arXiv:2304.06675v1 [cs.LG] 13 Apr 2023

约束多目标投资组合优化问题的学习启发式方法

SONIA BULLAH, 威特沃特斯兰德大学计算机科学与应用数学系，南非TERENCE L. VAN ZYL,约翰内斯堡大学智能系统研究所，南非

多目标投资组合优化是各个研究领域都关注的重要问题，其目标是在最大化预期收益的同时，最小化给定投资组合的风险。然而，许多研究未将现实约束纳入模型，从而限制了实际交易策略的应用。本研究将交易成本和持有成本等现实约束纳入优化模型。由于该问题具有非凸性质，NSGA-II、R-NSGA-II、NSGA-III 和 U-NSGA-III 等元启发式算法将在求解过程中发挥重要作用。此外，本研究采用学习启发式方法，通过代理模型增强所使用的元启发式算法。随后，我们将这些算法与基线元启发式算法进行比较；后者在不使用学习启发式方法的情况下求解约束多目标优化问题。研究结果表明，尽管学习启发式算法完成运行所需的时间显著更长，但在超体积和收敛速度方面优于基准算法。此外，回测结果表明，利用学习启发式方法生成资产配置权重，与不使用学习启发式方法进行回测相比，可降低风险百分比、提高预期收益并提升夏普比率。因此，我们得出结论：使用学习启发式方法求解约束多目标投资组合优化问题，所得结果优于且更值得采用的做法是不使用学习启发式方法求解该问题。

附加关键词和短语：多目标投资组合优化、学习启发式、元启发式、遗传算法、代理辅助算法、机器学习、深度学习、约束

ACM 参考格式：
Sonia Bullah and Terence L. van Zyl. 2023. 用于受约束多目标投资组合优化问题的学习启发式方法。1, 1（2023年4月），12页。
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 引言

本研究通过纳入众多因素，探讨一个多目标投资组合优化问题，同时努力在产生高于平均水平的收益率的情况下，将投资者的总体风险降至最低 [15]。

通过在优化模型中引入现实约束，并结合机器学习、深度学习 [10] 和元启发式算法来解决非凸问题。本研究探讨的元启发式算法为进化算法（EA），具体包括 NSGA-II、R-NSGA-II、NSGA-III 和 U-NSGA-III。相比之下，用于增强这些元启发式算法的机器学习模型，将采用所用各元启发式算法的代理辅助变体 [16, 20]。

在 Ma等人 [12], 的研究中，研究人员将机器学习和深度学习模型与投资组合优化模型进行比较，以确定最佳投资策略组合。最终得出的结论是，随机森林模型 + 结合预测的均值-方差（MVF）模型能够实现最高效率，且

作者地址：Sonia Bullah，2107762@students.wits.ac.za，威特沃特斯兰德大学计算机科学与应用数学系，南非约翰内斯堡；Terence L. van Zyl，约翰内斯堡大学智能系统研究所，南非约翰内斯堡，tvanzyl@gmail.com。

允许免费为个人或课堂使用而制作本作品全部或部分内容的数字或纸质副本，但不得为营利或商业利益而制作或分发副本，且副本必须附有本声明和首页上的完整引文。对于本作品中由 ACM 以外的其他方拥有版权的组成部分，必须尊重其版权。注明出处后允许摘录。以其他方式复制、再版、发布到服务器或分发至列表，均须事先获得具体许可和/或支付费用。请通过 permissions@acm.org 申请许可。© 2023年 计算机协会。XXXX-XXXX/2023/4-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

为实际投资带来有利结果。然而，唯一的问题是，高换手率给这些模型带来了巨大的挑战。高换手率意味着交易成本或交易费用（这两个术语可以互换使用）会相当高。由于这些成本在现实交易情境中确实存在，因此在分析投资组合优化问题时不应忽视它们。因此，本研究将把这些因素及其他现实因素作为投资组合优化问题中的约束。可以采用混合方法加入这些额外约束，即将元启发式与机器学习和深度学习技术结合起来。这种方法可称为“学习启发式” [7]。

总体而言，本研究将重点探讨如何将现实约束（如交易成本和持有成本）引入投资组合优化问题，从而在现实交易场景中充分考虑这些约束的存在；同时，还将研究如何利用学习启发式来协助解决具有约束、多目标和非凸特征的投资组合优化问题。通过引入一种独特的混合方法，本研究将解决当前投资组合优化问题中交易场景不切实际的问题，同时在使投资者受益的情况下，相较于其他方法取得更优的结果。

2 背景

2.1 投资组合优化

投资组合优化构成了本研究的基础，也是许多投资者面临的一个基本问题[17]。投资组合优化过程包括从所有可能的投资组合中选择最优投资组合，以在最大化投资者财富的同时最小化风险。实现这一目标的一种常见方法是使用夏普比率，该比率可以直接进行优化 [23]。

马克维茨的投资组合选择理论是投资组合选择问题的一项重要贡献[13]。马克维茨提出了一个模型，假设投资者旨在针对特定风险水平最大化收益率，反之亦然。许多其他研究在马克维茨理论的基础上纳入了约束。Kizys 等人[9] 指出，增加更多约束会显著提升该模型的复杂性。然而，可以使用元启发式算法来克服这些障碍。

Doering 等人 [7] 通过根据马克维茨的均值-方差理论构建无约束投资组合优化问题，扩展了 Streichert 等人 [21] 的研究，具体形式如下，旨在使风险最小化：

$$Min \sum_{i=1}^{N} \sum_{j=1}^{N} \omega_i \omega_j \sigma_{ij}$$

并以下述方式最大化利润或收益率：

$$Max \sum_{i=1}^{N} \omega_i \mu_i$$

满足以下条件：

$$\sum_{i=1}^{N} \omega_i = 1, \quad where\ 0 \leq \omega_i \leq 1, \forall i = 1, 2, ...N,$$

其中，N 表示可用资产的数量，μ_i 表示资产 i的预期收益，σ_{ij} 表示资产 i 与 j之间的协方差，ω_i 表示投资组合的决策变量。本研究将调整这类模型，以纳入交易成本等现实约束。

此外，投资组合优化可归类为多目标优化问题 [22] ，其目标是找到能够体现不同目标之间权衡关系的解 [17]。多目标投资组合优化问题需要同时满足两个目标 [21]。

机器学习和深度学习模型可以与传统优化模型相结合 [12] ，以找到适用于实际投资的最佳组合。下文将探讨这些思路。

2.2 用于多目标投资组合优化的机器学习模型和深度学习模型Zhang 等 [23] 主要关注如何利用深度学习模型评估投资组合优化问题：首先给出一个目标函数，其中定义了夏普比率，并通过梯度上升法使其最大化。

Ma 等 [12] 使用了常见的机器学习模型，如支持向量回归（SVR）和随机森林（RF），以及深度学习模型，如深度多层感知器（DMLP）、LSTM 神经网络和卷积神经网络（CNN）。此外，他们将每个模型与一种传统的投资组合优化模型配对，即均值-方差模型和 Omega 模型。

除了使用机器学习和深度学习模型外，许多研究还引入了混合算法。Kizys 等 [9], 的研究便是一个例子：他们采用了一种新算法 *ARPO*，以考虑不可避免的现实约束，而许多研究者都不愿分析这些约束。类似地，van Zyl 等 [22] 采用 *ParDen* ，纳入生成式或判别式机器学习模型，并将其用作代理模型，以降低回测过程中产生的成本。

由于问题本身的计算开销很大，优化问题依赖于耗时函数 [14]。实现这类函数会影响算法的计算开销，而算法在每次调用函数时都会用于寻找新解。因此，可以加入代理模型，以加快收敛速度，并近似当前用于解决这些优化问题的模型和算法中的耗时函数。

代理辅助优化能够高效地进行近似解评估（ASE），并利用这些评估结果减少代价高昂的解评估（ESE），同时将近似误差降至最低。总体而言，所采用的优化算法的收敛性将得到改善 [14]。

2.3 交易成本和持仓成本

交易成本源于换手率；换手率越高，交易成本就越高。这意味着，在考虑和不考虑相关交易成本[12]的情况下，模型的表现会有所不同。Ma 等人 [12] 扩展了 [7, 9, 11, 22] 和 [1], ，其中简要讨论了额外成本的概念，并将交易成本视为现实投资组合优化问题中的一项主要约束。

同样，在实际投资和交易场景中，持有成本也值得考虑。最简单的持有成本模型会计算建立空头头寸时借入资产所产生的费用 [5]。然而，在某些条件下，这些模型会变得复杂得多。与交易成本类似，持有成本通常为非负值，但出现负值也并非不可能。

如 van Zyl 等人 [22] 所做的那样，可以将交易成本和持有成本结合起来，并加入杠杆约束，从而构建一个多目标优化问题。如下所示：

$$\text{Maximise:} \quad w'\mu - \frac{\gamma}{2} w' \sum w - \gamma_t \phi^{trade}(w - w_0) - \gamma_h \phi^{hold}(w)$$

$$\text{满足：} \|w\|_1 \leq L^{max}$$

其中， w_0 表示初始投资组合， $w - w_0$ 表示为确保 w所需进行的交易， ϕ^{trade} 是交易成本函数，其权衡参数为 γ_t； $\phi^{hold}(w)$是交易成本函数，其权衡参数为 γ_h；最后， L^{max} 表示最大杠杆。

可以注　　意到，维持多头头寸的全额投资组合，其杠杆为　　1.

2.4 引入投资组合约束

大多数投资组合优化问题都会考虑某种形式的风险管理，而许多研究论文并未充分考虑随机优化问题，即在优化模型中使用约束[7]。

Abate 等人 [1] 认为应考虑投资组合约束，因为仅考虑均值-方差优化会导致不公平的资产配置。他们还实施了一系列受约束策略，包括经典约束、灵活的投资组合约束、基于范数的约束、基于方差的约束、跟踪误差波动率约束以及 beta 约束。他们指出，更严格的约束可以形成多元化的投资组合，而选择恰当的约束在评估过程中至关重要。投资者通常应选择与其目标相符的约束。

Kizys 等人 [9] 拓展了 Doering 等人 [7]和 Abate 等人 [1], 的研究；这两篇文章都考虑了投资者在现实投资场景中无法忽视的约束。

选择约束时，交易成本发挥着重要作用。受约束的投资组合优化模型有利于投资者，可视为经典投资方法的一种有效替代方案[7]。

2.5 用于求解优化问题的元启发式算法

元启发式算法能够为非凸投资组合优化问题提供解决方案。通常，只要问题是凸的，单期优化问题（SOP）就能得到高效求解[5]。然而，一旦将其调整为带有非凸约束的多目标优化问题（MOP），问题就会变成 NP 困难问题，求解也更加复杂。在这种情况下，元启发式发挥着至关重要的作用，因为它们可用于求解 NP 困难优化问题并提供出色的解决方案 [7], ，同时考虑问题中引入的现实非凸约束。

Doering 等人 [7] 将元启发式分为两类：受自然启发的元启发式和非自然启发的元启发式。本研究将主要关注受自然启发的元启发式的一个子类——进化算法。具体而言，研究将聚焦于遗传算法（GA）。不过，本研究关注多目标问题，因此必须纳入多目标元启发式。因此，本文将探讨遗传算法的不同形式，即 NSGA-II、R-NSGA-II、NSGA-III 和 U-NSGA-III [22],。

NSGA-II 是一种基于非支配关系的遗传算法，常用于求解多目标优化问题 [19]。首先，算法初始化种群，并考虑可能指定的任何约束。接下来，对种群进行非支配排序。然后，根据每个个体所属的前沿，为其赋予等级或适应度值。此外，还会为种群中的每个个体分配拥挤距离，以量化该个体与其邻近个体的接近程度。接下来，使用二元锦标赛选择从种群中选出父代，并由选出的种群生成子代。随后，再次使用非支配排序程序对相关种群和子代进行排序，并只保留最优个体。

R-NSGA-II算法是对NSGA-II算法的扩展，改进了生存选择过程[22]。在该算法中，父代与参考点之间的欧氏距离越小越受青睐 [8]。算法以与NSGA-II相同的方式开始，从而得到一个合并后的种群。随后，通过非支配排序将合并后的种群划分为多个非支配层级。然后，对种群的每个前沿进行聚类，以确保解具有多样性。接着，将整个种群缩减至与父代种群相同的规模 [1] [8]。该算法与NSGA-II的区别在于，划分前沿时依据的是解的等级或适应度值 [22]。

[1]在未……的情况下 在选出的 l 个成员中，那些成员本身与参照对象之间的欧氏距离较小的成员 ce
点将被保留 ed.

不同于NSGA-II 基于等级或适应度以及拥挤距离进行选择，NSGA-III 使用参考方向来确保解之间的多样性 [18]。NSGA-III算法从一个随机种群开始，并在超平面上分散参考点，旨在使每个参考点对应一个种群成员。种群被划分为多个非支配层，然后执行非支配排序，就像NSGA-II算法所做的那样。之后，从中产生一个子代种群，并由此形成一个合并种群。接下来，迭代地选择点，直到达到这样一个点：无法选择某个完整前沿中的所有解。最终前沿将无法被完全选择。然而，在这种情况下，只需从该前沿中选择少量解。然后，根据每个成员的最小垂直距离，将每个种群成员和最终前沿的成员与一个参考点关联起来。接下来，实施一种生态位策略，以获得与种群边缘化参考点相对应的最终前沿成员。与这些点相关联的种群成员将被优先选择。该算法试图找到与每个参考点相对应的一个种群成员，使其接近帕累托最优前沿 [18]。

对于多目标优化问题，将通过非支配排序过程把种群划分为多个非支配前沿。U-NSGA-III 使用小生境锦标赛选择过程，因为其性能优于随机选择 [22]。该过程更侧重于非支配解而非受支配解，并更倾向于靠近参考方向的解而非较远的解。之后，算法遵循与 NSGA-II 算法相同的过程 [18]。

2.6 学习启发式方法

‘学习启发式’ 是一种将元启发式算法与机器学习或统计学习模型相结合的混合方法 [7]。本研究将采用学习启发式方法，为四种多目标遗传算法分别配对一个代理模型。随后将使用代理辅助算法评估投资组合优化问题。

Calvet 等 [6] 解释说，机器学习与元启发式算法的混合可以分为两类：一类是利用机器学习增强元启发式算法，另一类是利用元启发式算法改进所使用的机器学习模型或技术。

使用机器学习增强元启发式算法，可分为特定位置的混合方法和全局层面的混合方法。特定位置的混合方法包括微调参数、快速初始化、低成本评估技术、种群管理、在算子中引入知识，以及最后将机器学习技术用作局部搜索。另一方面，全局混合方法侧重于缩减搜索空间、算法选择、使用超启发式（即用于选择启发式方法以求解计算搜索问题的搜索方法）、协作策略，最后还包括采用新型元启发式算法。

反过来，利用元启发式算法改进机器学习技术，涉及分类、回归、聚类和规则挖掘等主题。

采用学习启发式方法意味着上述两类方法中的一类或两类都需要实现。表 1 显示，尽管相关研究要么认可、要么实现了考虑某种形式现实约束的方法，但只有 Doering等人 [7] 提出的综述聚焦于使用元启发式算法解决多目标优化问题。因此，本研究将结合相关研究中概述的技术，并尝试通过使用机器学习技术增强元启发式算法，从而取得更好的结果。

表 1. 与本研究相关工作的总结。

作者	优化	约束	算法	优势
Ma等人	单目标-&多目标-目标	With con-约束	机器学习与深度学习	为 NP困难优化问题提供解决方案；生成高质量的解。
Zhang 等人	单一-目标	With con-约束	深度学习	性能提升；该方法合理且实用。
Doering等	单一目标	With and无约束	元启发式算法	产生较高的换手率。

3 方法论

为了检验针对多目标投资组合优化问题的学习启发式方法能否在降低投资风险的同时带来更高收益，我们构建了一种混合学习启发式算法，并将结果与基线算法进行比较。以往关于多目标投资组合优化的研究，例如 Ma 等人 [12], 开展的研究，都未考虑现实约束，例如交易成本。因此，在尝试同时最大化收益率和最小化风险的过程中，我们的研究将采用一种学习启发式方法纳入这些约束，其方法与 Calvet 等人 [6]的研究中所述的流程类似。

本研究的实施可分为五个相互依赖的主要部分。下文将详细介绍各部分，并在图1中进行总结。

3.1 数据清理与探索

我们对标准普尔500指数数据进行了预处理，以确保数据类型正确并删除冗余属性。同时也删除了重复项、缺失值和空值。我们仅选取预期收益排名前十的股票作为投资组合，以开展本研究的后续工作。

3.2 预测

为了预测数据，需要添加某种形式的噪声。这种噪声代表误差，会使预测值增加或减少，因此难以对数据做出准确预测。我们采用均值为0、标准差为0.1的正态分布来获取随机样本。由此生成一个包含噪声的数组，并将其添加到数据中，从而得到预测数据值。最后，在进行投资组合优化之前，需要删除所有空值。

3.3 遗传算法

在本研究中，需要使用四种遗传算法来满足两个主要目标，即优化投资组合并同时找出最优的约束值；其次，生成可用于执行回测的最佳权重集合。所有遗传算法均使用*Pymoo* [3]库实现，并且所采用的投资组合优化方法取决于*PyPortfolioOpt* [2]。

首先，调整目标函数，以便对数据进行投资组合优化。具体做法是获取预测数据集的预期收益和样本协方差。预期收益使用资本资产定价模型计算

$$R_i = R_f + \beta_i(E(R_m) - R_f),$$

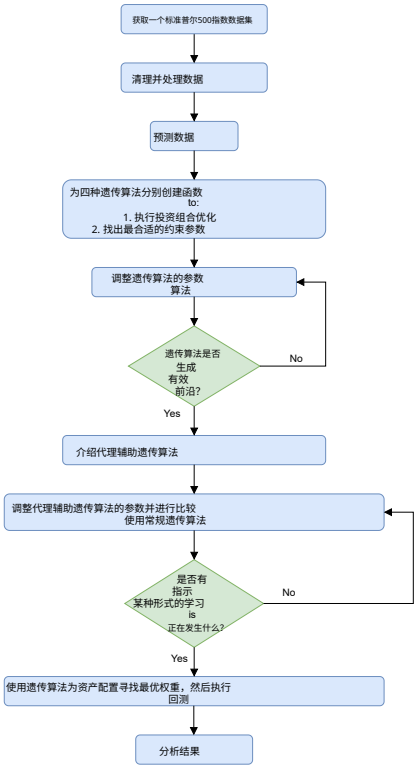


图1. 本研究所采用的方法论总结。

其中， R_f表示无风险利率， β_i是资产 i的 beta（或敏感度），而 $(E(R_m) - R_f)$ 表示市场风险溢价。此外，样本协方差使用 Ledoit-Wolf 协方差收缩估计量计算

$$\hat{\Sigma} = \alpha F + (1 - \alpha)S,$$

其中， α 表示收缩常数， F 表示结构化估计量， S 表示样本协方差矩阵。

这样一来，只要正确绘制结果，我们就能找到最优投资组合并绘制有效前沿。此外，需要注意的是，该方法求解的是无约束投资组合。然而，本研究的重点是有约束投资组合。因此，我们在优化问题中加入一项约束。

绘制出有效前沿后，我们接着为其添加一项约束。我们不会随意添加一个取值随机的约束，而是寻找最合适的参数。找到该参数后，夏普比率将达到最大值，并得出整体投资组合表现。

遗传算法通过随机生成初始种群并为每个个体分配适应度分数来运行。随后，个体经过选择和繁殖阶段，然后进行交叉和变异，将新的遗传信息添加到子代种群中。最后进行代际替换，之后算法根据停止条件终止。

每个遗传算法返回所获得的投资组合表现结果数组后，我们会调整该数组，从每个子数组中移除夏普比率值，因为我们不需要三维图。这样，每个子数组便只包含两个值，即最优投资组合的收益率和波动率（或风险）。随后将这些结果绘制成图。

此过程在种群规模为 12 的情况下运行 10 次，共 30 代。我们将四种遗传算法绘制在同一坐标轴上，以比较结果。为了比较各算法的表现，我们记录每种算法运行至完成所需的平均时间，以及算法产生的平均超体积。首先，每当算法启动时，我们就启动计时器，并在算法完成最后一代后停止计时。最后返回各算法运行所需的平均时间（秒）。每种算法运行后，我们会使用*Pymoo* [3] 库中的超体积函数。超体积是一种性能指标，它利用参考点计算解集相对于特定参考点所影响的面积 [3]。需要注意的是，超体积越大，表明算法表现越好。获得每种算法各代对应的超体积后，我们计算十次运行的平均超体积，并比较各算法，以确定它们在整个代数范围内的表现。

在目标函数中，我们还在有效前沿中加入一个目标，用于调整伽马值。需要更多不可忽略的权重时，可以增大这一 L2正则化参数。遗传算法会搜索伽马的最佳取值，进而生成用于回测的最优权重集合。下一节将详细阐述这一过程。与第一个案例一样，找到每只股票对应的权重后，我们便对投资组合进行优化，并求出最大夏普比率。随后，将优化后投资组合中每只股票的最终权重加入一个数组，以供回测过程使用。

3.4 学习启发式：代理辅助遗传算法

遗传算法运行后，我们会调整各个函数，使其包含代理模型。为此，我们调用*Pysamoo* [4] 框架中的 GPSAF 实现。此外，我们还会手动微调*alpha*、*beta*、*n_max_doe*、*n_max_infills*、*n_offsprings* 和*seed* 参数，以使每种算法达到最佳收敛速率。我们会持续调整这些参数，直到代理辅助遗传算法在迭代过程中持续进行“学习”。最终，与非代理辅助遗传算法相比，我们得到了更优的解。同样，这些学习启发式算法会在种群规模为 12、运行 30 代的条件下各运行十次，并在结束时记录平均耗时和平均超体积。我们将每种遗传算法与其代理辅助版本绘制在同一坐标轴上进行比较，并记录算法运行至完成时的收敛速率、平均超体积和平均耗时差异。

3.5 回测

我们通过运行模拟或回测来帮助优化策略。为此，我们调用 [2] 库中的回测函数*portfolio-backtest* 。将遗传算法得到的权重代入该函数，并输入目标收益率、目标方差、起始时期和结束时期。我们首先使用该函数内置的方法确定每项资产的权重并运行回测，然后使用四种代理辅助遗传算法分别搜索最优权重集，再将其传入回测函数。最后，我们记录每次回测输出的数据，即年化波动率、预期年化收益率和夏普比率，并比较结果。

所有算法和实现均在运行 macOS 12.3 的四核 Intel Core i5 上运行。完整代码、数据集和结果可在以下 GitHub代码库中找到。

[2]文档见：https://pypi.org/project/portfolio-backtest/

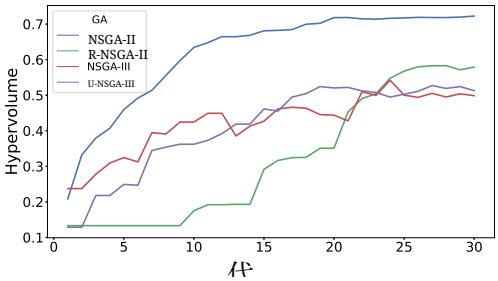
4 结果与讨论

本节将介绍用于优化投资组合的四种多目标遗传算法的结果，并探讨其局限性和未来工作。

4.1 结果

根据下方图2(a)所呈现的结果，可以明显看出，NSGA-III算法运行至完成所需的时间最短，其次是U-NSGA-III算法，然后是 R-NSGA-II算法，而 NSGA-II算法的运行时间最长。然而，鉴于 NSGA-II算法产生的超体积最高，其次依次为 R-NSGA-II、U-NSGA-III，最后是 NSGA-III算法，这表明表现最佳的算法是 NSGA-II。

算法	代数	种群	HV	时间（秒）
NSGA-II	30	12	0.7225	1.0115
R-NSGA-II	30	12	0.5790	0.7941
NSGA-III	30	12	0.4986	0.7557
U-NSGA-III	30	12	0.5126	0.7681



(a) 每种算法以种群规模为 12运行 30代数所需的平均超体积和平均时间（以秒为单位）。（b）种群规模为 12 时，四种遗传算法在 30 代数中的比较。

图 2. 不同遗传算法的比较。

根据图 2(b)，可以看出 NSGA-II 算法最先收敛，并生成更平滑的折线图。其余三种算法各自的折线图中包含更多尖峰，但总体结果表明，随着代数增加，超体积也随之增大。

接下来，我们将介绍四种学习启发式算法（或代理辅助遗传算法）的结果。图3(a)表明，代理辅助R-NSGA-III算法以最短时间运行完毕，其次是U-NSGA-III算法，然后是NSGA-III算法，而NSGA-II算法的运行时间最长。然而，与非代理变体类似，NSGA-II算法获得了最高的超体积，因此使其成为效率最高的算法。

下面的图4(a)、图4(b)、图4(c)和图4(d)均表明，学习启发式算法所产生的超体积高于基线算法。通过比较图2(a)和图3(a)，我们观察到，尽管学习启发式算法需要显著更长的运行时间才能完成，但它们产生的超体积高得多，收敛速度更快，同时还能生成更加平滑的图。如Blank 等 [2] 所述，学习启发式算法加速了用于求解优化问题的多目标元启发式算法的收敛。

最后，我们对数据进行模拟或回测，并记录由十项资产构成的切点投资组合所产生的年度波动率、预期年度收益率和夏普比率。根据图3(b)，我们看到，四种学习启发式算法各自得到的切点投资组合均优于未采用学习启发式算法的回测结果。利用代理辅助遗传算法寻找资产配置最优权重的投资组合，在较低风险比例下获得更高收益，并产生更高的夏普比率。

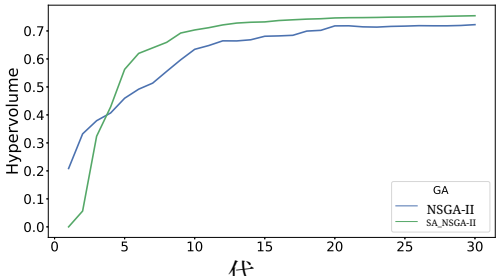
算法	代数	种群	HV	时间（秒）
SA_NSGA-II	30	12	0.7543	18.0166
SA_R-NSGA-II	30	12	0.7181	15.2820
SA_NSGA-III	30	12	0.6403	16.6078
SA_U-NSGA-III	30	12	0.7093	16.4171

(a) 各代理模型辅助元启发式算法以种群规模 12 运行 30 代时的平均超体积和平均耗时（秒）。

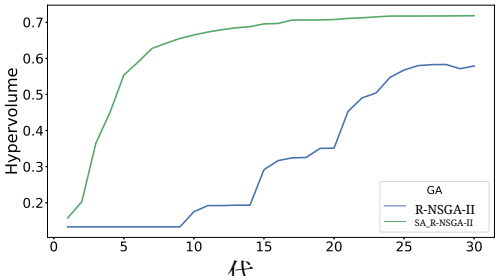
算法	波动率（%）	收益率 (%)	SR	时间（秒）
没有遗传算法	22.4	26.6	1.10	-
SA_NSGA-II	18.2	55.6	2.94	40.7051
SA_R-NSGA-II	18.2	55.6	2.94	43.0664
SA_NSGA-III	18.2	55.6	2.94	29.5306
模拟退火 U-NSGA-III_	18.2	55.6	2.94	28.7683

(b)由回测生成的切点投资组合的年化波动率、预期年化收益率和夏普比率（SR），以及每种算法运行所需的时间（以秒计）。

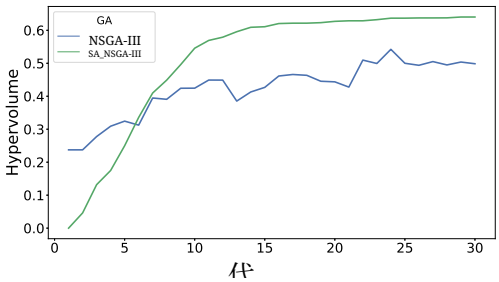
图 3. 不同代理辅助元启发式算法的比较。



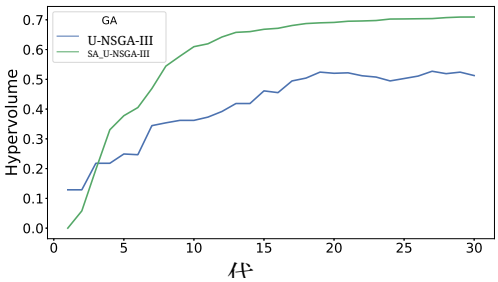
(a) NSGA-II 和代理辅助 NSGA-II 算法的比较。



(b) R-NSGA-II算法与代理辅助R-NSGA-II算法的比较。



(c) NSGA-III 与代理辅助 NSGA-III 算法的比较。



（d）U-NSGA-III 算法与代理辅助 U-NSGA-III 算法的比较。

图 4. 不同算法的比较。

在这些情况下，与未使用遗传算法寻找最优资产权重的回测相比，预期年度收益率和夏普比率均提高了一倍以上。在 Zhang 等人 [23],的研究中，深度学习模型评估投资组合优化问题之前，目标函数已将夏普比率最大化。同样，在本研究中，我们通过最大化目标函数中的夏普比率来求解优化问题，随后由各个元启发式算法调用该目标函数。

尽管四种学习启发式算法均能得出最优结果，但图 3(b) 显示，代理辅助 U-NSGA-III 算法完成运行所需时间最短，其次依次为 NSGA-III、NSGA-II 和 R-NSGA-II。因此，对于回测而言，代理辅助 U-NSGA-III 算法是效率最高的学习启发式算法。总体而言，显然，与不采用学习启发式方法求解投资组合优化问题相比，学习启发式算法能够得出更优结果。

4.2 局限性

尽管有上述发现，仍需注意，学习启发式算法对特定参数较为敏感，例如*n_后代*、*alpha*、*beta*、*n_max_doe*、*n_max_填充点*和随机种子参数。若其中任一参数略有偏差，算法可能会过早终止。此外，算法未能考虑大型投资组合，因为增加投资组合中的股票数量会加剧参数敏感性。因此，本研究仅考虑由十项资产构成的投资组合所产生的结果。

4.3 未来工作

4.3.1 实施搜索算法。 由于所研究算法的参数相当敏感，因此可以利用网格搜索或随机搜索等搜索算法，找到最适合输入算法的参数。

4.3.2 扩大投资组合规模。 获得稳定的参数后，未来的研究可以分析规模大得多的投资组合，例如完整的标普 500 指数，而非仅分析本研究所考虑的前十项。

4.3.3 应用其他多目标元启发式算法。 未来研究可以在本研究的基础上，采用不同的多目标元启发式算法来解决本研究确定的同一问题。可考虑的算法包括粒子群优化（PSO）、差分进化（DE）或进化策略（ES）。

5 结论

投资组合优化问题必须在考虑实际交易和投资场景中不可避免的现实约束的前提下求解。本研究采用一种称为学习启发式的混合方法，将交易成本等约束纳入多目标投资组合优化问题。该方法运用代理模型等机器学习技术，改进用于求解非凸优化问题的元启发式算法。

在实现四种学习启发式算法后，显而易见，NSGA-II 学习启发式算法的表现最佳，因为其超体积高于实现的另外三种学习启发式算法。此外，尽管学习启发式算法需要更长时间才能运行完成，但每种学习启发式算法的超体积都超过了相应非代理变体（即我们的基准算法）的超体积。

根据研究结果，我们可以得出结论：在超体积和收敛速率方面，学习启发式算法优于基线算法。不过，它们需要更长时间才能运行完成。总体而言，在回测中使用学习启发式方法获取各资产的权重，与不使用学习启发式方法进行回测相比，可以降低年度风险、提高年度预期收益和夏普比率。因此，可以得出结论：在求解受约束的多目标投资组合优化问题时，学习启发式方法能够产生更优、更理想的结果。

本研究还 o 存在一些局限性。首先，学习启发式算法对许多输入 ut 参数敏感，而且 由于我们的目标是找到最优解，因此确定准确的参数至关重要。其次，th e

本研究的实施仅针对由十只股票组成的投资组合进行，因为向投资组合中增加更多股票会因学习启发式参数的敏感性而无法得到最优解。因此，在未来的研究中，实施搜索算法（例如随机搜索或网格搜索）来调节参数将会有所裨益。反过来，这也可能有助于使学习启发式算法能够考虑包含十项以上资产的投资组合，并且仍然产生最优结果。此外，未来的研究还可以考虑其他多目标元启发式算法。

参考文献

[1] Guido Abate、Tommaso Bonafini和Pierpaolo Ferrari。2022年。《投资组合约束：实证分析》。《国际金融研究》10，1（2022年），9。[2] J. Blank 和 K. Deb。2020年。《Python中的多目标优化：pymoo》。《IEEEAccess》8（2020年），89497–89509。

[3] 2022 2204 05855 [ ]

[4] 2022

Julian Blank和 Kalyanmoy Deb。。《Python中的代理辅助多目标优化：pysamoo》。arXiv：。cs.NEJulian Blank和 Kalyanmoy Deb。。《Python中的代理辅助多目标优化：pysamoo》。arXiv预印本 arXiv:2204.05855（2022年）。[5] StephenBoyd、Enzo Busseti、Steve Diamond、Ronald NKahn、Kwangmoo Koh、Peter Nystrup、JanSpeth等。2017年。通过凸优化进行多期交易。《优化基础与趋势》3，1（2017年），1–76。[6] Laura Calvet、Jésica de Armas、DavidMasip和 Angel A Juan。2017年。学习启发式：将元启发式与机器学习相结合，以处理动态输入下的优化问题。《开放数学》15，1（2017年），261–280。[7] Jana Doering、Renatas Kizys、AngelA Juan、Angels Fito 和Onur Polat。2019年。复杂投资组合优化与风险管理的元启发式：当前状况与未来趋势。《运筹学研究展望》6（2019年），100121。[8] Ernestas Filatovas、Algirdas Lančinskas、OlgaKurasova 和Julius Žilinskas。2017年。带随机局部搜索的基于偏好的多目标进化算法 R-NSGA-II。《中欧运筹学杂志》25，4（2017年），859–878。[9] Renatas Kizys、Angel A Juan、Bartosz Sawik和LauraCalvet。2019年。用于复杂投资组合优化的偏置随机化迭代局部搜索算法。《应用科学》9，17（2019年），3509。[10] Siddeeq Laher、Andrew Paskaramoorthy 和Terence L VanZyl。2021年。用于金融时间序列预测融合和最优投资组合再平衡的深度学习。载于《2021年IEEE第24届信息融合国际会议》（FUSION）。IEEE，1–8。[11] Bin Li和 StevenCH Hoi。2014年。在线投资组合选择：综述。《ACM计算综述》（CSUR）46，3（2014年），1–36。[12] Yilin Ma、Ruizhu Han 和 Weizhong Wang。2021年。利用深度学习和机器学习进行收益预测的投资组合优化。《应用专家系统》165（2021年），113973。[13] HM Markowitz。[n.d.]。《投资组合选择》（1952年）7J。[14] Robert Andrew Martin。2021年。PyPortfolioOpt：Python中的投资组合优化。《开放源代码软件期刊》6，61（2021年），3066。https://doi.org/ 10 21105 03066 。/joss。Andrew B Paskaramoorthy、Tim JGebbie和TerenceLvanZyl。2020年。在线投资决策框架。《投资分析师期刊》49，3（2020年），215–231。[16] RylanPerumal和 Terence L vanZyl。2020年。基于智能体的模型参数化的代理辅助方法。载于《2020年第7届软计算与机器智能国际会议》（ISCMI）。[17] BY Qu、Q Zhou、JM Xiao、JJ Liang和 PN Suganthan。2017年。使用多目标进化算法和预选方法进行大规模投资组合优化。《工程数学问题》2017（2017年）。[18] Haitham Seada 和 KalyanmoyDeb。2014年。U-NSGA-III：用于单目标、多目标和多目标优化的统一进化算法。COIN报告 2014022（2014年）。[19] Aravind Seshadri。2006年。一种快速的精英多目标遗传算法：NSGA-II。MATLAB Central 182（2006年）。[20] L Stander、M Woolway和TL van Zyl。2020年。扩展代理辅助连续过程优化。载于《2020年第7届软计算与机器智能国际会议》（ISCMI）。IEEE，275–279。[21] Felix Streichert、HolgerUlmer 和 AndreasZell。2004年。进化算法与基数约束投资组合优化问题。载于《2003年运筹学会议论文集》。Springer，253–260。[22] Terence L van Zyl、MatthewWoolway 和 Andrew Paskaramoorthy。2021年。Parden：用于投资组合选择的代理辅助超参数优化。载于《2021年第8届软计算与机器智能国际会议》（ISCMI）。IEEE，101–107。[23] Zihao Zhang、Stefan Zohren 和Stephen Roberts。2020年。用于投资组合优化的深度学习。《金融数据科学期刊》2，4（2020年），8–20。

[15]