Journal of Mathematics and Modeling in Finance (JMMF)
Vol. 4, No. 1, Winter & Spring 2024
Research paper

An L_1 then L_0 approach to the cardinality constrained mean-variance and mean-CVaR portfolio optimization problems

Maziar Salahi[1], **Tahereh Khodamoradi**[2]

[1] Center of Excellence for Mathematical Modeling, Optimization and Combinatorial Computing (MMOCC), University of Guilan, Rasht, Iran.
salahim@guilan.ac.ir

[2] Department of Applied Mathematics, Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran.
ta_khodamoradi@yahoo.com

Abstract:
Cardinality constrained portfolio optimization problems are widely used portfolio optimization models which incorporate restriction on the number of assets in the portfolio. Being mixed-integer programming problems make them NP-hard thus computationally challenging, specially for large number of assets. In this paper, we consider cardinality constrained mean-variance (CCMV) and cardinality constrained mean-CVaR (CCMCVaR) models and propose a hybrid algorithm to solve them. At first, it solves the relaxed model by replacing L_0-norm, which bounds the number of assets, by L_1-norm. Then it removes those assets that do not significantly contribute on the portfolio and apply the original CCMV or CCMCVaR model to the remaining subset of assets. To deal with the large number of scenarios in the CCMCVaR model, conditional scenario reduction technique is applied. Computational experiments on 3 large data sets show that the proposed approach is competitive with the original models from risk, return and Sharpe ratio perspective while being significantly faster.

Keywords: Portfolio optimization, cardinality constraint, mean-CVaR.
Classifications: 91G10, 90C11.

1 Introduction

Portfolio selection is an important strategy in financial markets, which plans to choose a the best set of assets from the portfolio. Harry Markowitz proposed the well-known Mean-Variance (MV) model taking into account a trade-off between risk and return [17]. However, it lacks of not considering various real-world limitations. Later, his model was enhanced to include more realistic features like multi-period

[1] Corresponding author

Received: 21/02/2024 Accepted: 16/05/2024
https://doi.org/10.22054/JMMF.2024.78407.1125

optimization [16], transaction costs [18], cardinality constraint which restricts the number of assets in the portfolio and lower and upper bounds on the proportion of each asset in the portfolio [3], and option [9, 10].

Incorporating cardinality constraints in a portfolio optimization model, makes it an NP-hard problem and thus challenging. To deal with this issue, several solution methods are examined in the literature such as exact, approximate, and heuristic methods. Exact methods such as interior-point methods, guarantee optimal solutions in finite running time while may fail to compute them for large dimension [6]. Relaxation methods are approximate methods that provide solutions of compromised quality in reasonable time. In the sequel we review some approaches that are used to deal with the cardinality constraints in the MV and mean-CVaR models that are the models dealt in this paper.

Shaw et al. [21] applied Lagrangian relaxation to find lower bounds for Cardinality Constrained MV (CCMV) model. Xie et al. [23] used a randomized algorithm to obtain good feasible solutions for the CCMV model. Bertsimas and Shioda, [2] considered subset selection problems in regression and portfolio selection in asset management for CCMV portfolio optimization problem and proposed branch-and-bound algorithms that take advantage of the special structure of this problem. In [4] the authors constructed convex relaxations for the CCMV portfolio optimization model via a new Lagrangian decomposition scheme and reduced its dual problem to a second-order cone programming problem. It gives better bound than the continuous relaxation of the standard mixed integer quadratically constrained quadratic program reformulation. Ramshe et al. [19] proposed a firefly algorithm to solve the CCMV portfolio optimization problem. They compared the performance of this method with some other available techniques such as Genetic algorithm, Simulated Annealing, Tabu Search, and Particle Swarm Optimization. Recently, Leung and Wang [14] presented a collaborative neurodynamic optimization approach for CCMV model. They scalarized the expected return and risk as a weighted Chebyshev function. Also, Leung et al. [15] applied a two-timescale duplex neurodynamic approach for solving the CCMV portfolio optimization model and used particle swarm optimization to avoid local solutions. Also, most recently Zheng et al. [24] presented a new metaheuristic approach based on the Mayfly algorithm to solve the CCMV portfolio optimization problem. Their proposed algorithm includes a new approach to handle cardinality constraint, a new local search strategy and changes to the crossover operator. They performed comparison using five commonly used performance metrics.

In [25], the authors used continuous-relaxation-based heuristics to solve cardinality constraint mean-CVaR (CCMCVaR) model. The l_1 based approximation of the cardinality constraint are applied in [27]. Zhang et al. [28] proposed a new approach for CCMCVaR optimization model, using a relaxation formulation. They found feasible portfolios that are nearly as efficient as their non-cardinality constrained counterparts. Kobayashi et al. [13] reformulated the CCMCVaR portfolio optimization problem as a bilevel optimization problem and then developed a cutting-plane algorithm for solving the upper-level problem. Later, Zhao et al. [29] proposed an improved hybrid heuristic method for CCMCVaR optimization model.

Most recently, Dhingra et al. [5] proposed using L_1 or L_2-norm constraints instead of the cardinality constraints and reported promising numerical results by applying cross-validation to find appropriate upper bounds for the new norm constraints. In this paper, first we propose to solve the l_1-based relaxation of the l_0 constraint in the CCMV and CCMCVaR models. Then we remove those assets that do not significantly contribute in the portfolio. After that we solve the original models with the remaining assets, namely CCMV and CCMCVaR with less 0-1 variables. To deal with large number of scenarios in the CCMCVaR, we take advantage of the conditional scenario reduction (CSR) technique in [26] and [12]. Experiments on 3 large datasets show that the proposed approach gives comparable risks, returns, and Sharpe ratios compared to the original CCMV and CCMCVaR models that are solved directly by CVX-MOSEK.

The rest of the paper is organized as follows. In Section 2, we briefly review the CCMV model and then present a hybrid approach to solve it. In Section 3, we present the CCMVaR model, the CSR technique and the hybrid solution approach. Computational experiments are reported in Section 4 to show the efficiency of the hybrid approach to the direct one.

2 A Hybrid approach for the CCMV model

The CCMV model is as follows:

$$
\begin{aligned}
\min_x \quad & \lambda(x^T\Sigma x) - (1-\lambda)(r^T x - \sum_{i=1}^{N} \beta_i |x_i - x_i^0|) \\
\text{s.t.} \quad & 1^T x = 1, \\
& ||x||_0 \le K, \\
& \varepsilon \le x \le \delta, \quad i = 1, ..., N,
\end{aligned}
\tag{1}
$$

where $\Sigma \in R^{n\times n}$ is the covariance matrix, r is the vector of returns, 1 denotes the vector of all ones in R^n, $\lambda \in [0,1]$, ε and δ denote the vectors of lower and upper bounds of the assets, respectively and K stands for the desired number of assets in the portfolio. When $\varepsilon_i < 0$ then short selling is allowed, which is a technique used by investors to profit from the decline in value of an asset [11]. The term $\beta_i|x_i - x_i^0|$ in the objective function is the transaction costs for each asset i where x_i^0 is the proportion of the weight invested in asset i in the given portfolio [22]. Using 0-1 variables, the above problem can be rewritten as the following mixed-integer quadratic program form:

$$\begin{aligned} \min_{x,z} \quad & \lambda(x^T\Sigma x) - (1-\lambda)(r^T x - \sum_{i=1}^{N} \beta_i|x_i - x_i^0|) \\ \text{s.t.} \quad & 1^T x = 1, \\ & 1^T z \leq K, \\ & \varepsilon_i z_i \leq x_i \leq \delta_i z_i, \quad i = 1, ..., N, \\ & z_i \in \{0, 1\}, \quad i = 1, ..., N. \end{aligned} \tag{2}$$

As known, solving model (2) for large datasets is challenging. Thus to deal with this issue, here we propose to solve the following problem instead of (2):

$$\begin{aligned} \min_{x,z} \quad & \lambda(x^T\Sigma x) - (1-\lambda)(r^T x - \sum_{i=1}^{N} \beta_i|x_i - x_i^0|) \\ \text{s.t.} \quad & 1^T x = 1, \\ & ||x||_1 \leq UB, \\ & \varepsilon \leq x \leq \delta, \end{aligned} \tag{3}$$

Lemma 2.1. *If $UB = K \times \max_{i=1,\cdots,N}(|\varepsilon_i|, \delta_i)$, then model (3) is a relaxation of (2).*

Proof. For $UB = K \times \max_{i=1,\cdots,N}(|\varepsilon_i|, \delta_i)$, any feasible solution of (2) is also feasible for (3), thus (3) is a relaxation of (2). □

The relaxed model (3) is a convex quadratic program that can be solved efficiently even for large number of assets. After solving model (3), we solve CCMV model (2) with those assets contributing significantly in the portfolio, namely CCMV model with less 0-1 variables. This process is outlined in the algorithm below.

Algorithm 1

Input. ϵ, a small positive constant, K the desired number of assets and $\mathcal{I} = \{1, 2, \ldots N\}$;
Step 1. Solve (3) and let x^* be the optimal solution of it. Set $\mathcal{J} = \{i \in \mathcal{I} | \ |x_i^*| \geq \epsilon\}$. If $card(\mathcal{J}) \leq K$, then stop; x^* is also optimal for (2). Otherwise go to Step 2.
Step 2. Solve (2) with variables whose indices are in $\mathcal{J}$ and report its solution as approximate optimal solution of (1).

Remark 2.2. It is worth noting that one may apply cross-validation as in [5] to find appropriate bound UB. However, for large datasets it needs to solve several large-scale quadratic program which is time consuming.

Remark 2.3. It is worth noting that when short selling is not allowed in model (1), in the relaxed problem (3) when $UB \geq 1$ the constraint $||x||_1 \leq UB$ is redundant.

3 A Hybrid approach for the CCMCVaR model

The CCMCVaR model is as follows:

$$
\begin{aligned}
\min_{x,t,\gamma} \quad & \lambda\left(\gamma + \frac{1^T t}{(1-\alpha)m}\right) - (1-\lambda) r^T x \\
s.t. \quad & t_j \geq -\sum_{i=1}^{N} x_i y_i^j - \gamma, \quad j = 1, \ldots, m, \\
& 1^T x = 1, \\
& ||x||_0 <= K, \\
& \varepsilon \leq x \leq \delta, \\
& r_i x_i \geq 0, \quad i = 1, ..., N, \\
& t_j \geq 0, \quad j = 1, \ldots, m,
\end{aligned}
\tag{4}
$$

where m is the total number of scenarios which is usually a large number. Similar to the CCMV model, using 0-1 variables, the above problem can be rewritten as the following mixed-integer linear programming form:

$$
\begin{aligned}
\min_{x,t,\gamma} \quad & \lambda\left(\gamma + \frac{1^T t}{(1-\alpha)m}\right) - (1-\lambda) r^T x \\
s.t. \quad & t_j \geq -\sum_{i=1}^{N} x_i y_i^j - \gamma, \quad j = 1, \ldots, m, \\
& 1^T x = 1, \\
& 1^T z \leq K, \\
& \varepsilon_i z_i \leq x_i \leq \delta_i z_i, \quad i = 1, ..., N, \\
& r_i x_i \geq 0, \quad i = 1, ..., N, \\
& z_i \in \{0, 1\}, \quad i = 1, ..., N, \\
& t_j \geq 0, \quad j = 1, \ldots, m.
\end{aligned}
\tag{5}
$$

Solving model (5) for large datasets is challenging. For example, the authors in [8] proposed a penalty decomposition algorithm for Mean-Variance-CVaR model which is similar to (5). Here similar to Section 2, we consider the following relaxed version of (5):

$$\begin{aligned}
\min_{x,t,\gamma} \quad & \lambda\Big(\gamma + \frac{1^T t}{(1-\alpha)m}\Big) - (1-\lambda) r^T x \\
s.t. \quad & t_j \geq -\sum_{i=1}^{N} x_i y_i^j - \gamma, \quad j = 1, \dots, m, \\
& 1^T x = 1, \\
& ||x||_1 \leq UB, \\
& \varepsilon_i \leq x_i \leq \delta_i, \quad i = 1, ..., N, \\
& r_i x_i \geq 0, \quad i = 1, ..., N, \\
& t_j \geq 0, \quad j = 1, \dots, m,
\end{aligned} \tag{6}$$

where $UB = K \times \max_{i=1,\cdots,N}(|\varepsilon_i|, \delta_i)$. Model (6) is a linear program, however, in the case of large number of scenarios, the hybrid approach discussed in Section 2 might not be efficient. Thus we further apply the CSR technique to reduce the number of scenarios that is briefly illustrated in the next subsection.

3.1 The CSR technique

Denote the original set of scenarios as $\zeta^s = (\zeta_1^s, ..., \zeta_N^s)$ with their probability values $P_s = \dfrac{1}{m}, s = 1, ..., m$. For each $i \in N$, define the interval $I_i = [a_i, b_i]$, where

$$a_i = \min_s \{\zeta_i^s\}, \text{ and } b_i = \max_s \{\zeta_i^s\}.$$

Then each I_i is partitioned into C subintervals $I_e, e = 1, ..., C$ such that $I_i = \bigcup_e I_e$. After that all scenarios are classified into each subintervals I_e for $e = 1, ..., C$. Further, let c_{ie} be the cardinality of each S_{ie} (the set of scenarios into each subintervals I_e), then we determine the conditional scenario and its probability value as follows:

$$\begin{aligned}
\zeta_i^e &= \frac{1}{c_{ie}} \sum_{i \in S_{ie}} \zeta_i^s, \\
p_i^e &= \frac{s_{ie}}{m}.
\end{aligned}$$

Also, in the case of the empty classes, we delete them. We apply the CSR technique before and after relaxing the original model. The hybrid approach for solving CCMCVaR model can be outlined in Algorithm 2.

Algorithm 2

Input. ϵ, a small positive constant, K the desired number of assets and $\mathcal{I} = \{1, 2, \dots N\}$;
Step 2. Apply the CSR technique to CCMCVaR model, then solve (6) and let x^*

be the optimal solution of it. Set $\mathcal{J} = \{i \in \mathcal{I}|\ |x_i^*| \geq \epsilon\}$. If $card(\mathcal{J}) \leq K$, then stop; x^* is also optimal for (5). Otherwise go to Step 2.
Step 2. Solve (5) with variables whose indices are in $\mathcal{J}$ and reduced number of scenarios and report its solution as approximate optimal solution of (4).

4 Computational experiments

In this section, we compare Algorithms 1 and 2 with the original CCMV and CCMCVaR models from different perspective on 3 datasets of the S&P index when $\lambda = \frac{1}{2}$, $\delta_i = -\varepsilon_i = 0.2$, $\beta_i = 0.1\mu_i, i = 1, \cdots, N$ for different K values. Experiments are performed in MATLAB R2018a on a 2.50 GHz laptop with 4 GB of RAM, and CVX-MOSEK is used to solve all optimization models [7].

Computational results for the CCMV are reported in Tables 1-3, where we compare risks, returns, Sharpe ratios. Also, we report number of assets that are chosen both by the new approach and original model. In these tables, CCMV_1, denotes model (3) with $UB = K \times \max_{i=1,\cdots,N}(|\varepsilon_i|, \delta_i)$, and RCCMV_1 denotes model (2) with reduced number of 0-1 variables. Also, in these tables 'card(diff)' denote the number of assets for which $|x_i| > 0.001$ (number of assets different from the CCMV model).

As we see, Algorithm 1 is always faster than the case where we directly solve CCMV model for all assets. Also, risks, returns and Sharpe ratios are competitive with those of the CCMV model. From the row 'card(diff)' in Tables 1-3 we realize that the final assets chosen by the Algorithm 1 has around 10% difference with those chosen by the CCMV model. Thus Algorithm 1 is a significantly cheaper alternative to the CCMV model, specially for large number of assets. These results are also depicted in Figures 1-3.

We also have compared the case where short selling is not allowed in the CCMV model. As discussed before, in this case $||x|| \leq UB$ for $UB \geq 1$ becomes redundant and relaxed problem reduces to a convex quadratic program with simple linear and bounds constraints. The corresponding results for $K = 20$ are summarized in Table 4. As we see, Algorithm 1 again is a good alternative for directly solving the CCMV model. They have almost equal risks, returns and Sharpe ratios while the Algorithm 1 is faster and time difference for the largest dataset is significant.

In the sequel, we evaluate the performance of Algorithm 2 compared to the original CCMCVaR model with and without applying the CSR technique. We applied Geometric Brownian Motion (model in [1] to generate scenarios. The returns, risks, CVaR values, Sharpe ratios and CPU times are reported in Tables 5-7, for $m = 5000$ and different K values. The reduced number of scenarios is $\frac{m}{3}$. As we see, using CSR technique leads to improvement in CPU time but still directly solving the original model is time consuming. However, by applying Algorithm 2, we see significant reduction the CPU time while having competitive risks, returns and sharp ratios.

Table 1: Comparison of returns, risks and Sharpe ratios for different K values for 226 stocks of S&P index data when $\lambda = 0.5$.

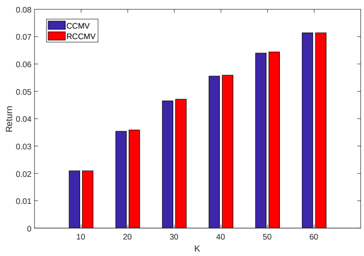
		$\beta = 0$			$\beta \neq 0$		
		$K = 20$	$K = 40$	$K = 60$	$K = 20$	$K = 40$	$K = 60$
CCMV	Return	0.0356	0.0555	0.0714	0.0317	0.0502	0.0642
	Risk	0.0039	0.0063	0.0086	0.0035	0.0066	0.0086
	Sharpe ratio	0.5688	0.6983	0.7690	0.5316	0.6153	0.6922
	CPU time	2.0007e+03	2.0007e+03	657.6238	2.0007e+03	2.0007e+03	846.2825
$CCMV_1$	Return	0.0368	0.0567	0.0719	0.0329	0.0507	0.0646
	Risk	0.0037	0.0063	0.0084	0.0036	0.0061	0.0083
	Sharpe ratio	0.6024	0.7118	0.7842	0.5516	0.6498	0.7095
	CPU time	1.8960	1.8885	2.0996	1.9153	2.0058	1.9899
	Card(diff)	29(10)	51(13)	68(10)	28(9)	53(15)	68(10)
$RCCMV_1$	Return	0.0359	0.0559	0.0714	0.0323	0.0503	0.0642
	Risk	0.0041	0.0065	0.0086	0.0040	0.0065	0.0086
	Sharpe ratio	0.5638	0.6924	0.7690	0.5085	0.6235	0.6922
	CPU time	3.6643	49.0523	13.2116	4.9382	10.5458	14.7442
	Card(diff)	20(2)	40(3)	60(0)	20(4)	40(4)	60(0)

Table 2: Comparison of returns, risks and Sharpe ratios for different K values for 476 stocks of S&P index data when $\lambda = 0.5$.

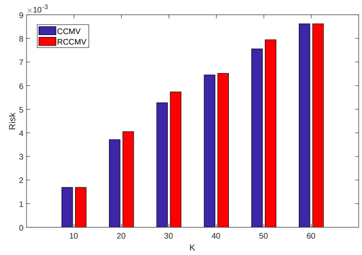
		$\beta = 0$			$\beta \neq 0$		
		$K = 20$	$K = 40$	$K = 60$	$K = 20$	$K = 40$	$K = 60$
CCMV	Return	0.0338	0.0512	0.0651	0.0302	0.0458	0.0578
	Risk	0.0038	0.0058	0.0078	0.0035	0.0057	0.0074
	Sharpe ratio	0.5491	0.6697	0.7368	0.5084	0.6087	0.6722
	CPU time	2.0010e+03	2.0010e+03	2.0009e+03	2.0009e+03	2.0010e+03	2.0004e+03
$CCMV_1$	Return	0.0346	0.0519	0.0656	0.0310	0.0465	0.0584
	Risk	0.0037	0.0059	0.0077	0.0035	0.0058	0.0074
	Sharpe ratio	0.5724	0.6767	0.7476	0.5242	0.6133	0.6785
	CPU time	4.0847	3.5166	2.3164	4.4711	5.0882	2.8021
	Card(diff)	29(10)	47(9)	74(14)	25(6)	48(8)	75(16)
$RCCMV_1$	Return	0.0338	0.0513	0.0649	0.0301	0.0458	0.0578
	Risk	0.0037	0.0059	0.0076	0.0035	0.0057	0.0074
	Sharpe ratio	0.5515	0.6657	0.7442	0.5103	0.6087	0.6723
	CPU time	8.7368	63.2774	100.7592	12.8560	101.3053	170.7512
	Card(diff)	20(2)	40(2)	60(10)	20(2)	40(0)	60(8)

Table 3: Comparison of returns, risks and Sharpe ratios for different K values for 2196 stocks of S&P index data when $\lambda = 0.5$.

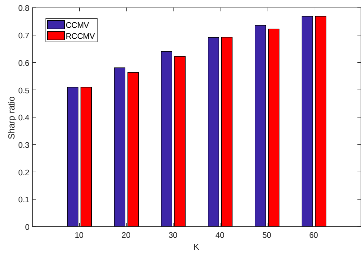
		$\beta = 0$			$\beta \neq 0$		
		$K = 20$	$K = 40$	$K = 60$	$K = 20$	$K = 40$	$K = 60$
CCMV	Return	0.0659	0.1115	0.1529	0.0592	0.1002	0.1285
	Risk	0.0100	0.0140	0.0182	0.0099	0.0140	0.0166
	Sharpe ratio	0.6596	0.9414	1.1339	0.5943	0.8486	0.9980
	CPU time	1.9968e+03	1.6926e+03	1.9969e+03	1.1871e+03	1.0844e+03	1.0672e+03
$CCMV_1$	Return	0.0669	0.1132	0.1539	0.0597	0.1008	0.1372
	Risk	0.0073	0.0115	0.0147	0.0068	0.0104	0.0134
	Sharpe ratio	0.7809	1.0562	1.2670	0.7223	0.9903	1.1860
	CPU time	26.4050	22.2193	39.7471	24.0941	26.4175	27.6225
	Card(diff)	26(7)	47(9)	74(16)	27(8)	48(8)	71(13)
$RCCMV_1$	Return	0.0649	0.1126	0.1522	0.0581	0.0991	0.1368
	Risk	0.0089	0.0147	0.0167	0.0086	0.0122	0.0166
	Sharpe ratio	0.6879	0.9298	1.1782	0.6270	0.8968	1.0634
	CPU time	227.9652	219.2356	221.1326	801.1458	103.8712	101.0022
	Card(diff)	20(2)	40(4)	60(5)	20(2)	40(4)	60(5)



(a) Return

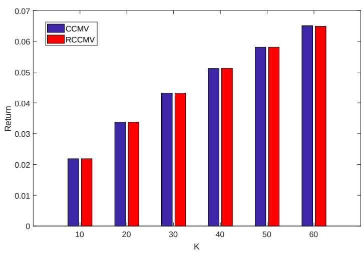


(b) Risk

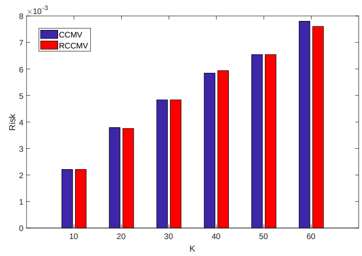


(c) Sharpe ratio

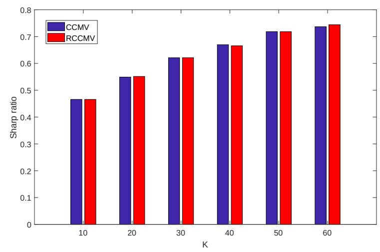
Figure 1: *Comparison of returns, risks and Sharpe ratios for different K values for 226 stocks of S&P index.*



(a) Return

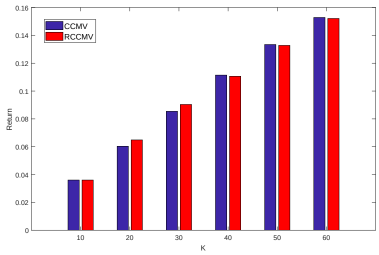


(b) Risk

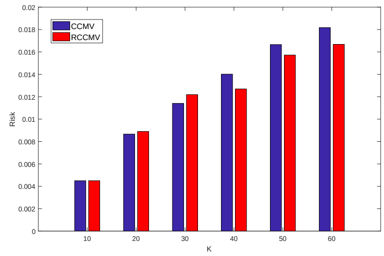


(c) Sharpe ratio

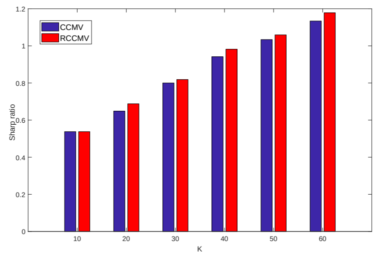
Figure 2: *Comparison of returns, risks and Sharpe ratios for different K values for 476 stocks of S&P index.*



(a) Return



(b) Risk



(c) Sharpe ratio

Figure 3: *Comparison of returns, risks and Sharpe ratios for different K values for 2196 stocks of S&P index.*

Table 4: Comparison of returns, risks and Sharpe ratios when $K = 20$, $\lambda = 0.5$ and $UB = K \times \max_{i=1,\cdots,N}(|\varepsilon_i|, \delta_i)$.

		$\beta = 0$		$\beta \neq 0$	
		CCMV	$RCCMV_1$	CCMV	$RCCMV_1$
N=226	Return	0.0130	0.0130	0.0117	0.0117
	Risk	0.0010	0.0010	0.0010	0.0010
	Sharpe ratio	0.4072	0.4072	0.3662	0.3668
	CPU Time	2.1544	1.6902	2.7311	2.0670
	Number of selected assets	6	6	5	6
N=476	Return	0.0160	0.0160	0.0144	0.0144
	Risk	0.0018	0.0018	0.0018	0.0018
	Sharpe ratio	0.3771	0.3772	0.3408	0.3408
	CPU Time	3.1575	2.3756	9.4438	2.8532
	Number of selected assets	6	6	6	6
N=2196	Return	0.0214	0.0214	0.0192	0.0192
	Risk	0.0022	0.0022	0.0021	0.0021
	Sharpe ratio	0.4553	0.4553	0.4192	0.4192
	CPU Time	186.5438	12.6267	317.9685	15.4123
	Number of selected assets	10	10	11	11

Table 5: Comparison of returns, risks and Sharpe ratios for different K values for 226 stocks of S&P index data when $\lambda = 0.5$.

		$K = 20$	$K = 40$	$K = 60$		$K = 20$	$K = 40$	$K = 60$
	Return	0.0174	0.0127	0.0201		0.0219	0.0217	0.0301
	Risk	0.0107	0.0020	0.0043		0.0102	0.0089	0.0109
CVaR	CVaR	-0.0155	-0.0786	-0.1627	CVaR-reduction	-0.0175	-0.0806	-0.1811
	Sharpe ratio	0.1682	0.2873	0.3056		0.2168	0.2300	0.2884
	CPU time	1.2829e+0	986.5185	612.2594		916.9856	439.9648	232.2425
	Return	0.0140	0.0118	0.0172		0.0201	0.0206	0.0309
	Risk	0.0056	0.0016	0.0029		0.0089	0.0076	0.0093
$CVaR_1$	CVaR	-0.0432	-0.1105	-0.1883	$CVaR_1$-reduction	-0.0950	-0.1632	-0.2430
	Sharpe ratio	0.1871	0.2950	0.3205		0.2131	0.2352	0.3204
	CPU time	106.7640	98.8680	92.9625		10.6319	9.7244	10.8735
	Card(diff)	93(75)	90(52)	74(36)		45(27)	64(29)	71(17)
	Return	0.0143	0.0136	0.0215		0.0188	0.0113	0.0198
	Risk	0.0078	0.0023	0.0049		0.0104	0.0024	0.0051
$RCVaR_1$	CVaR	-0.0192	-0.0762	-0.1643	$RCVaR_1$-reduction	-0.0831	-0.1543	-0.2328
	Sharpe ratio	0.1619	0.2836	0.3071		0.1940	0.2329	0.2633
	CPU time	333.0634	254.7662	201.4589		121.0670	114.6721	114.2365
	Card(diff)	20(7)	40(6)	60(7)		20(10)	40(9)	60(10)

Table 6: Comparison of returns, risks and Sharpe ratios for different K values for 476 stocks of S&P index data when $\lambda = 0.5$.

		$K = 20$	$K = 40$	$K = 60$		$K = 20$	$K = 40$	$K = 60$
	Return	0.0048	0.0078	0.0121		0.0177	0.0198	0.0188
	Risk	8.5697e-04	0.0017	0.0039		0.0059	0.0071	0.0079
CVaR	CVaR	-0.0257	-0.0814	-0.1725	CVaR-reduction	-0.0269	-0.0904	-0.1911
	Sharpe ratio	0.1651	0.1919	0.1945		0.1816	0.2350	0.2115
	CPU time	3.2884e+03	3.2901e+03	3.3003e+03		1.1354e+03	1.1652e+03	1.2043e+03
	Return	0.0062	0.0088	0.0108		0.0070	0.0113	0.0116
	Risk	5.4127e-04	0.0011	0.0024		9.6654e-04	0.0019	0.0030
CVaR_1	CVaR	-0.0537	-0.1073	-0.1956	CVaR_1-reduction	-0.0840	-0.1614	-0.2643
	Sharpe ratio	0.2664	0.2642	0.2187		0.2263	0.2620	0.2118
	CPU time	166.9591	163.5961	167.7118		16.5808	16.9371	16.6452
	Card(diff)	88(69)	92(54)	87(29)		54(34)	66(29)	71(16)
	Return	0.0057	0.0084	0.0126		0.0059	0.0109	0.0125
	Risk	0.0012	0.0019	0.0039		0.0012	0.0028	0.0038
RCVaR_1	CVaR	-0.0212	-0.0795	-0.1729	RCVaR_1-reduction	-0.0762	-0.1508	-0.2555
	Sharpe ratio	0.1643	0.1938	0.2018		0.1703	0.2061	0.2047
	CPU time	996.4651	911.5938	816.3695		411.2156	403.3264	366.3213
	Card(diff)	20(4)	40(7)	60(7)		20(6)	40(10)	60(11)

Table 7: Comparison of returns, risks and Sharpe ratios for different K values for 2196 stocks of S&P index data when $\lambda = 0.5$.

		$K = 20$	$K = 40$	$K = 60$		$K = 20$	$K = 40$	$K = 60$
	Return	0.0153	0.0160	0.0221		0.0216	0.0239	0.0274
	Risk	0.0028	0.0017	0.0042		0.0065	0.0077	0.0096
CVaR	CVaR	-0.0666	-0.0134	-0.1005	CVaR-reduction	-0.0813	-0.0246	-0.1985
	Sharpe ratio	0.2888	0.3823	0.3415		0.2679	0.2724	0.2797
	CPU time	3.1501e+03	3.1554e+03	3.1580e+03		1.4352e+03	1.210021e+03	1.0021e+03
	Return	0.0161	0.0108	0.0219		0.0161	0.0129	0.0201
	Risk	0.0019	6.7132e-04	0.0034		0.0027	0.0012	0.0037
CVaR_1	CVaR	-0.0794	-0.0391	-0.1156	CVaR_1-reduction	-0.1367	-0.0705	-0.1932
	Sharpe ratio	0.3694	0.4153	0.3765		0.3117	0.3724	0.3304
	CPU time	138.8997	140.2936	140.6240		12.3085	12.3938	12.4646
	Card(diff)	86(67)	96(58)	99(41)		45(27)	69(30)	88(31)
	Return	0.0155	0.0157	0.0218		0.0143	0.0153	0.0213
	Risk	0.0027	0.0017	0.0043		0.0030	0.0017	0.0046
RCVaR_1	CVaR	-0.0657	-0.0151	-0.1011	RCVaR_1-reduction	-0.1331	-0.0600	-0.1849
	Sharpe ratio	0.2983	0.3809	0.3324		0.2611	0.3685	0.3142
	CPU time	1.0816e+03	1.0783e+03	1.0881e+03		270.1568	262.1459	255.1256
	Card(diff)	20(2)	40(1)	60(8)		20(7)	40(5)	60(11)

5 Conclusions

In this paper we proposed a hybrid approach to solve two well-known problems in portfolio optimization, namely CCMV and CCMCVaR models. Computational experiments on three large datasets showed the hybrid approach is competitive with the direct approach in terms of risks, Sharpe ratios and returns while being significantly faster. Applying the proposed approach to other cardinality constrained models in portfolio optimization may be considered as future work [20].

Bibliography

[1] Agustini, W.F., Affianti, I.R., Putri, E.R. *Stock price prediction using geometric brownian motion.* Journal of Physics: Conference Series, 2018, 974, 012047.

[2] Bertsimas, D.; Shioda, R. *Algorithm for cardinality-constrained quadratic optimization.* Comput. Optim. Appl. 2009, 43, 1-22.

[3] Chang, T.J.; Meade, N.; Beasley, J.E.; Sharaiha, Y.M. *Heuristics for cardinality constrained portfolio optimisation.* Comput. Oper. Res. 2000, 27, 1271-1302.

[4] Cui, X.; Zheng, X.; Zhu, S.; Sun, X. *Convex relaxations and miqcqp reformulations for a class of cardinality-constrained portfolio selection problems.* J Glob Optim 2013, 56, 1409-1423.

[5] Dhingra, V.; Gupta, S.K.; Sharma, A. *Norm constrained minimum variance portfolios with short selling.* Comput. Manag. Sci. 2023, 20, 6.

[6] Fabozzi, F.J.; Kolm, P.N.; Pachamanova, D.A.; Focardi, S.M. *Robust portfolio optimization and management.* John Wiley Sons, 2007.

[7] Grant, M.; Boyd, S.; Ye, Y. *Cvx: Matlab software for disciplined convex programming, version 2.0 beta.* 2013.

[8] Hamdi, A.; Khodamoradi, T.; Salahi, M. *A penalty decomposition algorithm for the extended mean-variance-CVaR portfolio optimization problem.* Discret. Math. Algorithms Appl. 2024, 16, 3.

[9] Khodamoradi, T.; Najafi, A.R.; Salahi, M. *CCMV portfolio optimization with stocks and options using forecasted data.* Estud. Econ. Apl. 2021, 39, 10.

[10] Khodamoradi, T.; Salahi, M.; Najafi, A.R. *Portfolio optimization model with and without options under additional constraints.* Math. Probl. Eng. 2020, 1-10.

[11] Khodamoradi, T.; Salahi, M.; Najafi, A.R. *Cardinality-constrained portfolio optimization with short selling and risk-neutral interest rate.* Decis. Econ. Finance 2021, 1-18.

[12] Khodamoradi, T.; Salahi, M.; Najafi, A.R. *Multi-intervals robust mean-conditional value-at-risk portfolio optimisation with conditional scenario reduction technique.* Int. J. Appl. Decis. Sci. 2023, 16, 237-254.

[13] Kobayashi, K.; Takano, Y.; Nakata, K. *Bilevel cutting-plane algorithm for cardinality-constrained mean-CVaR portfolio optimization.* J Glob Optim 2021, 81, 493-528.

[14] Leung, M.F.; Wang, J. *Cardinality-constrained portfolio selection based on collaborative neurodynamic optimization.* J. Neural Netw. 2022, 145, 68-79.

[15] Leung, M.F.; Wang, J.; Che, H. *Cardinality-constrained portfolio selection via two-timescale duplex neurodynamic optimization.* J. Neural Netw. 2022, 153, 399-410.

[16] Li, D.; Ng, W.L. *Optimal dynamic portfolio selection: Multiperiod mean-variance formulation.* Math. Financ. 2000, 10, 387-406.

[17] Markowitz, H. *Portfolio selection.* J. Finance 1952, 7, 77-91.

[18] Patel, N.R.; Subrahmanyam, M.G. *A simple algorithm for optimal portfolio selection with fixed transaction costs.* J. Manag. Sci. 1982, 28, 303-314.

[19] Ramshe, M.; Gharakhani, M.; Feyz, A.; Sadjadi, S.J. *A firefly algorithm for portfolio optimization problem with cardinality constraint.* Int. J. Ind. Eng. 2021, 8, 24-33.

[20] Salahi, M.; Khodamoradi, T.; Hamdi, A. *Mean-standard deviation-conditional value-at-risk portfolio optimization.* The Journal of Mathematics and Modeling in Finance 2023, 3(1), 83-98.

[21] Shaw, D.X.; Liu, S.; Kopman, L. *Lagrangian relaxation procedure for cardinality-constrained portfolio optimization.* Optim Methods Softw. 2008, 23, 411-420.

[22] Woodside-Oriakhi, M.; Lucas, C.; Beasley, J.E. *Portfolio rebalancing with an investment horizon and transaction costs.* Omega 2013, 41, 406-420.

[23] Xie, J.; He, S.; Zhang, S. *Randomized portfolio selection with constraints.* PAC J OPTIM Journal 2008, 4, 89-112.

[24] Zheng, X.; Zhang, C.; Zhang, B. *A mayfly algorithm for cardinality constrained portfolio optimization.* Expert Syst. Appl. 2023, 120656.

[25] Angelelli, E.; Mansini, R.; Speranza, M.G. *A comparison of MAD and CVaR models with real features.* J. Banking Finance 2008, 32, 11881197.

[26] Beltran-Royo, C. *Fast scenario reduction by conditional scenarios in two-stage stochastic MILP problems.* Optimization Methods and Software 2019, 1-22.

[27] Cheng, R. and Gao, J. *On cardinality constrained mean-CVaR portfolio optimization.* The 27th Chinese Control and Decision Conference (2015 CCDC) 2015, 1074-1079.

[28] ZHANG, J.; LEUNG, T.; AND ARAVKIN, A. *A Relaxed Optimization Approach for Cardinality-Constrained Portfolio Optimization.* arXiv preprint arXiv:1810.10563 2018.

[29] ZHAO, Z.; WANG, H.; YANG, X; XU, F. *CVaR-cardinality enhanced indexation optimization with tunable short-selling constraints.* Applied Economics Letters 2021, 28, 201-207.

How to Cite: Maziar Salahi[1], Tahereh Khodamoradi[2], *An L_1 then L_0 approach to the cardinality constrained mean-variance and mean-CVaR portfolio optimization problems*, Journal of Mathematics and Modeling in Finance (JMMF), Vol. 4, No. 1, Pages:97–113, (2024).