Software Impacts 16 (2023) 100499

Contents lists available at ScienceDirect

Software Impacts

journal homepage: www.journals.elsevier.com/software-impacts

Original software publication

MOPO-LSI: An open-source multi-objective portfolio optimization library for sustainable investments

Yong Zheng [a,*], Kumar Neelotpal Shukla [b], Jasmine Xu [b], David (Xuejun) Wang [b], Michael O'Leary [b]

[a] *Illinois Institute of Technology, USA*
[b] *Morningstar, Inc., USA*

A R T I C L E I N F O

Keywords:
Multi-objective
Portfolio optimization
ESG
Sustainable investment

A B S T R A C T

Financial portfolio optimization is a process of selecting the optimal combination of assets to achieve a specific investment objective. Traditional portfolio optimization may only maximize returns and minimize risks, and ignore social responsibility or sustainability in financial investments. In this paper, we release MOPO-LSI which is a multi-objective portfolio optimization library for sustainable investments. More specifically, MOPO-LSI additionally considers Environmental, Social and Governance (ESG) factors as objectives to be optimized in financial portfolio, where investors' assets can be well allocated to mutual funds towards the improvements in sustainable development and practices. No matter client preferences on ESG factors are unknown or not, MOPO-LSI provides solutions to portfolio optimization in these scenarios. Moreover, MOPO-LSI is easy to be configured and used, and users can also extend its definition of multi-objective problems and adapt the solutions to customized requirements or applications.

Code metadata

Current code version	v1.0
Permanent link to code/repository used for this code version	https://github.com/SoftwareImpacts/SIMPAC-2023-117
Permanent link to Reproducible Capsule	https://codeocean.com/capsule/9097649/tree/v1
Legal Code License	MIT License
Code versioning system used	Git
Software code languages, tools, and services used	Python v3.9+, pymoo v0.6.0+, cvxpy v1.2.3+
Compilation requirements, operating environments & dependencies	https://github.com/irecsys/MOPO-LSI/
If available Link to developer documentation/manual	https://github.com/irecsys/MOPO-LSI/
Support email for questions	yzheng66@iit.edu

1. Introduction

Financial portfolio optimization [1,2] involves selecting the ideal mix of financial assets to achieve an investor's goals while considering their risk tolerance. Asset allocation is a crucial part of this process, as it involves dividing an investor's portfolio among various asset classes, such as stocks, bonds, and cash, based on their risk tolerance and investment goals. The goal of traditional portfolio optimization is to construct a portfolio that maximizes the expected return for a given level of risk or minimizes the risk for a given level of expected return.

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding author.
E-mail addresses: yzheng66@iit.edu (Y. Zheng), KumarNeelotpal.Shukla@morningstar.com (K.N. Shukla), Jasmine.Xu@morningstar.com (J. Xu), David.wang@morningstar.com (D. Wang), Michael.OLeary@morningstar.com (M. O'Leary).

https://doi.org/10.1016/j.simpa.2023.100499
Received 8 March 2023; Accepted 18 March 2023

In recent years, sustainable investments [3–5] have gained popularity, as more investors are recognizing the importance of considering Environmental, Social and Governance (ESG) [6,7] factors in investment decision-making. Sustainable investments, also known as socially responsible investing [3] or ESG investing [8], is an investment approach that considers not only financial returns but also the impact of the investment on the environment, society and corporate governance practices. This approach aims to generate long-term financial returns while promoting sustainable and responsible practices. The incorporation of ESG factors into investment decisions can help investors identify companies that are better positioned to manage environmental and social risks, as well as identify opportunities in companies that are leading the transition towards more sustainable practices. By investing in companies with strong ESG performance, investors can encourage and support these companies to continue to improve their sustainability practices over time.

Building an open-source library for sustainable investments is necessary because it allows for the democratization of sustainable investing, making it accessible to a wider range of investors and enabling the development of more effective and transparent investment strategies that take into account ESG factors. Additionally, an open-source library can provide a platform for collaboration and knowledge-sharing among investors, academics, and industry professionals, leading to more informed and impactful investment decisions. In this paper, we introduce and release MOPO-LSI which is a multi-objective portfolio optimization library for sustainable investments. MOPO-LSI can offer solutions (i.e., optimized portfolio for investing in mutual funds) for two scenarios: one in which the client's preferences for ESG factors are known in advance, and the other in which they are currently unknown. Users can also easily extend the definition of multi-objective optimization (MOO) problems in MOPO-LSI, and adapt the optimization to their own customized optimization objectives.

2. MOPO-LSI

The MOPO-LSI library works for asset allocations to a list of mutual funds, where the ESG scores associated with each mutual fund are available in the data set. The ESG factors are categories into positive and negative ones. The positive ESG factors refer to the ones that are considered beneficial for society, such as clean energy, well-beings, people's health, and so forth. The negative ESG variables are consider harmful, e.g., environmental pollution, regulatory violations, human rights abuses, etc.

MOPO-LSI is able to produce portfolio solutions in the following two scenarios:

- *When client preferences on ESG factors are known*: In this case, these preferences can be considered as weights assigned to each ESG variable. The optimization problem is transformed to a single-objective optimization task by using scalarization approaches [9]. More specifically, we use the weighted sum method as the scalarization method to optimize three objectives — maximizing positive ESG (i.e., weighed average of positive ESG scores in the portfolio), minimizing negative ESG (i.e., weighted average of negative ESG cores in the portfolio), and minimizing tracking error of the portfolio. The tracking error [10] is considered as a joint factor of risks, returns and volatility. It is calculated as the standard deviation of the difference between the returns of the portfolio and the benchmark associated with a specific risk level (e.g., conservative, moderate or aggressive). In MOPO-LSI, we adopted the embedded conic solver (ECOS) [11] in the CVXPY [12] library to solve the quadratic convex optimization problem in this scenario.
- *When client preferences on ESG factors are unknown*: We utilized several multi-objective evolutionary algorithms (MOEAs) [13] from the Pymoo [14] library to optimize specific objectives, including maximizing the positive ESG score, minimizing the negative ESG score and tracking error. Since client preferences are unknown, these positive and negative ESG scores are calculated based simple averages. In addition, we allow clients to specific individual positive and/or negative ESG factors, so that we can additionally maximize and/or minimize them. MOEAs will provide Pareto set which is a set of non-dominated solutions. MOPO-LSI allows users to give extra inputs in order to utilize multi-criteria decision making methods [15] to select a single optimal solution from the Pareto set.

In addition to the objectives mentioned above, there also several constraints defined in the MOO problem, such as the equality constraint where the summation of asset allocations must be equal to 100%, inequality constraint that guarantees the diversity of asset allocations (e.g., no more than 20% invested to a single mutual fund), and so forth.

3. How to use

The workflow by MOPO-LSI can be depicted by Fig. 1, with descriptions as follows.

- *Configurations*. At the beginning, users should well-define hyperparameters in different configuration files, e.g., yaml files. Regarding these parameters, refer to the user guide on the coding repository on Github.
- *Run MOPO-LSI*. A user can start running the library by using "python run.py" or "python run.py –config your.yaml". The default yaml to start running is the user.yaml file.
- *Load system configurations*. The library will first load system configurations defined by yaml files in the folder "yaml", including system.yaml, scalarization.yaml, moea.yaml. The file "system.yaml" defines file paths, output path, risk levels (i.e., conservative, moderate, aggressive), ESG groups and dimensions, the optimization model to be used, as well as the general parameters for system constraints. The files, "scalarization.yaml" and "moea.yaml", define the MOO problems and corresponding hyperparameters related to the optimizers.
- *Load user configurations*. The file, "user.yaml", is loaded, where the library will read user inputs, such as user option on the risk level, user preferences, and so on.
- *Load data*. The library will then load the data sets indicated in "system.yaml", including the list of funds with their ESG scores, the covariance matrix and standard asset allocations for each risk level.
- *Initialize optimizer*. The library initializes the optimizer by using the parameter "model" defined in "system.yaml". The MOO problem defined in corresponding yaml, either "scalarization.yaml" or "moea.yaml", will also be set up in this stage. A MOO problem defines the list of objectives and constraints to be considered in the optimization process.
- *Run optimization*. The library will run optimization by using corresponding MOO algorithms (e.g., NSGA2) or optimization solvers (e.g., ECOS for the weighted sum method).
- *Get optimization results*. The scalarization methods can return a single optimal solution directly, while the MOEAs will return a Pareto set which is a set of non-dominated solutions. There are several selection methods built in the library to help select a single optimal solution from the Pareto set.

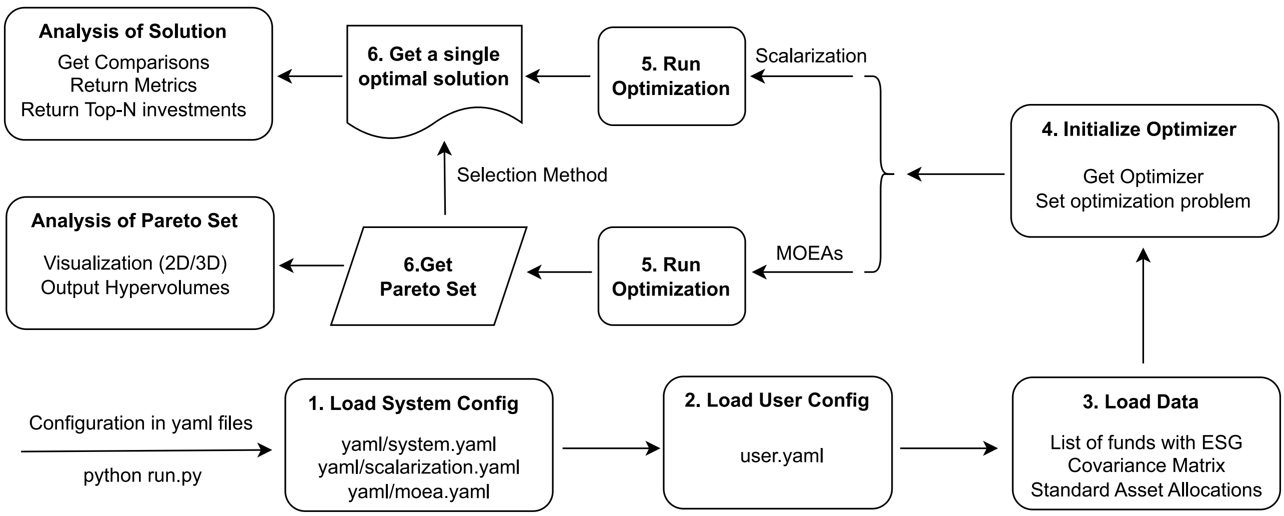

Fig. 1. The framework and workflow in MOPO-LSI.

```
08 Mar 10:14    SUCCESS  Time cost: 4.08 seconds. Optimization is done, with tracking error of 0.00601
08 Mar 10:14    INFO  Start results analysis...

08 Mar 10:14    SUCCESS  Gains on PosESG: 78.96%
08 Mar 10:14    INFO  Details on PosESG:
                    Solution  Benchmark    Ratio  Preference
clean_energy       11.271718   7.809283   44.34%        0.23
energy_efficient   11.678043   4.500237  159.50%        0.21
recycle_material   12.920997   8.555833   51.02%        0.10
wellbeing           2.045654   2.051497   -0.28%        0.04
human_development   8.242344   7.905945    4.26%        0.04

08 Mar 10:14    SUCCESS  Gains on NegESG: 54.41%
08 Mar 10:14    INFO  Details on NegESG:
                 Solution  Benchmark    Ratio  Preference
carbon           3.204334   3.877207   17.35%        0.23
thermal_coal     6.342649   6.805105    6.80%        0.23
palm_oil         0.323982   2.563921   87.36%        0.21
ocean_pollution  0.438748   0.787276   44.27%        0.21
tobacco          0.000000   2.494546  100.00%        0.05
alcohol          0.433342   2.706351   83.99%        0.05
drugs            1.016219   3.380177   69.94%        0.05
arms             0.145263   2.924516   95.03%        0.32
military         0.000000   3.363846  100.00%        0.32
war_support      6.535923   6.529733   -0.09%        0.32

08 Mar 10:14    SUCCESS  Top Investments (up to 10):
08 Mar 10:14    INFO
     WeightedSum       secid                                      name
135     0.168128  F00000PGG5          JOHCM Global Select Institutional
112     0.164752  F00000Y6I5                     Ecofin Global Water ESG
25      0.093759  F000000N4Y              TIAA-CREF Core Impact Bond Instl
46      0.093352  F00000MIQX                RBC BlueBay High Yield Bond I
74      0.081898  FOUSA00ECC                       Parnassus Fixed-Income
92      0.080866  F00000X1BL                 iShares ESG Aware MSCI EM ETF
69      0.074293  F000000D01           Brown Advisory Sustainable Growth I
162     0.064059  FOUSA00E4Y                     Neuberger Berman Genesis Inv
188     0.040665  FOUSA00DNP   Green Century Equity Individual Investor
110     0.036579  F00000N2G9                   Boston Trust Walden SMID Cap
```

Fig. 2. Example of the outputs by MOPO-LSI.

- *Analysis of the solutions*. By given a single optimal solution, the library can compare it with the standard benchmark (i.e., a non-ESG optimized solution associated with a specific risk level), output evaluation metrics (i.e., tracking error, improvement ratio on each ESG factor), and finally return the top-N mutual funds for investments. An example of the outputs by using the sample data and the weighted sum method in MOPO-LSI can be observed from Fig. 2, where the gain values refer to the weighted average of the improvement ratios over each ESG groups (i.e., positive and negative ESGs). In terms of the analysis for Pareto set, the library has the option to visualize the non-dominated solutions and hypervolumes.

Note that a sample data is provided in the folder "data/SampleData/". This data was produced by simulations. The list of ESG factors and the ESG scores associated with each mutual fund are generated and simulated for the purpose of running tests only.

4. Impacts

The library has several benefits and impacts that can be summarized as follows.

- We believe that MOPO-LSI is the first library to support sustainable investments in mutual funds. Existing ones either did not consider ESG factors, or are targeted for general investments, e.g., stocks or bonds.
- MOPO-LSI supports MOO solutions for portfolio optimization in two scenarios — run scalarization when user preferences are known, and run MOEAs when user preferences are unknown.
- MOPO-LSI provides enriched outputs, including standard metrics (e.g., tracking error), metrics in comparisons (e.g. gains or improvement ratios) and the top-N mutual funds for investments, where the transparency and user trusts can be improved by using these outputs as explanations.
- The potential visualizations can further enhance transparency. The library currently supports the visualization of hypervolumes and non-dominated solutions, as well as the selected optimal solution by a specific multi-criteria decision making method.
- MOPO-LSI allows users to extend the MOO problems (i.e., create subclasses of the pre-defined MOO problems, including the Python class *ScalarProblem* and *MOEAProblem*) and re-define the list of objectives and constraints, while users can run the library again without any changes to the optimizer or optimization process.
- MOPO-LSI, finally, can not only assist investors and fund managers to start investing practical by considering ESG factors, but also encourage the sustainable investments in a long run.

5. Limitations & future work

At the current stage, we only implemented limited solutions of scalarization approaches and MOEA methods. More optimizers will be added and implemented in our future work.

CRediT authorship contribution statement

Yong Zheng: Conceptualization, Methodology, Software, Validation, Resources, Writing. **Kumar Neelotpal Shukla:** Conceptualization, Validation, Resources. **Jasmine Xu:** Conceptualization, Validation, Resources. **David (Xuejun) Wang:** Conceptualization. **Michael O'Leary:** Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported and funded by Morningstar, Inc. at Chicago, Illinois, USA, under grant agreement IIT-Cayuse: Grant No. 22-0300, Project No. A23-0011.

References

[1] A. Ponsich, A.L. Jaimes, C.A.C. Coello, A survey on multiobjective evolutionary algorithms for the solution of the portfolio optimization problem and other finance and economics applications, IEEE Trans. Evol. Comput. 17 (3) (2012) 321–344.
[2] A. Gunjan, S. Bhattacharyya, A brief review of portfolio optimization techniques, Artif. Intell. Rev. (2022) 1–40.
[3] L. Renneboog, J. Ter Horst, C. Zhang, Socially responsible investments: Institutional aspects, performance, and investor behavior, J. Bank. Financ. 32 (9) (2008) 1723–1742.
[4] T. Busch, R. Bauer, M. Orlitzky, Sustainable development and financial markets: Old paths and new avenues, Bus. Soc. 55 (3) (2016) 303–329.
[5] G. Talan, G.D. Sharma, Doing well by doing good: A systematic review and research agenda for sustainable investment, Sustainability 11 (2) (2019) 353.
[6] G. Friede, T. Busch, A. Bassen, ESG and financial performance: Aggregated evidence from more than 2000 empirical studies, J. Sustain. Finance Invest. 5 (4) (2015) 210–233.
[7] A. Amel-Zadeh, G. Serafeim, Why and how investors use ESG information: Evidence from a global survey, Financ. Anal. J. 74 (3) (2018) 87–103.
[8] G. Halbritter, G. Dorfleitner, The wages of social responsibility—Where are they? A critical review of ESG investing, Rev. Financ. Econ. 26 (2015) 25–35.
[9] G. Eichfelder, Adaptive Scalarization Methods in Multiobjective Optimization, Springer, 2008.
[10] P. Jorion, Portfolio optimization with tracking-error constraints, Financ. Anal. J. 59 (5) (2003) 70–82.
[11] A. Domahidi, E. Chu, S. Boyd, ECOS: An SOCP solver for embedded systems, in: 2013 European Control Conference, ECC, IEEE, 2013, pp. 3071–3076.
[12] S. Diamond, S. Boyd, CVXPY: A Python-embedded modeling language for convex optimization, J. Mach. Learn. Res. 17 (1) (2016) 2909–2913.
[13] C.C. Coello, Evolutionary multi-objective optimization: A historical view of the field, IEEE Comput. Intell. Mag. 1 (1) (2006) 28–36.
[14] J. Blank, K. Deb, Pymoo: Multi-objective optimization in Python, IEEE Access 8 (2020) 89497–89509.
[15] S. Greco, J. Figueira, M. Ehrgott, Multiple Criteria Decision Analysis, Vol. 37, Springer, 2016.