Using Column Generation to Solve Extensions to the Markowitz Model

Lorenz M. Roebers, Aras Selvi, and Juan C. Vera

Department of Econometrics and Operations Research, Tilburg University, 5037 AB Tilburg, The Netherlands

ARTICLE HISTORY
Compiled December 4, 2018

ABSTRACT
We introduce a solution scheme for portfolio optimization problems with cardinality constraints. Typical portfolio optimization problems are extensions of the classical Markowitz mean-variance portfolio optimization model. We solve such type of problems using a scheme similar to column generation. In this scheme, the original problem is restricted to a subset of the assets resulting in a master convex quadratic problem. Then the dual information of the master problem is used in a sub-problem to propose more assets to consider. We also consider other extensions to the Markowitz model to diversify the portfolio selection within the given intervals for active weights.

KEYWORDS
portfolio optimization; Markowitz portfolio theory; column generation

1. Introduction

In portfolio optimization, an investor allocates funds among the available assets. The objective is to select the best portfolio among a set of feasible portfolios. The quality of a portfolio is measured in terms of different factors, such as the (expected) return and risk of that portfolio. That is, the portfolio optimization problem is naturally a multi-objective problem where there is a trade-off between the risk and return: typically to obtain a higher expected return a company is likely to face a higher risk and vice versa. For this reason, the objective usually represents a trade-off between the expected return and the risk. This trade-off is based on the investor's risk aversion. A famous model for optimization is the traditional Markowitz Mean-Variance Portfolio Problem (Markowitz, 1952). In the Markowitz model, the trade-off between expected return and risk is represented by a weighted combination of expected return and the variance of the return. Even though the Markowitz model is theoretically very strong, the standard model has in practice a lot of criticism since the setting is not realistic. A nice literature collection about the criticism on the practical concerns of the Markowitz model is discussed by (Cesarone, Scozzari, & Tardella, 2013). In this paper, they also address the importance of extending the simple Markowitz model with *cardinality* and *quantity* constraints. We propose a solution method for the traditional Markowitz Mean-Variance Portfolio Problem extended by some practical

CONTACT Aras Selvi. Email: a.selvi@tilburguniversity.edu

constraints including *cardinality* and *quantity* constraints.

We divide the constraints into two sets. The first group, the 'easy' ones, set the maximum allowed deviation of the assets in each period where deviation means the difference between the selected portfolio and the benchmark. Deviation limits on the assets are set for multiple features based on the market capital quantile, sector and benchmark. In addition, the constraints that specify an interval for the active weights are being added, which are set of linear inequalities. We define the *basic model* to be the Markowitz model including these 'easy' constraints. The *basic model* is a Convex Quadratic Problem, for which there exist several solvers. We compare several solvers and decide using Mosek considering the fast outputs given by this solver (Mosek, 2010). The basic model can be solved very fast, and in our methodology and algorithms we will use this power in order to be able to handle the hard constraints.

The second group of constraints are the hard ones in terms of computational complexity. This group consist of the *cardinality constraints*, which introduce a combinatorial structure, and the *tracking error constraints*, which are not convex. The cardinality constraints bound the number of active assets in the portfolio, which is a results of the practical considerations on the number of assets selected. In addition, the tracking error constraints bound the correlation between the returns of the selected portfolio and the benchmark portfolio. These *tracking error constraints* are solved by continuous adjustment of the risk-parameter. The cardinality constraints are handled by a asset selection sub-problem based on the marginal effect of investing in those assets. Our method resembles a variation of Column Generation that we extend to provide a good solution to the mixed integer quadratic problem. In the next section, we will show that the existing methodologies are not powerful enough in the existence of the constraints that we evaluate. The existing methods are based on heuristics and evolutionary computing, while our approach relies on the marginal values. Overall, the novelty of our work is the new approach which is not studied in the literature to the best of our knowledge.

The rest of this paper is structured as follows. In Section 2 we introduce the problem mathematically, provide the notation and the formulation. In Section 3 we give the methodology to solve the problem, explain the reasoning and provide a pseudo-code for the algorithm. In Section 4 we describe the S&P500 data we use, give the performance measurements and then share the performance of our solution by analyzing the results. The conclusion, alternative methods to extend the solution and final remarks are in Section 5.

2. Preliminaries

2.1. Notation and the Basic Model

Let us introduce the notation of the input of the problem:

- Risk-Ω: Return covariance matrix. This is a symmetric covariance matrix, i.e., the diagonal elements are the variances of the corresponding asset, and the off-diagonal elements are the covariances between the corresponding two assets.
- Alpha score- α: The performance indicator of an asset. Alpha score is the expected return of an asset.

- Identifier - SEDOL: We use Stock Exchange Daily Official List, SEDOL, as the identification key of the stocks
- Sector: This is a set containing different sectors. Each asset belongs to a sector.
- Beta - β: Asset's sensitivity relative to the equity market return as a whole.
- Benchmark weights - w_{bench}: A benchmark is provided and the weights denote the proportion of the funds invested to the corresponding asset.
- MCAPQ: Market cap captures the asset's capitalization size relative to the market. A smaller MCAPQ means a bigger company. MCAPQ is between 1-5, each reflecting 20% quintile. For example, a rank of 1 means the company belongs to the largest 20% of companies, 2 means 40% and so on.
- 4-Weekly returns- r: The 4-week forward real return faced in the time period. This will be used to evaluate our optimization results.

We also introduce the variable and parameter notations below:

- w: total portfolio
- w_i: proportion of the investment on asset i
- d_i: active weight held in asset i, stands for the difference between the weight we invest and the benchmark weight on the asset i. It is found as: $w_i - w_{bench}$
- λ: The tuning parameter. This parameter is for the balance between the risk and return of the Markowitz objective function. $\lambda = 0$ means the objective is equivalent to only minimizing the risk while a big value, like $\lambda = 10$ is very aggressive with the return.

The underlying problem closely resembles a standard Markowitz portfolio optimization problem for an investor with mean-variance preferences. In other words, we just try to find out how the risk-averse investor should allocate his/her wealth to assets optimally, i.e. to find a portfolio that best suits investor conservativeness, goals and possibilities. This can be especially clear if we consider the first few lines of the optimization problem:

$$\begin{aligned} \min_{d} \ & d^T \Omega d - \lambda d^T \alpha \\ \text{s.t. } & w_i \geq 0 \quad \forall i \quad (\textit{non-zero weight allocation}) \\ & \sum_i w_i = 1 \quad (\textit{entire portfolio invested}) \end{aligned}$$

This is a convex quadratic minimization problem involving portfolio variance and returns. If we change the objective to $w^T \Omega w - \lambda w^T \alpha$ for a moment, we obtain almost a classical portfolio optimization problem, where the first constraint restricts the weight allocation to be non-negative (short position on an asset is not allowed), and the second constraint ensures that the total allocation of asset weights sums up to 1 (simply indicating that all the funds have to be invested in our asset universe). The first constraint is a linear inequality while the second one is a linear equality, thus both are convex. As a result, from an optimization perspective, this investor problem does not create any computational difficulties to solve (i.e. find the optimal weights).

The relevance of this simple exercise to an investor is very crucial from an economic point of view. A short reason involving more technical terms is that it helps to identify the portfolio that maximizes investor's utility function (which can vary depending on his/her willingness and ability to bear risk). Clearly, the main assumption and the intuition of the model is that the investor considers not only about the expected return, but also about its dispersion (variance). Since the investor is usually assumed to dislike risk (which is quite a realistic assumption given large experimental evidence involving, for instance, lotteries (Holt & Laury, 2002)), in this study we try to minimize the portfolio variance. Though, each investor considers not only risk, but also the respective expected return that investments can generate. Thus, the objective function is adjusted by an extra term involving the return score (α) and the level of risk-tolerance of an investor (λ). It has been assumed that the investor cares here only about the variance of portfolio returns and we abstract here from all the higher moments like skewness (tail risk) or kurtosis (fatness of tails). Thus, up until this point we use exactly the same assumptions under which in a frictionless environment a mean-variance efficient frontier can be easily constructed (as in (Cochrane, 2009)) and the optimal portfolio chosen.

Though, the novelty here is that our portfolio optimization contains several extensions that bring the analysis closer to the practical implementation and more recent developments in the literature. All of these extensions are financially quite intuitive. Firstly, it is obvious that for an investor, let's say in mutual funds, it is very important to know how active the fund manager is. After all, any active portfolio management services involve certain costs (management fees) which are considered to be a compensation for a portfolio managers effort to generate positive abnormal returns (in finance jargon, positive alphas). One way to measure this active effort is to use the active share concept originally proposed by K. M. Cremers and Petajisto (2009) and further analyzed by M. Cremers (2017). The example with a mutual fund manager is quite illustrative, but the concept is quite broad to implement in any portfolio selection procedure and measure the relative activeness of a portfolio. Since by construction the portfolio fully invested in a benchmark has a zero active share, our optimization problem simply tries to find an optimal deviation from the benchmark, given the risk-tolerance of an investor (for a moment we abstract from other constraints). If the risk-tolerance is extremely low (λ equals 0), then the optimal portfolio is very close to the benchmark. However, if the investor is willing and able to bear risk (λ is high), most likely the portfolio weights will substantially deviate from the benchmark. Thus, it's a first simple reformulation of the investor's problem. The active share is controlled by constraint (*Active share*).

$$0.6 \leq 1 - \sum_i min(w_i, w_{bench_i}) \quad \leq \quad 1 \qquad (Active\ share)$$

Another real-world advantage of this study is illustrated by the following four constraints which essentially restrict the individual positions in individual assets, sectors or even in beta risk:

$$
\begin{aligned}
-0.05 \leq d_i &\leq 0.05 \quad \forall i \quad (\textit{Deviation from Benchmark Weight}) \\
-0.1 \leq \sum_{i \in sectorj} d_i &\leq 0.1 \quad \forall j \quad (\textit{Sector Active Weight}) \\
-0.1 \leq \sum_{i \in MCAPQk} d_i &\leq 0.1 \quad \forall k \quad (\textit{Market Capital Quintile Active Weight}) \\
-0.1 \leq \sum_{i} d_i \beta_i &\leq 0.1 \quad \quad (\textit{Beta Active Weight})
\end{aligned}
$$

Namely, deviation from benchmark weight constraint restricts the individual deviation from the benchmark weight from -5% to +5%. The assets in the given problem are being distributed into separate sectors. Thus, sector active weight constraint restricts the total summed deviation for each of the sectors to be less than 10%. Another concept has been introduced in market cap quintile constraint, which indicates how each assets capitalization size is relative to the other assets. This parameter divides the assets into quintiles based on their size. This constraint ensures that the total summed deviation per quintile does not exceed 10%, similar to this constraint, thus ensuring weight allocation of the assets are distributed across sectors the assets belong to and also according to their capitalization size based on the benchmark weights given to assets. This can be clearly relevant from a risk management perspective and other policies which limit the portfolio exposure to idiosyncratic (e.g. firm specific) shocks. Beta active weight constraint ensures the total sum of the product of beta, a measure of each of the assets sensitivity to the whole market, and the deviation, the beta active weight to be restricted to deviation of 10%. Overall, these constraints each can be converted into two matrices of linear inequalities and are convex. Thus, again this part of the problem does not impose much of computational challenges.

2.2. Advanced Extensions

The basic model and extensions discussed in the previous sub-section is computationally easy to solve. In this section, the constraints we introduce are the ones which make the problem computationally intractable.

The active share is closely related to the tracking error calculation (introduced by Roll (1992), analyzed by Rudolf, Wolter, and Zimmermann (1999)) which can be also seen as a measure of how the portfolio returns are dispersed relative to the benchmark. Tracking error constraint is exactly dealing with that: the tracking error comprising of the square root of the product of deviation squared and the variance-covariance matrix to be between 5% and 10%. The left part of the constraint is concave and the right part is convex. As the left side of the constraint is always satisfied, this constraint is also formulated and considered as convex for this project.

$$
0.05 \leq \sqrt{d^T \Omega d} \quad \leq \quad 0.1 \qquad (\textit{Tracking error constraint})
$$

A big extension of this model is the cardinality constraint which closely relates to the idea that financial markets are not frictionless and there are substantial transaction costs and divisibility limitations. Though, in this analysis we ignored the transaction cost, the re-balancing of portfolio which contains hundreds of assets can be extremely costly and erode all the net returns. This motivates why there is a value of having just a limited amount of assets in the portfolio. Another idea is that a small amount of portfolio is also easy to oversee and analyze individually namely, a portfolio of 70 could be actively monitored, inspected, if needed. Thus, the cardinality constraint is clearly a big step towards a practical implementation of Markowitz theoretical model. Cardinality constraint is the constraint that ensures the number of assets to be invested in to be at least 50 and at most 70. This constraint is an integral constraint, and it is computationally very hard to solve. Therefore this hard constraint is tackled by using a column generation algorithm, which is further discussed in the methodology section.

$$50 \leq card(w_i \neq 0) \quad \leq \quad 70 \qquad (Cardinality)$$

In the literature there is a significant concentration around the real-life trading costs and monitoring availability, thus such extensions as the cardinality constraints are being studied. Chiu and Li (2006) provide an exact solution for the classical mean variance optimization problem where they also study discrete extensions such as the cardinality constraint as well as transaction costs. The approved algorithms in the literature are mainly based on local search and multi-objective evolutionary algorithms. The effect of genetic algorithms, tabu search and simulated annealing on the cardinality extension is seen in the detailed work of Chang, Meade, Beasley, and Sharaiha (2000). The extensions to the evolutionary algorithms, such as memetic algorithms are also studied for tackling the cardinality constraint (Streichert, Ulmer, & Zell, 2004). More experiments and comparisons in evolutionary algorithms can be found in the work of Anagnostopoulos and Mamanis (2011).

On the other hand, the novelty of our work, with respect to existing literature is our contribution by using a column generation approach to tackle the cardinality constraint as well as satisfying the other extra constraints for diversification.

3. Solution Approach

In our methodology we divide the problem into a master problem and a sub-problem. Since the sub-problem is the convex part of the the problem, this problem can be solved using quadratic programming. The convex sub-problem is given by $Psim$:

$$
\begin{array}{rrrlll}
Psim = \min\limits_{d} & d^T \Omega d - \lambda d^T \alpha & & & & \\
\text{s.t.} & w_i & \geq & 0 & \forall i & (1) \\
& \sum\limits_i w_i & = & 1 & & (2) \\
& |d_i| & \leq & 0.05 & \forall i & (3) \\
& |\sum\limits_{i \in sector j} d_i| & \leq & 0.1 & \forall j & (4) \\
& |\sum\limits_{i \in MCAPQk} d_i| & \leq & 0.1 & \forall k & (5) \\
& |\sum\limits_i d_i * \beta_i| & \leq & 0.1 & & (6) \\
& \sum\limits_i |d| & \geq & 1.2 & & (7)
\end{array}
$$

For the Psim problem constraint (7) only needs to be used when the sum of the bench weights for the assets we invest in is smaller than 0.4. When this is the case we have a (convex) Quadratic Problem, for which we have very efficient ways to solve. Constraint (7) of the Psim problem is equivalent to the original $ActiveShare$ constraint. The following lemma shows this equivalence.

Lemma 3.1. *Under Full Portfolio and non-negative investment, the following equivalence holds:*

$$\sum_i min(w_i, w_{i,bench)} \leq 0.4 \leftrightarrow \sum_i |d_i| \geq 1.2 \tag{1}$$

Proof. w_i can be rewritten as $w_{bench_i} + d_i$ which implies the equivalence (2) and vice versa w_{bench_i} can be rewritten as $w_i - d_i$ which implies the equivalence (3), by $FullPortfolio$ and non-negative investment. The equivalences (2), (3) and (4) conclude Lemma 3.1.

$$\begin{aligned} &\sum_i min(w_i, w_{bench_i}) \leq 0.4 \\ \Leftrightarrow &\sum_i min(w_i, w_i - d_i) \leq 0.4 \\ \Leftrightarrow &\sum_i 1 + min(0, -d_i) \leq 0.4 \end{aligned} \tag{2}$$

$$\begin{aligned} &\sum_i min(w_i, w_{bench_i}) \leq 0.4 \\ \Leftrightarrow &\sum_i min(w_i, w_i - d_i) \leq 0.4 \\ \Leftrightarrow &\sum_i 1 + min(0, -d_i) \leq 0.4 \end{aligned} \tag{3}$$

$$\begin{aligned} &\sum_i 2min(w_i, w_{bench_i}) &&\leq 0.8 \\ \Leftrightarrow &1 + \sum_i min(0, d_i) + 1 + \sum_i min(0, -d_i) &&\leq 0.8 \\ \Leftrightarrow &\sum_i |d_i| &&\geq 1.2 \end{aligned} \tag{4}$$

$\square$

To fulfill this constraint we randomly deselect assets we can invest in to reduce the sum of the bench weights of the assets we consider investing in.

We solve this convex sub-problem (Psim) many times therefore it is important that we use an efficient solver. To solve Psim we use the Yalmip (Lofberg, 2004) environment in Matlab with the solver Mosek. This solver was tested the most efficient solver for our problem of the solvers we tested. The test was performed on the initial portfolio for the 31^{th} of January 2007, in the test all solvers got the optimal solution.

For the initialization of the problem we start with solving the basic problem (Psim) for the assets in which previous re-balance portfolio where selected and the investments on the other assets are fixed on 0. In the case when a selected asset from the previous balance portfolio is no longer in the market, it is replaced by the possibility of investing in a random other asset for the initialization of this iteration.

The first non-convex constraint ($0.05 \leq \sqrt{d'\Omega d}$) and convex constraint ($\sqrt{d'\Omega d} \leq 0.1$) are being solved during each step by re-adjusting the λ when the constraint is violated. This is possible since the square root is an increasing function for positive values, and $d'\Omega d$ is positive for any d since Ω is positive semi-definite. Therefore $\sqrt{d'\Omega d}$ is increasing with $d'\Omega d$. Notice that this factor is also in the objective function, hence by adjusting λ we can make the convex program change the priority: decreasing λ (ergo reducing $d'\Omega d$) gives more importance on minimizing the tracking error, or increasing λ gives more weight on maximizing the expected revenue by allowing a higher risk.

The last non-convex part of the problem is the cardinality constraints that we handle using a variation of Column Generation by Ford Jr and Fulkerson (1958); which we call CG2. The columns represent the individual asset-possibilities to invest in. In our method we select the assets with the lowest weight and all assets we do not invest in (investment$< 10^{-5}$) and remove them. Then to adjust the cardinality, ergo the number of assets that we consider in the portfolio to 70, we reselect assets based on the best marginal effects of investing in those assets. The marginal effect for asset i (δ_i) is obtained from the effect from the shadow-values (k_i) and the direct marginal change to the objective function (m_i). The marginal direct effect on the objective function is obtained from formula (M1).

$$m_i = d^T\Omega_{\cdot,i} - \lambda\alpha_i - 2\lambda 10^{-3}\mathbf{I}(w_{i,t}^{Pre} \geq 10^{-5}) \tag{M1}$$

The indirect effect can be obtained by the shadow-values, which are typically described for inter linear problems, but can be used for nonlinear models as well. A shadow-value of a constraint of the form $ax \leq b$, is the marginal effect on the objective function if the right hand side of the constraint would be slightly smaller. The Psim problem we iteratively solve has only linear constraints therefore we can write the constraints as $\bar{A}x \leq b$ and obtain the shadow-values s corresponding to each of the constraint. This is used to compute the indirect effect of investing in an asset i (k_i) by formula M2.

$$k_i = s \cdot A_{\cdot,i} \tag{M2}$$

Then the marginal effect of investing in asset i (δ_i) is obtained by using formula (M3).

$$\delta_i = m_i - k_i \tag{M3}$$

The assets with the most negative marginal effect are added to the list of assets that we can invest in, such that the cardinality is set to 70. Then the convex problem ($Psim$) is solved for the 70 assets in the list. With this way, the dimension of our variable list is set to 70 assets for each time solving the convex problem ($Psim$). The

reason we choose allowing to invest in a fixed number of 70 assets is because investing in less than 70 assets leads to a smaller feasible area in the $(Psim)$ problem and therefore leads to a worse objective function. We then could improve our portfolio objective function by allowing to invest in more assets.
This process is repeated until the time limit of 2 minutes 50 seconds (less than 3 minutes) is reached. Algorithm 1 describes the process of optimizing the portfolio step-by-step.

Data:

α:	Mean return parameter,
Ω:	Variance-Covariance matrix,
w_{t-1}:	Previous period portfolio,
w_{bench}:	Benchmark portfolio,
$\bar{\lambda} = 5$:	Risk parameter
$\lambda = 5$:	Adjusted risk parameter
w_t^{Pre}:	Adjusted obtained weights in period $t-1$
$Removed = I(w_{t-1} = 0)$:	Assets we do not invest in in period $t-1$
$nonzeros = \sum_i \neg Removed_i$:	Number of assets in which we do not invest
$w_t = Psim : w(\neg Removed)$:	Initial portfolio
$d_t = w_t - w_{bench}$:	Difference in weights compared to the benchmark
$\epsilon = 10^{-3}\lambda$:	Turnover penalty

Result: $w_{t,best}$, $d_{t,best}$

```
while time ≤ 2.9 minutes do
    while Σ_Removed |w_bench(Removed)| < 0.6 do
        Add arbitrary asset to Removed
    end
    Obtain w_t and d_t from Psim with w_t(Removed) = 0
    if sqrt(d^T Ω d) < 0.05 then
        Set λ = 0.9λ
    else
        if sqrt(d^T Ω d) > 0.1 then
            Set λ = 1.1λ
        else
            if Psim(w_t, λ̄) + ε turnover(w_t, w_bench) < Psim(w_t,best, λ̄)) + ε
              turnover(w_t,best, w_t^Pre) then
                Set w_t,best = w_t
            end
            set oldRemoved = Removed
            set Removed = oldRemoved ∪ {i : argmin_i w_t(i) | i ∈ ¬Removed}
            set w_t(Removed) = 0
            set Removed = (w_t < 10^-5)
            set nonzeros = Σ_i ¬Removed
            Select 70 − nonzeros assets based on with best marginal effect at w_t
              on the objective
            newRemoved = Removed excluding the selected assets
            if (oldRemoved = newRemoved) then
                reselect 70 − nonzeros random non-removed assets
                newRemoved = Removed excluding the reselected assets
            end
            Removed = newRemoved
        end
    end
end
```

Algorithm 1: CG2 for time period t

4. Application

We apply the solution approach on the data set used and evaluate our solution with relevant performance metric. Further details are given in this section.

4.1. Data

The problem is defined by Principal Inc, a global investment company. The company also provided us the data with which we test our solution approach. We are provided with 10 years of time series data starting at 2007-01-03 which we refer to as S&P 500 data and are provided with estimators for the 4 weeks variance covariance matrix of the return of those assets over the same time interval. The entries are being updated every 28 days (4 weeks). The elements of each entry in a date are: the identifier which is the unique identifying SEDOL code, the sector of the company which will be used to set the sector active weight in an interval, the beta value which reflects the volatility of an asset compared to the whole market, the alpha score which is the performance estimation shown as the expected return, name of the asset, benchmark weight which is the investment amount in the provided benchmark and the market cap quantile which reflects the size of the asset's company by the quantile in the market.

We are also provided with the upper half of a symmetric variance-covariance matrix. An entry of the matrix is the covariance between the two assets, where the diagonal entries give the variance of each asset. We assume that the upper half contains the half covariance between each pair of assets i and j.

In the results sheet we have the historical returns for each 4 week period. We assume that the returns on a date are the results of the investment done 4 weeks before. So, the return entry at the first date 2007-01-03 is actually the return coming from the investment made on 2006-12-06.

4.2. Performance Measurements

For the calculations of our performance measures we use the turnover adjusted returns. The turnover is penalized by 0.5% in our method to avoid high costs corresponding to changing portfolio. The turnover is calculated by the following two formulas.

$$w_{i,t}^{Pre} = \frac{w_{i,t-1} * (1 + r_{i,t-1})}{\sum_i w_{i,t-1}(1 + r_{i,t-1})}$$

$$turnover(w_{i,t}, w_{i,t}^{Pre}) = \sum_i |w_{i,t} - w_{i,t}^{Pre}|$$

We evaluate our solution by comparing the following results with the benchmark results both including and excluding the turnover costs:

- **Cumulative Return** gives what the chosen portfolio scheme has done to the initial investment at the end of the very last period. This is the most straightforward calculation to see the final return.

- **Annual Return** converts the cumulative period into an average annual return. If the returns of each year were all equal to the Annual Return, the cumulative return would still be the same.
- **Annualized Excess Return** is the difference between the annual return of the proposed portfolio and the annual return of the benchmark.
- **Tracking Error** is the standard deviation of the difference between the returns of the chosen portfolio and the benchmark. We use **Annualized Tracking Error** by considering the number of re-balances in a year. By looking at this, we can see if the portfolio follows the benchmark closely or not.
- **Sharpe Ratio** is calculated by subtracting the best risk-free option from the final return and dividing it by the standard deviation of the observed returns. Since risk-free option is not investing at all, we take this value as 1. This ratio helps us to understand how choosing riskier assets affect the extra return we have compared to the risk-free option. This ratio is helpful to see the payoff between the return and the risk.
- **Information Ratio** is found by dividing the difference of the returns of portfolio and benchmark by the tracking error. Information Ratio is similar to Sharpe Ratio, however IR gives the risk and return by taking benchmark as the base case. Higher IR is given by higher difference between portfolio and benchmark, also lower tracking error. So higher IR can be used to see "how closely the benchmark is followed, with how much better return".

4.3. Results and Analysis

To check the performance of our method we compare our portfolio's performance on the Prinicipal dataset to the benchmark. In this comparison, we do not consider turnover costs for the returns of the benchmark portfolio. We therefore split our result analysis into two parts. We first compare the results of our method excluding turnover costs. In the second part, we do include the turnover costs in our portfolio.

It can be seen from our performance statistics metrics in Table 1 that the performance of our method excluding turnover costs outperforms the benchmark. In each of the performance statistics, our model's results is better than the performance statistics of the benchmark.

Table 1.: Portfolio Performance Statistics Excluding Turnover costs.

2007-01-01 to 2016-12-31	**Portfolio**	**Benchmark**
Cumulative Return	275.65%	239.10%
Annualized Return	14.15%	12.99%
Annualized Excess Return	1.16%	–
Sharpe Ratio	55.75	40.81
Information Ratio	24.25	–

In practice, the turnover costs are very relevant. This can also be seen in Figure 1, which plots the turnover costs of our portfolios over the review periods. These turnover costs are calculated over a 0.5% cost per unit turnover. Figure 1 indicates that our method does not focus enough on having a small turnover.

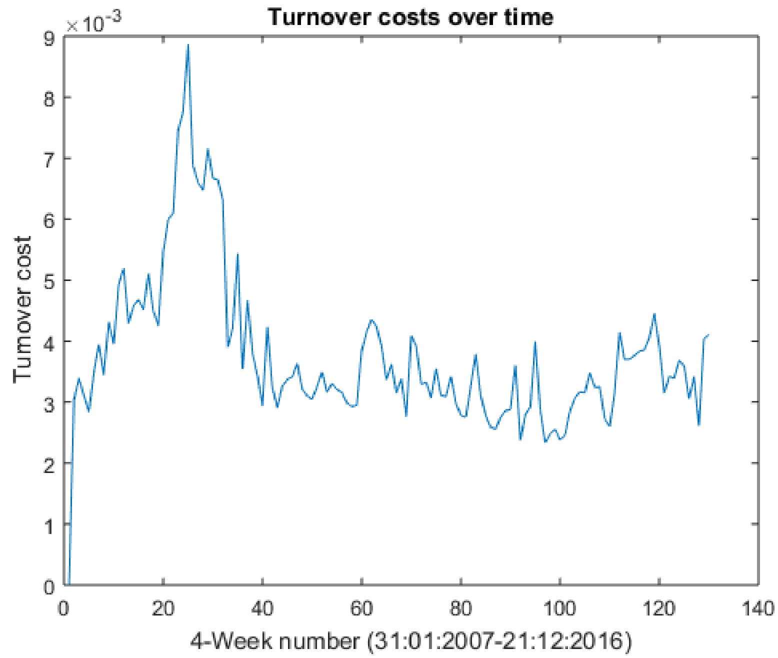

Figure 1.: Turnover costs: based on 0.5% turnover costs per unit turnover

The relative high turnover costs of our portfolios leads to that our method performs slightly worse than the benchmark. These performance statistics can be seen in Table 2. These differences in performance suggest that our method adjusts the portfolio too much after each review period. That is why, for future development of this model, it is important to focus more on the minimization the turnover costs. In the section *Alternative Methods*, we propose a few adjustments to the method to achieve lower turnover.

Table 2.: Portfolio Performance Statistics Including Turnover costs

2007-01-01 to 2016-12-31	**Portfolio**	**Benchmark**
Cumulative Return	207.18%	239.10%
Annualized Return	11.88%	12.99%
Annualized Excess Return	–1.11%	–
Annualized Tracking Error	5.54%	–
Sharpe Ratio	41.95	40.81
Information Ratio	–20.75	–

The difference in returns is due to the 0.5% turnover cost per unit turnover. Figure 3 illustrates the turnover costs per 4 week review date beginning in 31th January 2007 to 21th December 2016. Figure 2 shows that the resulting returns including turnover costs are on average not as high as the benchmark. This is in part compensated by a smaller risk, which is shown as a smaller spread of the box plot. Therefore our resulting portfolio can be seen as more robust.

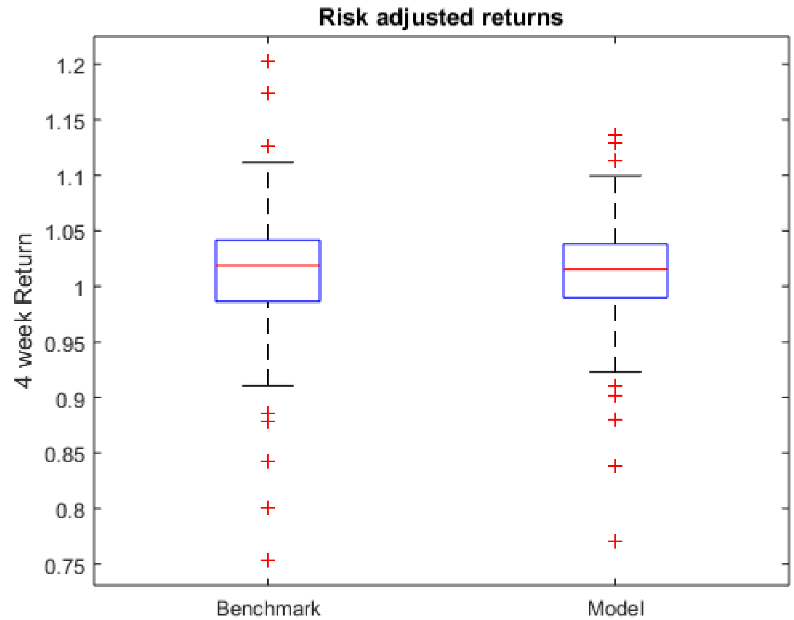

Figure 2.: Boxplot of the turnover adjusted 4-week returns

Next, we show the performance of our method over each review period. These results are shown in Figure 3. Figure 3 shows that the model follows the benchmark quite closely and our portfolios even lead to smaller peeks, this can be interpreted as our method leads to more robust portfolios than the benchmark.

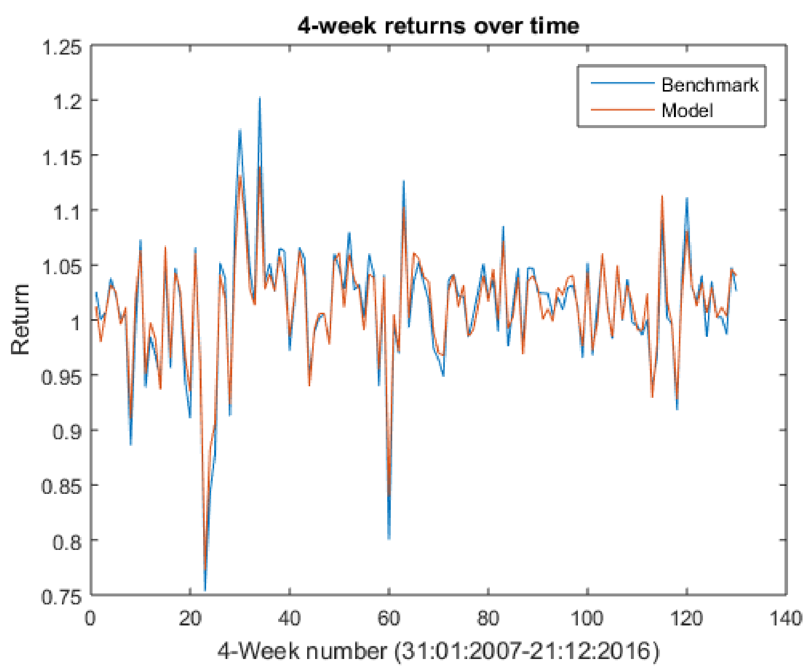

Figure 3.: Turnover adjusted 4-week returns per review period

5. Discussion

Our method shows promise in solving linear or in this case quadratic cardinal selective problems. Table 1 shows that the performance without the turnover costs is outperforming the benchmark, in terms of both risk and expected return. But the method could be improved in terms of turnover, Figure 3 and Table 2 both show that the method has difficulties with the limitations of the turnover cost. Therefore, for a future research we recommend to constrain the allowed turnover. Although the turnover adjusted performance in terms of cumulative return is slightly worse than the benchmark, this is partially due to the absence of the turnover costs for the benchmark. Overall, this method incorporates a comprehensive model that performs close to the benchmark keeping many practical issues in consideration.

Since in our model we do not allow infeasible solutions to exist, it is necessary on the one hand to make our model more flexible or to allow (small) errors in the constraints, if no solution has been found. Another improvement that could be made is to consider multiple period portfolio selection to reduce the significant turnover costs. The turnover costs can also be reduced by considering the changes in the performance parameters α and Ω of the assets over time, which can be used to make more consistent portfolio or find trends in the performance of the assets.

One of the other methods is a column generation algorithm based on a random selection, this would make the problem more flexible. In this method, the selection of the assets which leave the investment portfolio and enter the portfolio of 70 assets are chosen randomly. In another method they are chosen probabilistically with probabilities being proportional to the marginal effects or Reduced Cost, where the Reduced Costs are determined by the shadow values. This is a possible improvement to our method.

In our algorithm, for each time period we have the initial portfolio referring back to the previous portfolio and the algorithm changes it by improvements. However, since we initiate the previous portfolio, the portfolio will end up close to the previous portfolio. With this way, we made sure the algorithm is efficient and can be terminated timely. This allows the model to produce well-based results on even a much larger scale dataset (10-fold or 20-fold). On the other hand, one would allow to look at every possible asset to initialize the portfolio for the algorithm in each time period. This will increase the computational burden, but will allow to diversify the selections more.

For future development of the method, we believe that the development of the parameters α and Ω, may be a good indicator of reliability of a portfolio. When incorporating this multiple period analysis for the portfolio selection, it could lead to less turnover without losing a lot of expected returns or leading to significantly higher risk.

In the whole project, it has been assumed that α and Ω are deterministic. However, in the real-world flow, α and Ω have an uncertainty. In order for the optimization problem to account the uncertain nature of these two variables, concepts of robust optimization can be used. However, there is a limitation to the existing popular robust optimization methods in regard to this problem. Ellipsoidal method has widely been used to construct the uncertainties of α by using the variance-covariance matrix, Ω. In the ellipsoidal uncertainty model, Ω is deterministic and α is stochastic. To study the case where α is stochastic and ω is deterministic, one can see the robust formulation of (Kim, Kim, & Fabozzi, 2014):

$$min \quad d^T \Omega d - \lambda(\mu^T d - \sigma ||\Omega_\mu^{\frac{1}{2}} d||_2),$$

where μ is the mean of the deviations and Ω_μ is Ω/(# observations).

Another option is semi-parametric estimation. Let g is a known function, based on historic data to be tested and $y_t = g(x_t^T \beta) + \epsilon_i$ where y_t is the a vector of mean return and the rolled out vector of historically estimated covariances $(r_i * r_j)$ for time period t, x_t is vector of of returns and the crossproduct of returns for time period t. β is computed by the minimization of $\sum_t (y_t - g(x_t^T \beta))^2$. This model only assumes that g is known and y_t depends on x_t only through $x_t^T \beta$.

References

Anagnostopoulos, K. P., & Mamanis, G. (2011). The mean–variance cardinality constrained portfolio optimization problem: An experimental evaluation of five multiobjective evolutionary algorithms. *Expert Systems with Applications*, *38*(11), 14208–14217.

Cesarone, F., Scozzari, A., & Tardella, F. (2013). A new method for mean-variance portfolio optimization with cardinality constraints. *Annals of Operations Research*, *205*(1), 213–234.

Chang, T.-J., Meade, N., Beasley, J. E., & Sharaiha, Y. M. (2000). Heuristics for cardinality constrained portfolio optimisation. *Computers & Operations Research*, *27*(13), 1271–1302.

Chiu, M. C., & Li, D. (2006). Asset and liability management under a continuous-time mean–variance optimization framework. *Insurance: Mathematics and Economics*, *39*(3), 330–355.

Cochrane, J. H. (2009). *Asset pricing: Revised edition.* Princeton university press.

Cremers, K. M., & Petajisto, A. (2009). How active is your fund manager? a new measure that predicts performance. *The Review of Financial Studies*, *22*(9), 3329–3365.

Cremers, M. (2017). Active share and the three pillars of active management: Skill, conviction, and opportunity. *Financial Analysts Journal*, *73*(2), 61–79.

Ford Jr, L. R., & Fulkerson, D. R. (1958). A suggested computation for maximal multi-commodity network flows. *Management Science*, *5*(1), 97–101.

Holt, C. A., & Laury, S. K. (2002). Risk aversion and incentive effects. *American economic review*, *92*(5), 1644–1655.

Kim, J. H., Kim, W. C., & Fabozzi, F. J. (2014). Recent developments in robust portfolios with a worst-case approach. *Journal of Optimization Theory and Applications*, *161*(1), 103–121.

Lofberg, J. (2004). Yalmip: A toolbox for modeling and optimization in matlab. In *Computer aided control systems design, 2004 ieee international symposium on* (pp. 284–289).

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

Mosek, A. (2010). The mosek optimization software. *Online at http://www. mosek. com*, *54*(2-1), 5.

Roll, R. (1992). A mean/variance analysis of tracking error. *Journal of portfolio management*, *18*(4), 13–22.

Rudolf, M., Wolter, H.-J., & Zimmermann, H. (1999). A linear model for tracking error minimization. *Journal of Banking & Finance*, *23*(1), 85–103.

Streichert, F., Ulmer, H., & Zell, A. (2004). Evolutionary algorithms and the cardinality constrained portfolio optimization problem. In *Operations research proceedings 2003* (pp. 253–260). Springer.