robust in probabilities

$\epsilon \cdot 1000$	0	1	2	4	8	16	32	64	128	256	512
MMM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
AXP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
AAPL	0.52	0.45	0.45	0.44	0.43	0.43	0.49	0.47	0.35	0.35	0.35
BA	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
CAT	0.00	0.00	0.00	0.00	0.08	0.08	0.00	0.00	0.00	0.00	0.00
CVX	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
CSCO	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
KO	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DIS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DWDP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.01	0.06
XOM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
HD	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
IBM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
INTC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
JNJ	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
JPM	0.00	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08
MCD	0.29	0.31	0.31	0.16	0.42	0.42	0.29	0.18	0.24	0.25	0.38
MRK	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
MSFT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
NKE	0.19	0.21	0.21	0.37	0.07	0.07	0.22	0.27	0.33	0.33	0.13
PFE	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PG	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
TRV	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
UTX	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
UNH	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.08	0.04	0.05	0.00
VZ	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
V	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
WMT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
WBA	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

3.6 Out-of-sample performance

We wanted to test the performance of our portfolios calculated using the programs with robustness. The most straightforward application is that we build our portfolio based on the some observations that are available and then we analyze what happens in the observations to come. We want to know whether stochastic dominance still holds and we want to find a suitable value of ϵ. We used the first 10 observations (1-10) to determine the weights of the portfolios and now we use the next 10 observations (11-20) to assess the performance. We used an approach from the worst case distributions similar to (2.17) to measure if the portfolio dominates the benchmark.

$$\begin{aligned}
&\min_{z,\pi_{ts}} z \\
&\text{subject to } \sum_{s=1}^{T} y_s \pi_{ts} - x_t \leq z, && t = 1, ..., T \\
&\sum_{s=1}^{T} \pi_{ts} = 1, && t = 1, ..., T \\
&\sum_{t=1}^{T} \pi_{ts} = 1, && s = 1, ..., T \\
&\pi_{ts} \in \{0, 1\}, && t, s = 1, ..., T \\
&x_t = \sum_{i=1}^{N} w_i r_{it}, && t = 1, ..., T \\
&y_t = \sum_{i=1}^{N} \tau_i r_{it}, && t = 1, ..., T.
\end{aligned} \tag{3.1}$$

If z is non-positive, the portfolio does dominate the benchmark in the FSD as all the differences of corresponding returns are non-positive.

3.6.1 Robust FSD in returns

We now use the portfolios calculated using robustness in returns for the FSD. In table 3.11 we can see the values of z for portfolios created with different values of ϵ. We can see that the portfolio derived using the program without robustness does not dominate the benchmark as z is positive. We can also see that using higher value of ϵ when creating the portfolio does not provide us with FSD (apart from using the benchmark portfolio itself), but the distance z is much smaller which is an expected and wanted outcome when using robustness. We can also see the expected return in the table for each portfolio, there is one portfolio with high value of ϵ that had only slightly better expected return. The FSD constraint is very strict and even this approach was not enough to ensure it. There might be some ϵ between the last two values for which the portfolio would dominate benchmark but those are already portfolios very close to the benchmark.

3.6.2 Robust FSD in probabilities

We use the same approach but now we take the portfolio weights that we calculated using the program for robust probabilities. The results can be found in

Table 3.11: Fulfilling of the FSD for different portfolios calculated using robustness in returns

ϵ	0.000	0.001	0.004	0.016	0.032	0.064	0.128	0.256	0.512
z	0.063	0.063	0.062	0.059	0.055	0.061	0.043	0.011	0.000
ex. ret.	0.058	0.058	0.058	0.057	0.054	0.052	0.051	0.059	0.049

table 3.12. We can see that the FSD condition was not improved using these portfolios, we cannot see a significant local minimum for some values of ϵ. There is one value of ϵ for which the expected return is higher than for the portfolio without robustness. We should look for the optimal values of ϵ in the lower values, because the programs over-fits the data as it must have returns in every scenario for higher values of ϵ. One may prefer the portfolios from robustness in returns since they violate the FSD condition less for higher values of ϵ.

Table 3.12: Fulfilling of the FSD for different portfolios calculated using robustness in probabilities

ϵ	0.000	0.001	0.004	0.008	0.016	0.032	0.064	0.128	0.256
z	0.063	0.059	0.057	0.045	0.056	0.050	0.056	0.050	0.045
ex. ret.	0.058	0.053	0.052	0.060	0.054	0.056	0.053	0.055	0.053

3.6.3 Robust SSD in returns

We now use the same approach as for the FSD, but we use $\pi_{ts} \geq 0$ instead of $\pi_{ts} \in \{0, 1\}$. We can see the results in table 3.13. This time, choosing high enough values of ϵ helps us ensure the SSD, which shows the upside of using robust approach. We can again see that the portfolios with robustness also have higher expected return for those values.

Table 3.13: Fulfilling of the SSD for different portfolios calculated using robustness in returns

ϵ	0.000	0.001	0.004	0.016	0.032	0.064	0.128	0.256	0.512
z	0.017	0.017	0.017	0.015	0.014	0.011	-0.004	-0.008	0.000
ex. ret.	0.058	0.058	0.058	0.057	0.056	0.054	0.062	0.060	0.049

3.6.4 Robust SSD in probabilities

We again use the same method to test the portfolios we calculated using the programs with robust SSD in probabilities, the results can be found in table 3.14. We can see that the behavior is the same as it was for the FSD, the SSD constraint is not fulfilled for any values of ϵ and the value z does not really react to the value of ϵ. One may again prefer the portfolios calculated using using the robustness in returns for those reasons.

Table 3.14: Fulfilling of the SSD for different portfolios calculated using robustness in probabilities

ϵ	0.000	0.001	0.004	0.016	0.032	0.064	0.128	0.256	0.512
z	0.017	0.015	0.016	0.014	0.015	0.015	0.014	0.012	0.014
ex. ret.	0.058	0.057	0.060	0.053	0.058	0.060	0.057	0.058	0.053

Conclusion

In this work we introduced the concept of stochastic dominance and stated the programs for portfolio optimization with stochastic dominance of the first, the second and the third order in constraints. We introduced the robust optimization and the Wasserstein distance and proved the convexity of a set defined as a neighborhood of the empirical (or any observed) distribution with the Wasserstein distance.

We defined the worst case distribution for the second order stochastic dominance in two ways, first one which follows the equivalent condition with the expected value (based on integrated distribution function) and the other one following the equivalent condition by Luedtke [2008]. Based on the equivalent conditions, we derived tests of robust stochastic dominance for both approaches. We also defined the worst case distribution and derived the robust stochastic dominance test for the first order stochastic dominance.

In the next part, we derived programs to obtain robustly stochastic dominant portfolios when considering robustness in returns or in probabilities for both the first and the second order stochastic dominance, which to the best of our knowledge nobody was able to derive before. We used a conservative approach and picked a stronger constraint than the robust stochastic dominance to make the program computationally tractable.

In the empirical part, we tested all the derived programs on real life data, specifically on returns of assets captured by Dow Jones Industrial Average. The dataset was shortly introduced using some basic characteristics. We used the programs for the FSD, the SSD and the TSD without robustness to achieve our portfolios for further testing.

We analyzed the development of the worst case distribution with increasing value of the radius of the neighborhood around the empirical distribution. The results for all the tests were confronted with intuition and expectations and also compared between each other. The main features were captured graphically and by running the programs with multiple set ups, we were able to understand the behavior in detail. All of the tests showed some level of numerical instability because the programs are not linear and not even convex and as for the robust FSD test, the problem was also mixed integer. Even though the programs faced numerical challenges, we were able to get results, which made good sense, followed the definitions and used the strictest parts of the dominance conditions.

When testing the robust optimization problems with robust stochastic dominance, we used smaller dataset because of the numerical complexity of the problem. The programs were again non-linear and for the FSD also mixed-integer. We ran the program for multiple values of ϵ, which defines how much the returns/probabilities can differ from the observed values. The portfolio weights were presented in tables and the results correspond with our expectations. We also compared the results for all the approaches using the FSD/SSD and robustness in returns/probabilities.

The robustness in returns was successfully validated on out-of-sample observations and it was shown on real data that the robustness helps to reduce the violation of the FSD/SSD condition for higher values of epsilon. The expected returns were also better for the portfolios with higher values of ϵ than using for the portfolios derived without robustness, especially for the case of the SSD. The portfolios achieved by the robustness in probabilities turned out not to influence the fulfillment of the FSD/SSD constraint.

The subsequent research could focus on generalizing the programs for the FSD for probabilities different from $1/T$, where a permutation matrix might have to be introduced. One could also try to improve the approximation we used so that the set of feasible solutions is less affected or even try to solve the problem without any approximation. To further analyze the programs we derived, multiple datasets could be used and the results compared in a larger scale empirical study. When using robustness, the FSD and the SSD portfolios become more similar, this behavior could be analyzed more with the dependence on the value of ϵ. One could consider robust approaches for higher order stochastic dominance including TSD, which was not studied in detail in this work.

Bibliography

Daily historical stock prices (1970 - 2018), 2018. URL `https://www.kaggle.com/ehallmar/daily-historical-stock-prices-1970-2018/version/1`. Accessed 25.3.2019.

Darinka Dentcheva and Andrzej Ruszczynski. Convexification of stochastic ordering. *Comptes Rendus de l'Academie Bulgare des Sciences*, 57(4):4–11, 2004.

Darinka Dentcheva and Andrzej Ruszczyński. Portfolio optimization with stochastic dominance constraints. *Journal of Banking & Finance*, 30(2):433–451, 2006.

Darinka Dentcheva and Andrzej Ruszczyński. Robust stochastic dominance and its application to risk-averse optimization. *Mathematical Programming*, 123(1): 85, 2010.

GAMS Development Corporation. General Algebraic Modeling System (GAMS) Release 24.7.4. Washington, DC, USA, 2016. URL `http://www.gams.com`.

Giora Hanoch and Haim Levy. The efficiency analysis of choices involving risk. In *Stochastic Optimization Models in Finance*, pages 89–100. Elsevier, 1975.

Václav Kozmík. Eficience portfolií při spojitém rozdělení výnosů. Master's thesis, 2010.

Timo Kuosmanen. Efficient diversification according to stochastic dominance criteria. *Management Science*, 50(10):1390–1406, 2004.

Haim Levy. *Stochastic dominance: Investment decision making under uncertainty*. Springer, 2015.

James Luedtke. New formulations for optimization under stochastic dominance constraints. *SIAM Journal on Optimization*, 19(3):1433–1450, 2008.

Harry Markowitz. Portfolio selection. *The journal of finance*, 7(1):77–91, 1952.

Georg Ch Pflug and Alois Pichler. *Multistage stochastic optimization*. Springer, 2014.

Thierry Post and Miloš Kopa. Portfolio choice based on third-degree stochastic dominance. *Management Science*, 63(10):3381–3392, 2017.

R Tyrrell Rockafellar and Stanislav Uryasev. Conditional value-at-risk for general loss distributions. *Journal of banking & finance*, 26(7):1443–1471, 2002.

The R Project for Statistical Computing. R. URL `https://www.r-project.org/`.

List of Figures

List of Tables

List of Abbreviations

SD	Stochastic Dominance
FSD	First order Stochastic Dominance
SSD	Second order Stochastic Dominance
TSD	Third order Stochastic Dominance
SCTSD	Super-Convex Third order Stochastic Dominance
CDF	Cumulative Distribution Function
DJIA	Dow Jones Industrial Average
MIP	Mixed Integer Programming
LP	Linear Programming
QCP	Qquadratically-Constrained Programming
NLP	Non-Linear Programming
MINLP	Mixed Integer Non-Linear Programming