Early portfolio pruning: a scalable approach to hybrid portfolio selection

Knowledge and Information Systems
https://doi.org/10.1007/s10115-023-01832-7

REGULAR PAPER

Early portfolio pruning: a scalable approach to hybrid portfolio selection

Daniele G. Gioia[1] · **Jacopo Fior**[2] · **Luca Cagliero**[2]

Received: 5 October 2022 / Revised: 14 December 2022 / Accepted: 7 January 2023

Abstract
Driving the decisions of stock market investors is among the most challenging financial research problems. Markowitz's approach to portfolio selection models stock profitability and risk level through a mean–variance model, which involves estimating a very large number of parameters. In addition to requiring considerable computational effort, this raises serious concerns about the reliability of the model in real-world scenarios. This paper presents a hybrid approach that combines itemset extraction with portfolio selection. We propose to adapt Markowitz's model logic to deal with sets of candidate portfolios rather than with single stocks. We overcome some of the known issues of the Markovitz model as follows: (i) *Complexity*: we reduce the model complexity, in terms of parameter estimation, by studying the interactions among stocks within a shortlist of candidate stock portfolios previously selected by an itemset mining algorithm. (ii) *Portfolio-level constraints*: we not only perform stock-level selection, but also support the enforcement of arbitrary constraints at the portfolio level, including the properties of diversification and the fundamental indicators. (iii) *Usability*: we simplify the decision-maker's work by proposing a decision support system that enables flexible use of domain knowledge and human-in-the-loop feedback. The experimental results, achieved on the US stock market, confirm the proposed approach's flexibility, effectiveness, and scalability.

Keywords Portfolio selection · Early portfolio pruning · Artificial intelligence · Decision support systems · Parallel itemset mining

Daniele G. Gioia, Jacopo Fior, Luca Cagliero have contributed equally to this work.

✉ Jacopo Fior
jacopo.fior@polito.it

Daniele G. Gioia
daniele.gioia@polito.it

Luca Cagliero
luca.cagliero@polito.it

[1] Department of Mathematical Sciences, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy

[2] Department of Control and Computer Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy

Published online: 31 January 2023

1 Introduction

Stock portfolio selection aims at allocating funds to financial equities. The pioneering work by [1] presents the popular mean–variance model to address portfolio optimization. In a nutshell, stock return and risk of investment are quantified using first- and second-order moments of per-stock historical price distributions. Although lots of efforts have been performed by researchers to solve and expand Markowitz's model, scalability issues [2] and reliability of the estimated values still need to be faced. In fact, the information required by [1] to estimate expected value and higher-order moments super-linearly scales with the candidate stocks [3]. More specifically, the Markowitz approach concerns the estimation of covariance values and the cardinality of the model parameters is quadratic with the number of considered assets. Estimating a so large number of model parameters, beyond requiring a considerable computational effort, raises serious questions on the reliability of these values [4]. Furthermore, investors commonly need to enforce additional, more complex constraints, e.g., incorporating transaction costs and sector-based stock diversification strategies [5]. Many of them require the adoption of heuristic methods as the new problem becomes NP-hard. For this reason, the most common portfolio selection approaches apply heuristic methods to shortlist the most relevant stocks [6–8].

The recent advances in data mining and machine learning techniques have fostered the development of *hybrid solutions* to portfolio optimization. They consist of a two-step process, where a subset of the most relevant stocks is selected first based on data-driven models and then an optimization step is applied on top of the shortlisted stocks [9]. The techniques used for stock selection at the first stage encompass, among other, machine learning and deep learning models [10], clustering techniques [11], and swarm intelligence and other metaheuristics [12]. To the best of our knowledge, all previous hybrid methods select portfolios on top of a shortlist of individual stocks. This limits the efficiency and scalability of the optimization step. To efficiently and effectively integrate complex portfolio-level constraints deeply into the portfolio selection step, it would be desirable to early prune part of the portfolio candidates at this previous stage.

The present paper proposes a scalable hybrid method, namely *early portfolio pruning* (EPP), where a set of candidate portfolios is early generated at the first step by means of an itemset-based heuristic. Then, the selection problem is no longer solved on top of a set of *single stocks*, but rather on a *portfolio shortlist*. In other words, the main analytical complexity is moved up to the itemset-based heuristic phase and accomplished by means of ad hoc, scalable algorithms. The name assigned to the presented method (EPP) emphasizes its peculiar characteristic to early discard the less interesting portfolios from the search space as soon as possible. To overcome the limitations of the Markowitz approach, we quantify the interactions among stocks only within a restricted number of portfolios previously shortlisted by an itemset mining algorithm, thus reducing the model complexity. Moreover, we provide decision-makers with a configurable system that does not forcibly depend on the first two moments, which are often contested in the financial world because of their instability and subject to great variation, thus allowing the incorporation of other metrics.

The proposed hybrid portfolio generation method allows investors to customize the selection process at their complete discretion even while coping with a large set of stocks. To this aim, we first reformulate the optimization problem by [1] to tailor it to the modified task. Then, we propose a scalable implementation of the EPP method integrating a parallel, scalable implementation of an itemset mining algorithm [13]. Finally, we integrate the presented

method into a financial decision support system (DSS), which leverages fundamental data analysis to choose the portfolio according to the end-users' preferences.

We run both performance evaluations and scalability tests on stocks belonging to the US market. The outcomes of the backtesting simulations confirm the effectiveness of EPP compared to previous approaches: EPP produces, on average, a good combination of profitable yet low volatile portfolios, also in adverse market conditions (e.g., during the COVID-19 pandemic). Furthermore, the scalable implementation allows EPP to handle stock sets not manageable by existing itemset-based heuristics, e.g., [14].

The key contributions of the paper are enumerated below:

- We present EPP, a hybrid method to select stock portfolios where candidate portfolios are early pruned by means of an itemset mining process.
- We propose an adapted version of the established mean–variance logic proposed by [1]. The proposed approach partly overcomes the inherent complexity of the traditional Markowitz model due to the excessively large number of estimated parameters as well as supports the enforcement of portfolio-level constraints on real features other than the stock prices [15].
- We present a decision support system allowing experts to monitor the portfolio selection process and leverage domain knowledge by enforcing the portfolio-level constraints (see Fig. 1).
- We adopt a parallel itemset mining implementation to perform candidate portfolio generation in a scalable way. We also empirically demonstrate the scalability of the proposed approach with the number of analyzed stocks and the size of the training time window.
- We compare EPP performance with that of both existing methods and real US funds in terms of portfolio payout (profit measure) and volatility (risk measure), showing competitive results.

The rest of the paper is organized as follows. Section 2 reviews the state of the art. Sections 3 and 4, respectively, formalize the traditional and adapted mean–variance model. Section 5 describes the architecture of the decision support system. Section 6 reports the main experimental results, whereas Sect. 7 draws conclusions and discusses future works.

2 Literature review

Stock portfolio optimization aims at allocating funds to a set of selected equities [16]. The traditional mean–variance model proposed by [1] focuses on finding the best trade-off between return of investment and risk by, respectively, quantifying them as the mean and variance of the historical stock prices. Several extensions of the original model have been proposed in the literature. They propose, for example, integrating more advanced risk measurements, to handle a maximum number of selected stocks, and to incorporate transaction costs in the optimization model [15]. Another commonly used approach based on the traditional mean–variance model is the capital asset pricing model (CAPM). It is still widely used for portfolio construction, although it is often criticized for its poor empirical performance and strong simplifications, which invalidate its application use [17]. In its classic Sharpe–Lintner version, the expected return on a given asset is constructed through the risk-free interest rate plus a risk premium (market beta of the asset) multiplied by the premium per unit of beta risk.

Rather than finding the portfolio that best matches a set of restrictive conditions, portfolio optimizers can be integrated into financial decision support systems [18–20]. The aim is to

allow end-users to specify their personal preferences, targets, and attitude to risk. This work presents a decision support system integrating a hybrid strategy to portfolio selection.

The research community has paid a particular attention to properly handle efficiency and estimation issues. In fact, linear and quadratic mixed-integer programming solvers may encounter issues while coping with a large number of securities. Moreover, allowing end-users to personalize the portfolio selection commonly entails enforcing ad hoc constraints, which may further increase the complexity of the optimization problem [21]. To find computationally effective solutions to NP-hard problems, the most common strategy is to adopt heuristic approaches to shortlist the candidate stocks [6–8]. Similar to [6–8], this paper addresses the selection of a subset of profitable stocks to buy, while introducing additional ad hoc constraints on the candidate portfolio.

More recently, researchers have tried to combine optimization strategies with data mining and machine learning techniques with the goal of heuristically choosing the most convenient stocks to buy. Specifically, hybrid approaches apply machine learning techniques to forecast future stock prices and then shortlist the stocks with a higher expected return to create the portfolio. For example, [22] and [3] rely on an investment decision model that predicts the direction of the stock prices first. Next, only those stocks designed to reach the expected return are considered eligible for the Markowitz optimization model. Similarly, [23] apply a genetic algorithm to select good quality assets at the first stage. This paper proposes a hybrid approach that combines established data mining techniques, i.e., itemset mining, with optimization techniques. Unlike [3, 22, 23], the proposed approach is fully unsupervised.

Other hybrid approaches rely on a two-stage process that performs stock evaluation and scoring. For example, [10] first evaluate each individual stock by performing a prediction of the stock return in the next time period. Next, they compute a scoring function that takes into account fundamental factors such as the net profit margin and the cash flow ratio. Alternative stock evaluation and scoring strategies encompass the use of clustering to find groups of similar stocks [11], genetic algorithms [24] or swarm intelligence methodologies [25–27] to deal with portfolio optimization, and itemset mining to generate candidate portfolios satisfying global constraints on the expected portfolio returns [14]. [14] perform a greedy selection of the candidate portfolios based on a set of a user-specified constraints related to portfolio size and diversification level. Unlike [10, 11] we early perform not only single stock selection but also global portfolio evaluation based on a parallel itemset mining approach. Unlike [14], on top of the itemset mining phase we shortlist the best candidate portfolio using an adapted Markowitz logic that incorporates a variety of additional constraints (including those based on fundamental analysis). Furthermore, we adopt a parallel implementation of the itemset mining process to scale toward large sets of stocks.

3 Problem statement

3.1 Notation

Hereafter, we will adopt the notation reported in Table 1.

3.2 The Mean–Variance model

The original mean–variance (MV) model is among the most established stock portfolio optimization strategies [1]. The key idea is to deal with the return of a single stock as a

Table 1 Summary of notations and their meanings

F	Financial statements of the candidate stocks
S	Set of candidate stocks
$\mathbb{P}$	Power set of S that represents all the possible portfolios
P_q	Stock portfolio consisting of a selection of candidate stocks, indexed by $q \in \{1, \ldots, \lvert\mathbb{P}\rvert\}$
H	Historical price series of the candidate stocks within the reference time period
w_i	Proportion of the total amount available for investment applied to stock $s_i \in S$
x_q	Binary vector in $\{0, 1\}^{\lvert S\rvert}$ with 1 when the stock s_i is selected in portfolio P_q and 0 otherwise
$E[\cdot]$	Expected value function
$E_{min}[\cdot]$	Lower-Bound estimate of the portfolio return (LBPR)
R_i	Random variable return of stock $s_i \in S$ over the holding period
$\mu_i = E[R_i]$	Expected return of the individual stock s_i over the holding period
$\mu^{\mathbb{P}} : \mathbb{P} \rightarrow \mathbb{N}^+$	Generalized expected return of the portfolio P_q over the holding period
$\boldsymbol{\mu}$	Vector with the expected return of all the stock in S
$\sigma_{ij} = Cov(s_i, s_j)$	Covariance of the returns for the pair of stocks s_i and s_j
Σ	Covariance in matrix form for all the stock in S
$\Sigma^{\mathbb{P}} : \mathbb{P} \rightarrow \mathbb{N}^+$	Generalized risk measure for the portfolio P_q over the holding period
$c^{\mathbb{P}} : \mathbb{P} \rightarrow \mathbb{N}^+$	Technical and fundamental analysis constraint function for the portfolio P_q over the holding period
$\mathbf{1}, \mathbf{0}$	Vectors with all elements set to 1 and 0

random variable and to consider expected return and variance to model stock profitability and risk level, respectively. To quantify the return of investment and the risk level of each individual stock, the distribution descriptors are computed over the historical stock prices H [28].

According to the MV model, the return of a candidate stock s_i is modeled as a random variable R_i, with associated expected return $\mu_i = E(R_i)$. By identifying the vector of these latter values as $\boldsymbol{\mu}$, the expected portfolio return is formulated as follows,

$$\boldsymbol{\mu}^{\mathsf{T}}\mathbf{w} = \sum_{i=1}^{|S|} w_i \mu_i . \tag{1}$$

Where the participation weights of the candidate stocks are stored into vector $\mathbf{w} \in \mathbb{R}^{|S|}$, denoting by w_i, i=1,2,. . .,$|S|$ the weight of stock $s_i \in S$ in portfolio P.

Beyond maximizing the expected return of the selected portfolio, the MV model incorporates portfolio diversification by estimating the return dispersion as

$$\mathbf{w}^{\mathsf{T}}\Sigma\mathbf{w} = \sum_{i=1}^{|S|}\sum_{j=1}^{|S|} w_i \sigma_{ij} w_j . \tag{2}$$

According to the MV model, the stock portfolio optimization problem can be modeled as a linear combination of the aforesaid objectives [4]

$$\begin{aligned} &\textbf{maximize } \boldsymbol{\mu}^{\mathsf{T}}\mathbf{w} - \lambda \cdot \mathbf{w}^{\mathsf{T}}\Sigma\mathbf{w} \\ &\textbf{s.t. } \mathbb{1}^{\mathsf{T}}\mathbf{w} = 1 \\ &\mathbf{w} \geq \mathbf{0} \end{aligned} \quad , \tag{3}$$

where

- $\lambda \in \mathbb{R}^{+}$ is the risk aversion coefficient, i.e., the larger the coefficient, the more risky the generated portfolio, and
- $\mathbf{w} \geq \mathbf{0}$ defines as positive (short-selling operations are not permitted) value each weight w_i.

Hereafter, we will make the following assumptions:

- The total amount available for stock investments is allocated.
- The amount allocated to each stock is kept fixed until the end of the holding time.

The traditional MV model formulation treats the set of input stocks S as a unique, large portfolio and assigns a continuous weight w_i to each stock $s_i \in S$. Conversely, in the present work, we address the binary stock selection problem [29] and rely on a uniform investment strategy. This entails a selection of an equally weighted portfolio over the power set $\mathbb{P}$ of S. Moreover, we apply a buy-and-hold strategy to invest in the stock markets (i.e., buy the securities and sell them at the end of the holding time).

4 The proposed mean–variance model adaptation

We will present here the adapted version of the traditional MV philosophy, whose goal is not to generate the desired portfolio by shortlisting single stocks, but rather to identify the best portfolio from a set of portfolio candidates.

As a preliminary step, portfolio candidates are generated by means of an itemset-based heuristic presented later on in Sect. 5.4. The idea behind it is to approximate the expected return of a candidate portfolio as a combination of daily returns of the least performing stock in the portfolio. However, notice that the EPP approach can be conveniently generalized and adapted to an arbitrary portfolio-level heuristic that can be computed in a scalable way.

In the itemset-based heuristic, each candidate stock portfolio satisfies a lower-bound estimate of the portfolio return [14] (LBPR, in short), which is defined and computed as follows

$$E_{\min}[P_q] = \quad \mathsf{average}_{d \in H} \left\{ \mathsf{minret}(P_q, d) \right\}, \tag{4}$$

where P_q is a selected portfolio identified as an element of the power set $\mathbb{P}$ of S, thus $q \in \{1, \ldots, |\mathbb{P}|\}$. The function minret$(\cdot)$ returns the least daily return over all the portfolio stocks on a given day d.

Then, we look for the portfolio $P \in \mathbb{P}$ that is best placed with regard to a single rank-based objective function, where the portfolio evaluation relies on an additional combination of expert-driven decision criteria. Thus, the key idea is to combine the portfolio-level return provided by an ad hoc measure of performance (in this work LBPR) with additional measures of performance of the candidate portfolios that can be independently generated using different strategies (e.g., volatility, wisdom of crowds).

The suggested selection model explores a subset of the portfolios $\mathbb{P}$ and stores into vector[1] $x_q \in \mathbb{P}$ the binary choice relative to each candidate portfolio

$$\begin{aligned} &\textbf{minimize } (1-\lambda)\mu^{\mathbb{P}}(x_q) + \lambda \cdot \Sigma^{\mathbb{P}}(x_q) \\ &\textbf{s.t. } \mathrm{c}^{\mathbb{P}}(x_q) \geq \mathsf{Th} \quad \mathsf{Th} \in \mathbb{R}^T \\ &x_q \in \mathbb{P} \end{aligned} \quad , \tag{5}$$

where

- The return ranking function $\mu^{\mathbb{P}}: \mathbb{P} \to \mathbb{N}^+$ directly interfaces with the adopted portfolio-level heuristic by returning the rank of a candidate portfolio, and in this work, it boils down to a ranking based on the lower bound estimate of the return $E_{\min}[P_q]$.
- The risk ranking function $\Sigma^{\mathbb{P}}: \mathbb{P} \to \mathbb{N}^+$ returns the rank of a candidate portfolio based on its risk measure, avoiding the estimation of statistical measures for all the combination of available stocks.
- Constraints $\mathrm{c}^{\mathbb{P}}: \mathbb{P} \to \mathbb{R}^{\mathrm{T}}$ allow end-users to set up multiple decision criteria based on a variety of T factors through a threshold Th, among which the stock diversification over sectors, the observed trends in the historical stock prices, and the fundamentals behind the considered assets. Constraint enforcement will be discussed later on (see Sect. 5.3).
- $\lambda \in [0, 1]$ is the *risk aversion* of the end-users, which allows us to make a combination of portfolio payoff and risk.

The ranking strategy replaces a combinatorial optimization approach, supplying a meaningful way to compare the generalizations of the risk and the return that could be a priori not comparable with a fully quantitative approach. Notice that in Eq. (5) the dependency on the portfolio family $\mathbb{P}$ is made explicit and the set of additional constraints can be conveniently adapted to the end-users' needs.

5 The early portfolio pruning method

The proposed hybrid method consists of a two-step process:

1. *Candidate portfolio generation* It analyzes historical stock-related data by means of a parallel itemset mining approach to generate a selection of candidate stock portfolios. The aim is to early identify a subset of promising stock portfolios based on a global trend analysis of the composing stock prices.
2. *Portfolio selection* It identifies, among the candidate portfolios generated at the previous step, the best choice according to both a set of end-users preferences and the analysis of additional stock-related data (e.g., fundamental analysis). This step is accomplished by a solver that addresses the adapted mean–variance philosophy described in Sect. 4.

5.1 Data model

We consider the following stock-related data:

- The daily Open-High-Low-Close-Volume (OHLCV) values, associated with each considered stock in the reference time period.

[1] Notice that for each portfolio P_q holds a biunivocal relation with the binary vector x_q with 1 when the stock s_i is selected in portfolio P_q and 0 otherwise.

Table 2 Transactional data representation. Reference time period $[d_1, d_6]$

Time stamp	Transaction
d_1	⟨A,5%⟩, ⟨B,5%⟩, ⟨C,−1%⟩, ⟨D,7%⟩, ⟨E,5%⟩
d_2	⟨A,2%⟩, ⟨B,6%⟩, ⟨C,0%⟩, ⟨D,2%⟩, ⟨E,2%⟩
d_3	⟨A,4%⟩, ⟨B,5%⟩, ⟨C,−2%⟩, ⟨D,4%⟩, ⟨E,5%⟩
d_4	⟨A,4%⟩, ⟨B,2.5%⟩, ⟨C,−4%⟩, ⟨D,10%⟩, ⟨E,4%⟩
d_5	⟨A,1%⟩, ⟨B,4%⟩, ⟨C,−2%⟩, ⟨D,7%⟩, ⟨E,1%⟩
d_6	⟨A,−1%⟩, ⟨B,6%⟩, ⟨C,0%⟩, ⟨D,1%⟩, ⟨E,−1%⟩

- A taxonomy that clusters stocks into homogeneous categories/financial sectors.
- The financial statements that periodically report the updates of the key economic stock indicators.[2]

OHLCV data are widely used to analyze stock price and volume trends by means of technical analyses as they are expected to inherently incorporate all the underlying effects. They are analyzed in the first step of the hybrid method (candidate portfolio generation).

The taxonomy consists of a set of aggregation hierarchies built over stocks. It is instrumental for diversifying the fund allocation across different sectors thus reducing the overall risk exposure [30].

Financial reports are commonly exploited in fundamental analysis to measure the equity intrinsic value by examining related economic and financial factors [31]. Aggregation hierarchies and financial reports are both used to drive the portfolio selection step.

5.1.1 Transactional stock price model

To generate the candidate itemset-based portfolios, the selected heuristic extracts from the OHLCV data the daily closing prices of each of the considered stocks and stores them into a transactional dataset [32]. Each transaction tr_x corresponds to a distinct trading day d_x in the reference time period $[d_{start}, d_{end}]$. tr_x consists of the set of pairs $\langle s_i, r_i^x \rangle$, where r_i^x is the percentage variation of the closing prices of stock s_i between days d_x and d_{x-1}.

An example of transactional dataset is reported in Table 2. It consists of six transactions, each one collecting the closing price variations (w.r.t. the preceding day) of the stocks A, B, C, D, and E on different trading days. For instance, on day d_1 the closing price of stock A has increased by 5% w.r.t. to the preceding day. Notice that in transactional data model the temporal order of the contained transactions is not relevant, i.e., the temporal order of day d_1-d_6 does not matter.

5.1.2 Taxonomy over stocks

We build a taxonomy over stocks to incorporate the information about stock membership into specific financial sectors. Each stock is mapped to the corresponding sector. In our experiments, the hierarchical stock relationships are derived from the standard GICS sector-based stock categorization.[3] Alternatively, end-users could automatically infer the relationships using ad hoc clustering-based methods, e.g., [33–35], subspace factorization or genetic algorithms, e.g., [36, 37].

[2] In the experiments, we will consider the quarterly reports published by **Yahoo! Finance** and available at https://finance.yahoo.com/ (latest access: December 2021).

[3] https://www.msci.com/gics (latest access: December 2021).

5.1.3 Financial statements

Fundamental analysis focuses on examining the economic and financial factors related to a stock (e.g., production, earnings, employment, housing, manufacturing, management). In the current work, we focus on the subset of fundamental factors selected by [38] to forecast stock performance, namely (i) rate of sales growth over the past year (SGI) [39], (ii) gross margin (GMG) [40], (iii) earning surprise (CHGEPS) [41], (iv) total capital expenditures (CAPX), (iv) revenues earned or expenses incurred (ACCRUAL) [42], and (v) level of research and development investments (R&D). However, it is possible to use as constraint other factors as well.

5.2 Candidate portfolio generation

Frequent itemset mining is an established unsupervised technique to discover recurrent item correlations from transactional data [43]. A frequent itemset is an arbitrary set of l items ($l \geq 1$) whose observed frequency of occurrence (support) is above a given threshold. In our context, itemsets represent arbitrary stock portfolios consisting of l stocks.

Traditional itemset mining algorithms such as Apriori [44] and FP-Growth [45] do not consider the weights associated with the items occurring in each transaction. In our context, item weights indicate the percentage closing price variation w.r.t. the preceding trading day.

More recently, various algorithm extensions have been proposed to incorporate item weights into the itemset mining process, e.g., [46, 47]. In parallel, lots of efforts have been devoted to parallelizing the extraction of frequent itemsets using Hadoop–Spark framework in order to scale toward Big datasets, e.g., [13].

The candidate portfolio generator in EPP extracts all the itemsets representing promising stock portfolios by adopting the portfolio-level heuristic evaluator previously proposed by [14]. The key idea is to filter out the combinations whose average least return of the composing stocks is below a given threshold. Since the current implementation of the presented heuristic method is centralized, it is unsuitable in its current form for coping with a very large initial stock set (see Sect. 6.4).

To overcome the above issue, we leverage the parallel and distributed itemset mining techniques presented by [48] and currently supported by the ML-Lib library [49]. Specifically, we tailor the parallel mining process to successfully cope with transactional data including item weights.

5.3 Portfolio selection

Modern financial decision support systems allow end-users to specify their preferences for portfolio selection by different levels of insight, thus extending the original Markowitz's work that was based only on the first two moments of the distribution of the returns. Decisions are commonly driven by (i) the current market conditions, (ii) the economic investors' preferences and attitude to risk, and (iii) the intrinsic economic value of the considered assets.

To identify the portfolio, EPP relies on the adapted mean–variance model previously described in Sect. 4. It allows the enforcement of a set of user-specified constraints both at the portfolio-level constraints. Specifically,

- *Fundamental analysis* Portfolios are evaluated in terms of the relative strength of the financial stock fundamentals. The ranking strategy shortlists the portfolios including top-ranked stocks across a variety of established financial indicators.
- *Diversification* Portfolios are expected to include stocks well diversified across sectors and markets. The portfolios that do not meet a sufficient level of diversification are early pruned.
- *Trend* Stock price trends are commonly used to plan trading strategies. Portfolios are shortlisted based on the underlying long-term price trends of the composing stocks, which are estimated using established technical analysis indicators.

A more detailed description of the supported constraints is given below.

5.3.1 Portfolio-level constraints based on fundamental factors

We assign a fundamental score to each portfolio based on the characteristics of the composing stocks. Specifically, according to [50], we first derive the financial/economic strength of each stock based on a variety of fundamental factors and then combine the per-stock scores to assign the portfolio-level score.

Starting from the initial set of fundamental indicators/ratios available in the fundamental reports (see Sect. 5.1), we extract a summary consisting of a sample of key financial information. The sample is extracted according to [38]. These values are discussed to be effective on stocks with extreme returns, and thus, they well fit an early pruned family of stocks that have been selected in the first stage of the algorithm by returns. To meet the end-user preferences, the selection process of the considered indicators/ratios is expert-driven. She decides from the default pool the indicators that are worth including.

The per-stock score is an integer number, ranging from zero to the number of activated indicators/ratios. It considers the number of factors that are placed in the upper percentile in the overall stock ranking.[4] The idea behind it is to appreciate the relative strength of the financial fundamentals of the stock only in terms of relative global quality, by looking at the distribution of each indicator amongst the pruned stocks.

5.3.2 Portfolio-level constraints based on diversification

To assess the level of risk of a portfolio, we verify that the selected candidates satisfy a minimum (user-provided) level of stock diversification according to the stock categorization specified in the input taxonomy. Specifically, we compute the diversification level of the portfolio as the percentage of stocks belonging to different categories. Notice that to manage risk exposure the minimum diversification level can be manually specified by the domain expert.

5.3.3 Portfolio-level constraints based on technical analyses

Stocks prices can be aggregated and analyzed using the classical technical analysis indicators and oscillators [51] such as Simple Moving Average (SMA) and Exponential Moving Average (EMA). They provide useful information about the underlying stock price trends and exchanged volumes.

[4] In compliance with [50] in the experiments we rank the stocks by decreasing summarizing factor score and then pick only those in the upper 80% percentile.

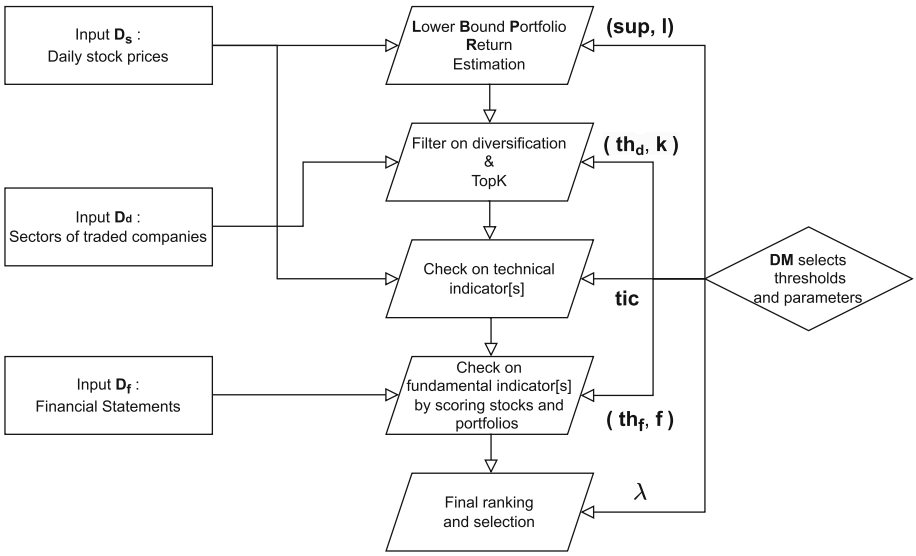

Fig. 1 Graphical representation of the main decision support system steps

To prevent the selection of portfolios that include stocks characterized by negative trends we also incorporate a portfolio evaluation based on technical analysis.

For example, EPP supports the comparison between the current portfolio prices and the simple/exponential moving average at a fixed periodicity (e.g., when the price is above the SMA with period 50 then a price uptrend is likely).

5.3.4 Risk aversion

To meet end-users' preferences, we ask them to set the value of risk aversion $\lambda \in [0, 1]$ in the adapted mean–variance logic. The higher λ, the higher the risk aversion (see Sect. 4).

Notice that alternative, more sophisticated approaches to attach risk aversion levels to stock portfolios (e.g., [52]) can be integrated as well.

5.4 The decision support system

Algorithm 1 presents the proposed decision support system. First, it runs the LBPR algorithm on the dataset of daily stock prices $\mathbf{D_s}$ Then, it ranks the result set shortlisting the top-**k** portfolios by filtering out all the portfolios with diversification level lower than the given threshold $\mathbf{th_d}$ and with fundamental score below $\mathbf{th_f}$. Finally, it selects the stock portfolio achieving the optimal balance between expected return and risk.

To better clarify the key steps adopted by the decision support system, Fig. 1 shows a sketch of the decision-making process.

5.4.1 Computational complexity

The computational complexity of the LBPR algorithm is mainly influenced by the itemset mining step, which is used to heuristically generate the candidate stock portfolios. The further

Algorithm 1 EPP: pseudocode of the decision support system

Input : $\mathbf{D_s}$: Dataset containing daily stock prices of traded companies
$\mathbf{D_f}$: Dataset containing financial statements of traded companies
$\mathbf{D_d}$: Dataset containing sectors of traded companies
sup: Minimum support threshold for LBPR (default: 8%)
l: Maximal portfolio size for LBPR (default: 7)
k: number of portfolios to maintain after LBPR step (default: 100)
$\mathbf{th_d}$: diversification threshold (default: $\mathbf{th_d}$ = 70%)
tic: Defined condition on technical indicator[s]
(default: daily closing price above/under SMA-50 periods)
f: Fundamental indicator[s] to be employed
$\mathbf{th_f}$: Fundamental indicators scoring threshold (default: 20^{th} percentile)
λ: Risk-based weight between rankings [0, 1] (default: 0.5)
Output: **ranking**: Ranking of suggested portfolios

```
/* Lower-Bound Portfolio Return Estimation                    */
R_lbpr, ranking_lbpr ← LBPR(D_s, sup, l)
R_div ← filterDiversification(R_lbpr, th_d, D_d)
R_topk ← filterTopK(R_div, ranking_lbpr, k)
/* Filtering portfolios                                       */
foreach portfolio p ∈ R_topk do
    if p satisfies tic then
        R_ti.insert(p)
L_stocks ← extractStocks(R_ti)
foreach stock s ∈ L_stocks do
    score_s ← scoringStock(s, f, D_f)
    L_scores.insert(score_s)
foreach portfolio p ∈ R_ti do
    score_p ← scoringPortfolio(L_scores)
    if score_p > th_f then
        R_f.insert(p)
/* Ranking result set                                         */
ranking_risk ← rankByRisk(R_f)
ranking ← (1 − λ) · ranking_lbpr + λ · ranking_risk
return ranking
```

steps, applied on top of a restricted subset of portfolios, have negligible impact on time and memory complexity.

Enumerating all the possible frequent itemsets in a large dataset is known to be NP-hard [53]. In particular, the number of generated candidates is linear with the dataset size and combinatorial with the number of input items. However, as discussed in [14], the optimal portfolio size is at least one order of magnitude lower than the number of candidate stocks. Hence, its impact is much less critical for maximal itemset mining.

LBPR adopts a parallel implementation of a maximal itemset mining algorithm [13], which guarantees a time complexity of $O(\frac{|D_s|}{P})$, where $|D_s|$ is the dataset size and P is the number of partitions used in the parallel and distributed computation. Empirical evidence of the algorithm scalability is given in Sect. 6.4.

6 Experiments

6.1 Experimental design

6.1.1 Data sources

We crawled stock-related data from Yahoo! Finance.[5]

6.1.2 Hardware and code

We run the experiments on a hexa-core 2.67 GHz Intel Xeon with 32GB of RAM, running Ubuntu Linux 18.04.4 LTS. The framework is written in the Python and Spark languages. The source code is available for research purposes upon request to the authors.

6.1.3 Backtesting

We run a set of backtesting trading simulations to evaluate the profitability and riskiness of EPP. The test periods are defined according to the bearish and bullish market states previously introduced in [54]. To this end, we first segment the raw price series of the analyzed market index into bearish and bullish market states, highlighted in Fig. 2, and then select reference time periods accordingly.

- *Bearish period* Period 2008–2009. It was mainly characterized by a bearish market condition due to the global financial crisis originated by the subprime mortgage crisis.
- *Bullish period* Period 2012–2015. It was characterized by a bullish market condition due to global economic growth. This period is a subsection of the 10-year long bullish period identified. Specifically, the selected subsection is characterized by the fastest growing market of the full period.
- *COVID-19 pandemic period* Period 2018–2020. It was characterized by a mix of bearish and bullish market states. This particular case study is related to the outbreak of the COVID-19 pandemic, the imposition of restrictions, and the end of the first epidemic wave. We consider it as real-life, challenging scenario.

Separately for each period, we run several backtesting simulations to assess the effectiveness and robustness of the portfolio optimization strategies on historical stock-related data relative to the NASDAQ-100 index. Specifically, we learn the itemset-based model using a six-month period (e.g., from July 1, 2007, to December 31, 2007, for the bearish period) and apply it to the next 12 months (e.g., for year 2018). For each simulation, we apply a buy-and-hold strategy, i.e., we buy the portfolio stocks at the beginning of the period and sell them at the end.

In all the performed simulations we consider an initial equity of 100,000 USD, we adopt a fixed-fractional money management strategy (with neither stop loss nor stop profit limits) and approximate per-trade transaction costs to 0.15% [55].

6.1.4 Competitors

We compare the performance of EPP with that of

[5] https://finance.yahoo.com. (Latest access: August 2020).

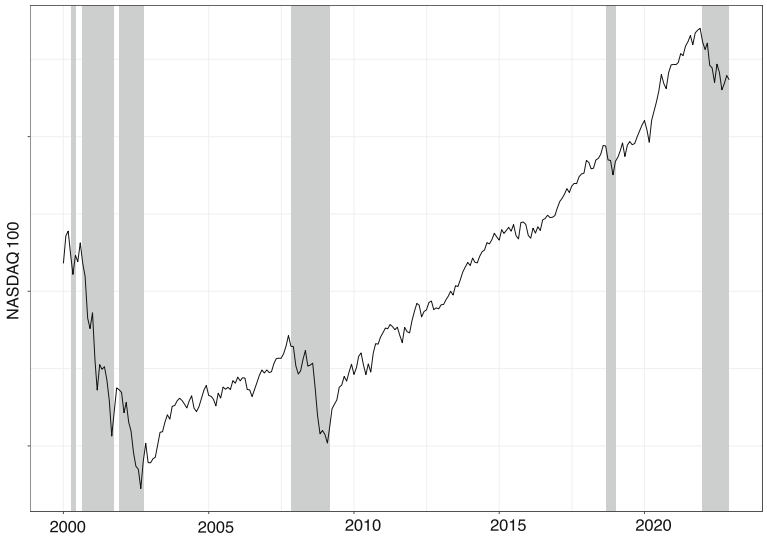

Fig. 2 The NASDAQ-100 index. Bearish periods are colored in gray whereas bullish ones are in white

- The **NASDAQ-100 benchmark**, which replicates the NASDAQ-100 index with no leverage.
- The established mean–variance model by Markowitz [55], where, to choose the optimal portfolio on the efficient frontier, we follow the strategy presented by [56] and optimize the choice according to the value of the Sharpe ratio [57], which measures the reward-to-variability ratio of a portfolio compared to a risk-free asset. This portfolio will be identified as **Markowitz–Sharpe** from now on.
- A set of recently proposed deep reinforcement learning (DRL) strategies to stock portfolio allocation available in the **FinRL** library, namely **A2C, TD3, and DDPG** [58].
- The most recently proposed itemset-based heuristic for portfolio generation, namely **DISPLAN** [14].
- Three established **US hedge funds** (only for the most recent COVID-19 pandemic period 2018-2020) investing on the same assets, i.e., MSEGX-Morgan Stanley Inst Growth A,[6] OLGAX-JPMorgan Large Cap Growth A,[7] PIODX-Pioneer Fund Class A.[8]

For Markowitz, we generate portfolios by using the *estimateMaxSharpeRatio* function from MATLAB (R2020b). For DISPLAN and EPP, we vary the minimum support threshold in the range [3%,12%], whereas the diversification threshold is set to 70%. To train FinRL, we consider 10 years of historical data to avoid the negative effects of data overfitting.

6.1.5 Evaluation metrics

For each trading simulation we analyze

[6] https://www.morganstanley.com/im/en-us/intermediary-manager-research/product-and-performance/mutual-funds/us-equity/growth-portfolio.shareClass.A.html (Latest access: March 2022).

[7] https://am.jpmorgan.com/us/en/asset-management/adv/products/jpmorgan-large-cap-growth-fund-a-4812c0506 (Latest access: March 2022).

[8] https://www.amundi.com/usinvestors/Products/Mutual-Funds (Latest access: March 2022).

- The **equity line plot**, which graphically shows the temporal variation of the equity during the test period [51].
- The **payout**, which is computed as the overall percentage return/loss of the portfolio at the end of the test period [4].
- The **Volatility**, which measures the standard deviation of the overall portfolio value with regard to the daily returns [4].

We graphically analyze the metrics above by plotting time series representing the percentage variation of the equity w.r.t. the initial value of the investment (e.g., see the equity line in Fig. 6a) and the daily percentage change in the price of the portfolio (e.g., see the volatility plot in Fig. 6b).

6.1.6 Scalability

We test the scalability of EPP with both the number of considered stocks and the size of the historical time window used in the learning phase to generate the itemset-based model.

To test the scalability with the number of stocks, we randomly add stocks of the Standard&Poor500 index to the initial stock set and rerun the simulations until the whole S&P500 index is covered.

6.2 Results of the backtesting simulations

Here, we present the following results:

- The comparison between the equity lines of the portfolios generated by EPP and those of the tested competitors (see Figs. 3a, 4a, 5a, 6a, 7a, and 8a). They provide a high-level view of the overall performance achieved by different approaches. For the sake of clarity, the comparisons with the Reinforcement Learning strategies are reported in separate plots (see Figs. 3b, 5b, and 7b).
- The comparison between the equities selected by EPP with those selected by the real hedge funds (see Fig. 9). The aim is to show the applicability of the proposed system in a real scenario.
- The volatility of EPP compared with those of the other approaches (see Figs. 4b, 6b, and 8b).

Hereafter, we will separately analyze each market period.

6.2.1 COVID-19 pandemic period

Figure 3a and b compares the equities in the COVID-19 pandemic period (years 2018–2020). EPP shows good resilience properties against negative market trends. For example, note the frequency with which DISPLAN values (dotted in orange) are subject to declines during different temporal windows (e.g., 09/2019-12/2019, 01/2020-05/2020), while EPP maintains profitable values. It outperforms both DISPLAN and Markowitz–Sharpe, while maintaining comparable results with regard to the DRL-based methods. By deepening the analysis of the COVID-19 pandemic outbreak period (see the equity lines in Fig. 4a and the volatility plot in Fig. 4b), EPP and DRL-based methods show a good capability to counteract the market drawdown than DISPLAN and Markowitz–Sharpe during the peak of the epidemic wave. The portfolio capability to be adaptive against adverse market movements is inherent in DRL agents, whereas turns out to be an empirical property of the combination of a static itemset-based model with the adapted Markowitz model.

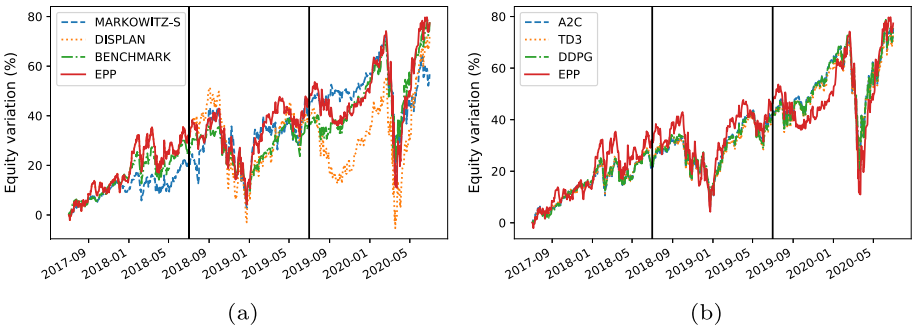

Fig. 3 Percentage variation of the equities. COVID-19 pandemic period. NASDAQ-100 index. **a** Comparison with benchmark, DISPLAN, and Markowitz–Sharpe. **b** Comparison with the Deep Reinforcement Learning strategies

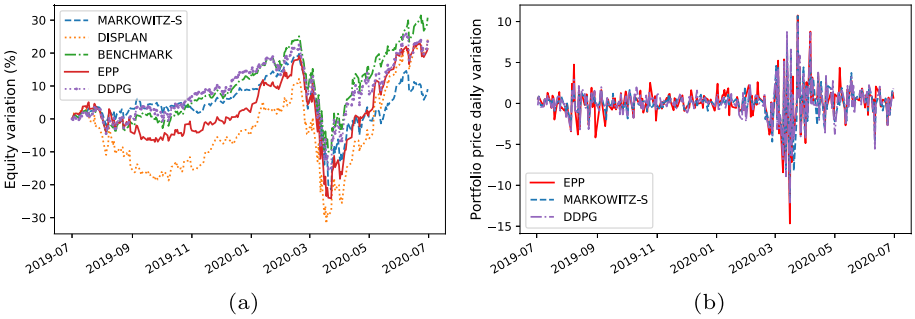

Fig. 4 Performance comparison during the outbreak of the COVID-19 pandemic. **a** Percentage variation of the equities. **b** Volatility plot

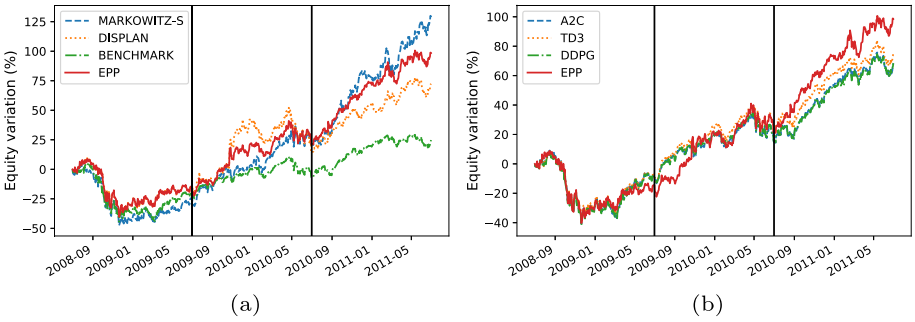

Fig. 5 Percentage variation of the equities. Bearish period. NASDAQ-100 index. **a** Comparison with benchmark, DISPLAN, and Markowitz–Sharpe. **b** Comparison with deep reinforcement learning strategies

6.2.2 Bearish market period

The analyzed bearish period (2008–2011) is relative to the 2008 financial crisis and the subsequent market recovery. In the aforesaid challenging scenario, EPP maintains a good performance in the first two years (2008–2010), only to be overtaken by Markowitz–Sharpe during the rally following the financial crisis. An opposite result occurs in the comparison

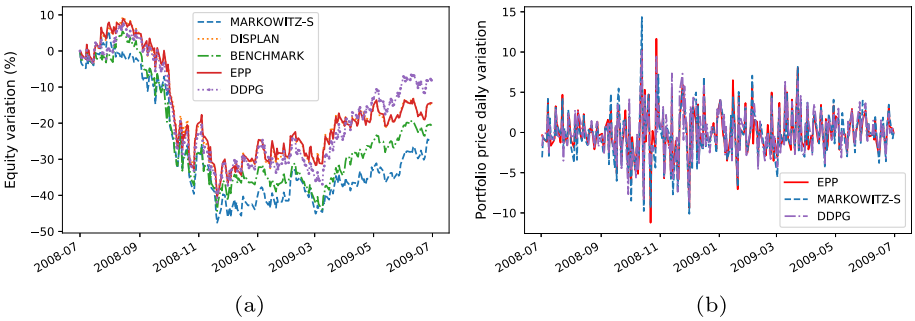

Fig. 6 Performance comparison during the outbreak of the financial crisis. **a** Percentage variation of the equities. **b** Volatility plot

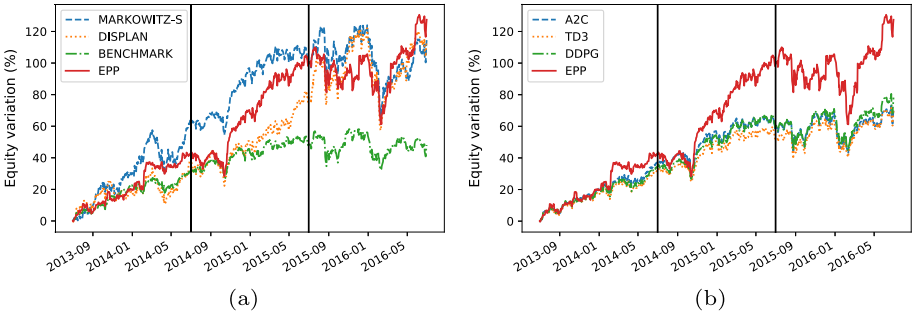

Fig. 7 Percentage variation of the equities. Bullish period. NASDAQ-100 index. **a** Comparison with benchmark, DISPLAN, and Markowitz–Sharpe. **b** Comparison with deep reinforcement learning strategies

with DRL-based models, where many crossovers occur in the first two years, culminating in a final overtaking in the rally phase by EPP. (see Fig. 5a and b). However, by focusing on the outbreak of the financial crisis, the EPP portfolio has shown to be less volatile than Markowitz–Sharpe and its drawdown and payout are roughly comparable to those of the DRL-based methods (see Fig. 6a and b).

6.2.3 Bullish market period

EPP performs better than the other tested competitors in the bullish period (see Fig. 7a and b). Deepening the analysis on the period of maximal market growth (see Fig. 8a), Markowitz–Sharpe payout is superior to those of EPP. However, the volatility is significantly higher (see Fig. 8b). The reason is that thanks to the portfolio-level constraints EPP is more conservative than Markowitz–Sharpe even in bullish market conditions when risky strategies that rely on very few stocks are rewarded.

6.2.4 Comparison with hedge funds

This confirms the usability of the proposed system in real-world scenarios.

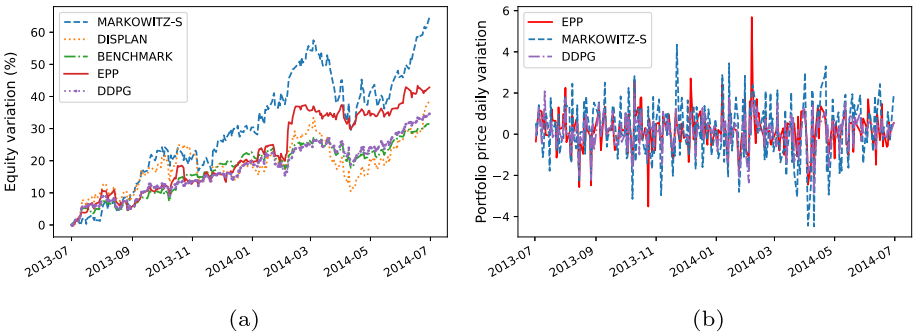

Fig. 8 Performance comparison during the period of most significant market growth (2013–2014). **a** Percentage variation of the equities. **b** Volatility plot

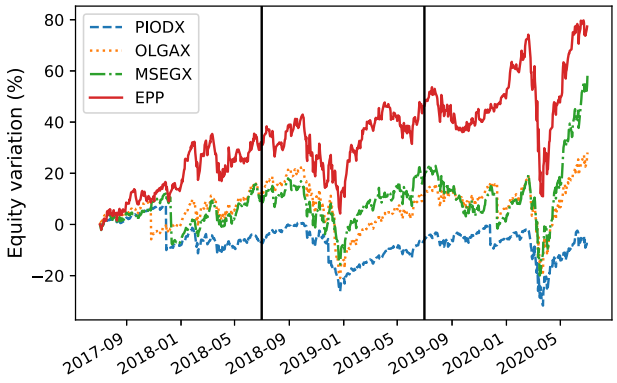

Fig. 9 Percentage variation of the equities. COVID-19 pandemic period. Comparison with real funds

6.3 Effect of the risk aversion

End-users can personalize the risk exposure of the EPP portfolio by conveniently setting the risk aversion λ. The higher λ, the more important is the risk-based ranking classification of the candidate portfolios (see Sect. 4).

We run a set of experiments to analyze the effect of the risk aversion on the performance of the generated portfolios. Figure 10a and b compares the daily volatility and payout distributions over all the analyzed years (2008–2020) achieved by setting a medium risk aversion ($\lambda = 0.5$) and no risk aversion ($\lambda = 0$), respectively. The mean payout values are roughly comparable with each other, whereas the volatility of the configuration setting with no risk aversion is consistently higher. However, setting an extreme configuration is not advisable because, on the one hand, taking into account no risk rankings may expose investors to more relevant market oscillations without yielding significant returns. On the other hand, the risk ranking alone would erase the pivotal role of the selected heuristic.

6.4 Scalability tests

We run several backtesting simulations using the EPP method to test the scalability with the number of considered stocks and the size of the training window.

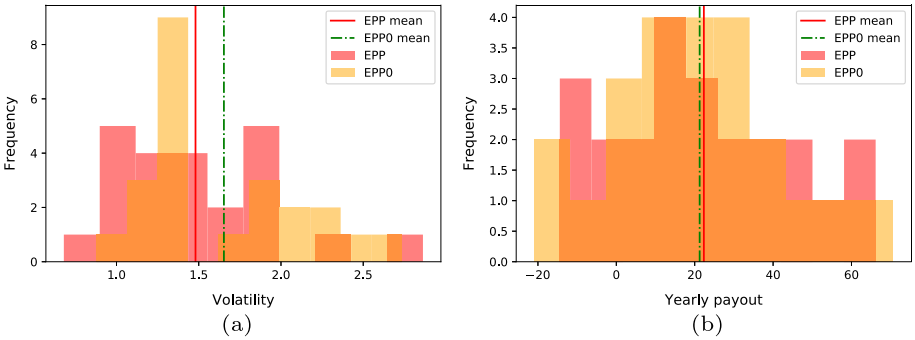

Fig. 10 EPP standard configuration with medium risk aversion (λ=0.5) vs. EPP with no risk aversion (λ=0). Years 2008-2020. **a** Distributions of the daily volatility statistic. **b** Distributions of the yearly payouts

Firstly, we tested EPP on a larger set of stocks, i.e., the entire S&P 500 index (500 stocks). Figure 12 shows the equities lines of both EPP and the benchmark S&P 500 index over the COVID-19 pandemic period. The results confirm the effectiveness of the proposed strategy.

Secondly, we vary the number of initial stocks from 10 to 500 to test the system scalability (see Fig. 11a). To evaluate the impact of the parallel itemset mining phase on the EPP time complexity, we test both a parallel and a centralized version of the system. In the centralized variant of EPP, we replace the PFP algorithm [13] with an efficient, centralized FP-Growth algorithm implementation.[9]

As expected, the increase in the number of initial stocks results in a super-linear increase of the execution time, mainly due to the generation of a combinatorial number of candidate itemsets, which are then processed in a sequential manner. Conversely, in the parallel version the job is distributed across multiple workers and the increase is approximately linear with the number of stocks.

Figure 11a shows a similar scalability test executed by keeping the number of initial stocks fixed to 100 (i.e., the stocks in the NASDAQ-100 index) and by varying the number of training months from 3 to 18. A nonlinear increase in the time complexity comes out by considering a training window size larger than 12. However, the latter scalability issue seems to be less critical than the former one as the standard window size configuration is typically between 6 and 12 months.

7 Conclusions and future works

The paper presented a hybrid financial decision support system for selecting stock portfolios. The framework allows for the combination of a parallel itemset mining process applied to historical stock price data with a tailored set of constraint and a risk-averse adjustment. The key idea is to simplify the complexity of stock-based approaches by early filtering part of the candidate portfolios during the initial itemset mining phase. Specifically, the extracted itemsets represent candidate stock portfolios, where we directly apply the traditional Markowitz's philosophy, allowing also for the enforcement of further portfolio-level constraints based on complementary knowledge provided by taxonomies and financial reports.

[9] http://fimi.uantwerpen.be/src/ (Latest access: December 2021).

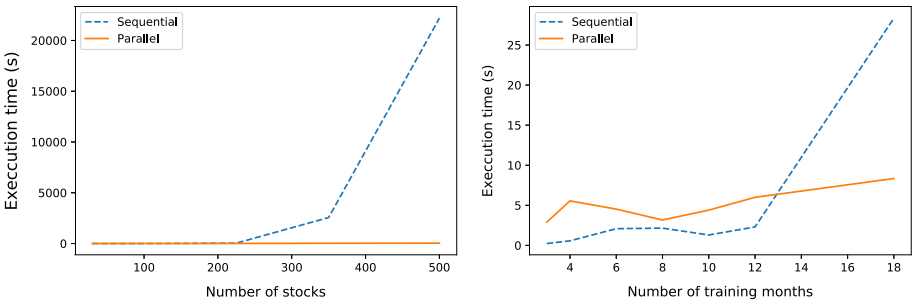

Fig. 11 Time complexity analysis. Comparison between EEP with parallel itemset mining with the EPP variant with sequential itemset mining. **a** Scalability with the number of initial stocks. **b** Scalability with the training window size

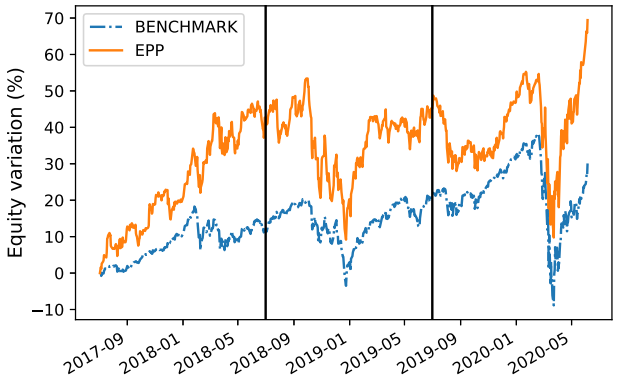

Fig. 12 Percentage variation of the equities. COVID-19 pandemic period. S&P 500 stocks. Comparison between EPP and the benchmark

The integration of a parallel itemset mining allows for the analysis of large stock sets not manageable by a centralized approach and the selected portfolios achieves good performances in terms of payout and risk exposure when compared to:

- Deep reinforcement learning methods. (Even if EPP relies on static stock portfolios, whereas DRL dynamically adapts the model to the current market situation.)
- Markowitz–Sharpe models.
- Established US hedge funds.

As future work, we plan to: (i) Extend the current hybrid method and DSS by integrating financial instruments other than stocks (e.g., exchange-traded funds); (ii) test the proposed approach on non-US stocks by simulating multinational scenarios and properly managing both multinational financial data and the geographical diversification over stocks; (iii) improve the way decision-makers reflect their risk aversion; and (iv) investigate the use of fuzzy rule models [59] and probabilistic itemset mining [60] to model market uncertainty.

Author Contributions DG: conceptualization, methodology, writing—original draft, writing—reviewing and editing, formal analysis, software, validation, visualization. JF: Conceptualization, methodology, writing—original draft, writing—reviewing and editing, supervision, project administration, software, validation. LC: Conceptualization, methodology, writing—original draft, writing—reviewing and editing, supervision, formal analysis, project administration.

Funding Open access funding provided by Politecnico di Torino within the CRUI-CARE Agreement.

Data Availability The datasets generated during and/or analyzed during the current study are available from the corresponding author upon request.

Declarations

Conflict of interest The authors did not receive support from any organization for the submitted work. The authors have no relevant financial or non-financial interests to disclose.

References

1. Markowitz H (1952) Portfolio selection. J Financ 7(1):77–91
2. Soleimani H, Golmakani HR, Salimi MH (2009) Markowitz-based portfolio selection with minimum transaction lots, cardinality constraints and regarding sector capitalization using genetic algorithm. Expert Syst Appl 36(3, Part 1):5058–5063. https://doi.org/10.1016/j.eswa.2008.06.007
3. Chen B, Zhong J, Chen Y (2020) A hybrid approach for portfolio selection with higher-order moments: empirical evidence from shanghai stock exchange. Expert Syst Appl 145:113104. https://doi.org/10.1016/j.eswa.2019.113104
4. Brandimarte P (2017) An introduction to financial markets: a quantitative approach. John Wiley & Sons
5. Konno H, Yamazaki H (1991) Mean-absolute deviation portfolio optimization model and its applications to tokyo stock market. Manage Sci 37(5):519–531. https://doi.org/10.1287/mnsc.37.5.519
6. Mansini R, Speranza MG (1999) Heuristic algorithms for the portfolio selection problem with minimum transaction lots. Eur J Oper Res 114(2):219–233. https://doi.org/10.1016/S0377-2217(98)00252-5
7. Chiodi L, Mansini R, Speranza M (2003) Semi-absolute deviation rule for mutual funds portfolio selection. Ann Oper Res 124(1):245–265. https://doi.org/10.1023/B:ANOR.0000004772
8. Angelelli E, Mansini R, Speranza MG (2012) Kernel search: a new heuristic framework for portfolio selection. Comput Optim Appl 51(1):345–361. https://doi.org/10.1007/s10589-010-9326-6
9. Bustos O, Pomares-Quimbaya A (2020) Stock market movement forecast: a systematic review. Expert Syst Appl 156:113464. https://doi.org/10.1016/j.eswa.2020.113464
10. Yang F, Chen Z, Li J, Tang L (2019) A novel hybrid stock selection method with stock prediction. Appl Soft Comput 80:820–831. https://doi.org/10.1016/j.asoc.2019.03.028
11. Kedia V, Khalid Z, Goswami S, Sharma N, Suryawanshi K (2018) Portfolio generation for indian stock markets using unsupervised machine learning. In: 2018 fourth international conference on computing communication control and automation (ICCUBEA), pp. 1–5
12. Thakkar A, Chaudhari K (2020) A comprehensive survey on portfolio optimization, stock price and trend prediction using particle swarm optimization. Arch Comput Methods Eng, 1–32
13. Li H, Wang Y, Zhang D, Zhang M, Chang EY (2008) Pfp: Parallel fp-growth for query recommendation. In: Proceedings of the 2008 ACM Conference on Recommender Systems. RecSys '08, pp. 107–114. Association for computing machinery, New York, NY, USA. https://doi.org/10.1145/1454008.1454027
14. Baralis E, Cagliero L, Garza P (2017) Planning stock portfolios by means of weighted frequent itemsets. Expert Syst Appl 86:1–17. https://doi.org/10.1016/j.eswa.2017.05.051

15. Mansini R, Ogryczak W, Speranza MG (2014) Twenty years of linear programming based portfolio optimization. Eur J Oper Res 234(2):518–535. https://doi.org/10.1016/j.ejor.2013.08.035
16. Merton RC (1969) Lifetime portfolio selection under uncertainty: the continuous-time case. Rev Econ Stat 51(3):247–257
17. Fama EF, French KR (2004) The capital asset pricing model: theory and evidence. J Econom Perspect. https://doi.org/10.1257/0895330042162430
18. Zopounidis C, Galariotis E, Doumpos M, Sarri S, Andriosopoulos K (2015) Multiple criteria decision aiding for finance: an updated bibliographic survey. Eur J Oper Res 247(2):339–348. https://doi.org/10.1016/j.ejor.2015.05.032
19. Zopounidis C, Doumpos M, Niklis D (2018) Financial decision support: an overview of developments and recent trends. EURO J Decis Processes 6(1):63–76. https://doi.org/10.1007/s40070-018-0078-3
20. Xidonas Panos (2021) Doukas, Haris, Sarmas, Elissaios: a python-based multicriteria portfolio selection dss. RAIRO-Oper Res 55:3009–3034. https://doi.org/10.1051/ro/2020140
21. Angelelli E, Mansini R, Speranza MG (2008) A comparison of mad and cvar models with real features. J Bank Financ 32(7):1188–1197. https://doi.org/10.1016/j.jbankfin.2006.07.015
22. Paiva FD, Cardoso RTN, Hanaoka GP, Duarte WM (2019) Decision-making for financial trading: a fusion approach of machine learning and portfolio selection. Expert Syst Appl 115:635–655. https://doi.org/10.1016/j.eswa.2018.08.003
23. Lai KK, Yu L, Wang S, Zhou C (2006) A double-stage genetic optimization algorithm for portfolio selection. In: King I, Wang J, Chan L-W, Wang D (eds) Neural information processing. Springer, Berlin, Heidelberg, pp 928–937
24. Chen C, Lu C, Lin C (2020) An intelligence approach for group stock portfolio optimization with a trading mechanism. Knowl Inf Syst 62(1):287–316. https://doi.org/10.1007/s10115-019-01353-2
25. Ertenlice O, Kalayci CB (2018) A survey of swarm intelligence for portfolio optimization: algorithms and applications. Swarm Evol Comput 39:36–52. https://doi.org/10.1016/j.swevo.2018.01.009
26. Chou Y-H, Kuo S-Y, Jiang Y-C (2019) A novel portfolio optimization model based on trend ratio and evolutionary computation. IEEE Transact Emerg Top Comput Intell 3(4):337–350. https://doi.org/10.1109/TETCI.2018.2868939
27. Chou Y-H, Jiang Y-C, Hsu Y-R, Kuo S-Y, Kuo S-Y (2022) A weighted portfolio optimization model based on the trend ratio, emotion index, and angqts. IEEE Transact Emerg Top Comput Intell 6(4):867–882. https://doi.org/10.1109/TETCI.2021.3118041
28. Markowitz HM (1991) Portfolio selection: efficient diversification of investments. Monograph/cowles foundation for research in economics at Yale University. Wiley. https://books.google.it/books?id=T2PHRWxp_RkC
29. DeMiguel V, Garlappi L, Uppal R (2007) Optimal versus naive diversification: how inefficient is the 1/N portfolio strategy? Rev Financ Stud 22(5):1915–1953. https://doi.org/10.1093/rfs/hhm075
30. Wang GY (2010) Portfolio diversification and risk reduction- evidence from taiwan stock mutual funds. In: Management and service science (MASS), 2010 international conference On, pp. 1–4. https://doi.org/10.1109/ICMSS.2010.5576482
31. Ponsich A, Jaimes AL, Coello CAC (2013) 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):321–344. https://doi.org/10.1109/TEVC.2012.2196800
32. Tan P, Steinbach MS, Kumar V (2005) Introduction to data mining. Addison-Wesley. http://www-users.cs.umn.edu/%7Ekumar/dmbook/
33. Nanda SR, Mahanty B, Tiwari MK (2010) Clustering indian stock market data for portfolio management. Expert Syst Appl 37(12):8793–8798. https://doi.org/10.1016/j.eswa.2010.06.026
34. Nair BB, Kumar PKS, Sakthivel NR, Vipin U (2017) Clustering stock price time series data to generate stock trading recommendations: an empirical study. Expert Syst Appl 70:20–36. https://doi.org/10.1016/j.eswa.2016.11.002
35. Fior J, Cagliero L, Garza P (2020) Price series cross-correlation analysis to enhance the diversification of itemset-based stock portfolios. In: Burdick D, Pujara J. (eds.) Proceedings of the sixth international workshop on data science for macro-modeling, DSMM 2020, In Conjunction with the ACM SIGMOD/PODS conference, Portland, OR, USA, June 14, 2020, pp. 1–116. ACM. https://doi.org/10.1145/3401832.3402680
36. Parque V, Mabu S, Hirasawa K (2009) Global portfolio diversification by genetic relation algorithm. In: ICCAS-SICE, 2009, pp. 2567–2572
37. Chen Y, Mabu S, Hirasawa K, Hu J (2007) Genetic network programming with sarsa learning and its application to creating stock trading rules. In: Evolutionary computation, 2007. CEC 2007. IEEE congress on, pp. 220–227. https://doi.org/10.1109/CEC.2007.4424475

38. Beneish MD, Lee CMC, Tarpley RL (2001) Contextual fundamental analysis through the prediction of extreme returns. Rev Acc Stud 6(2):165–189. https://doi.org/10.1023/A:1011654624255
39. Beneish MD (1999) The detection of earnings manipulation. Financ Anal J 55(5):24–36. https://doi.org/10.2469/faj.v55.n5.2296
40. Abarbanell JS, Bushee BJ (1998) Abnormal returns to a fundamental analysis strategy. Account Rev 73(1):19–45
41. Bernard VL, Thomas JK (1989) Post-earnings-announcement drift: delayed price response or risk premium? J Account Res 27:1–36
42. Sloan RG (1996) Do stock prices fully reflect information in accruals and cash flows about future earnings? Account Rev 71(3):289–315
43. Agrawal R, Imielinski T, Swami AN (1993) Mining association rules between sets of items in large databases. In: Buneman P, Jajodia S (eds.) Proceedings of the 1993 ACM SIGMOD international conference on management of data, Washington, DC, USA, May 26-28, 1993, pp. 207–216. ACM Press. https://doi.org/10.1145/170035.170072
44. Agrawal R, Srikant R (1994) Fast algorithms for mining association rules in large databases. In: Proceedings of the 20th international conference on very large data bases. VLDB '94, pp. 487–499. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA
45. Han J, Pei J, Yin Y (2000) Mining frequent patterns without candidate generation. In: Proceedings of the 2000 ACM SIGMOD international conference on management of data. SIGMOD '00, pp. 1–12. Association for computing machinery, New York, NY, USA. https://doi.org/10.1145/342009.335372
46. Tao F, Murtagh F, Farid M (2003) Weighted association rule mining using weighted support and significance framework. In: Proceedings of the ninth ACM SIGKDD international conference on knowledge discovery and data mining. KDD '03, pp. 661–666. Association for computing machinery, New York, NY, USA. https://doi.org/10.1145/956750.956836
47. Cagliero L, Garza P (2014) Infrequent weighted itemset mining using frequent pattern growth. IEEE Transact Knowl Data Eng 26(04):903–915. https://doi.org/10.1109/TKDE.2013.69
48. Han J, Pei J, Yin Y (2000) Mining frequent patterns without candidate generation. SIGMOD Rec 29(2):1–12. https://doi.org/10.1145/335191.335372
49. Meng X, Bradley J, Yavuz B, Sparks E, Venkataraman S, Liu D, Freeman J, Tsai D, Amde M, Owen S, Xin D, Xin R, Franklin MJ, Zadeh R, Zaharia M, Talwalkar A (2016) Mllib: machine learning in apache spark. J Mach Learn Res 17(1):1235–1241
50. Mohanram PS (2005) Separating winners from losers among lowbook-to-market stocks using financial statement analysis. Rev Acc Stud 10(2):133–170. https://doi.org/10.1007/s11142-005-1526-4
51. Murphy JJ (1999) Technical analysis of the financial markets: a comprehensive guide to trading methods and applications. New York institute of finance series. New York institute of finance, (1999). https://books.google.it/books?id=5zhXEqdr_IcC
52. Escobar-Anel M (2022) Multivariate risk aversion utility, application to ESG investments. North Am J Econom Financ. https://doi.org/10.1016/j.najef.2022.101790
53. Yang G (2004) The complexity of mining maximal frequent itemsets and maximal frequent patterns. KDD '04, pp. 344–353. Association for computing machinery, New York, NY, USA. https://doi.org/10.1145/1014052.1014091
54. Lunde A, Timmermann A (2004) Duration dependence in stock prices. J Business Econom Stat 22(3):253–273. https://doi.org/10.1198/073500104000000136
55. Williams T, Turton V (2014) Trading economics: a guide to economic statistics for practitioners and students. The Wiley Finance Series. Wiley. https://books.google.it/books?id=vYlPAwAAQBAJ
56. Bailey D, Lopez de Prado M (2012) The sharpe ratio efficient frontier. J Risk 15:3–44
57. Sharpe WF (1994) The sharpe ratio. J Portf Manag 21(1):49–58. https://doi.org/10.3905/jpm.1994.409501
58. Liu X, Yang H, Chen Q, Zhang R, Yang L, Xiao B, Wang CD (2020) Finrl: a deep reinforcement learning library for automated stock trading in quantitative finance. CoRR abs/2011.09607
59. Yao K, Qin Z (2021) Barrier option pricing formulas of an uncertain stock model. Fuzzy Optim Decis Mak 20(1):81–100. https://doi.org/10.1007/s10700-020-09333-w
60. Li Z, Chen F, Wu J, Liu Z, Liu W (2021) Efficient weighted probabilistic frequent itemset mining in uncertain databases. Expert Syst J Knowl Eng. https://doi.org/10.1111/exsy.12551

Daniele G. Gioia is a Mathematical Engineer pursuing a Ph.D. in dynamic optimization under uncertainty at Politecnico di Torino. His application context deals with decision-making under uncertainty in engineering/management applications. He previously earned his master's and bachelor's degrees in mathematical engineering at Politecnico di Torino.

Jacopo Fior is a PhD student at the Department of Control and Computer Engineering of Politecnico di Torino. He obtained his Bachelor's and Master's Degree in Computer Science at UniTO (Università degli Studi di Torino) and collaborated as a research assistant with the University of Helsinki. His current research interests are related to the study and application of machine learning and data mining techniques to time series data and, more specifically, to financial data.

Luca Cagliero has been associate professor at the Dipartimento di Automatica e Informatica of the Politecnico di Torino since January 2020. His current research interests are in the fields of pattern mining and deep natural language processing. Specifically, he has worked on text summarization, classification, and association rule mining. He is currently an Associate Editor of two international journals. He has published 140+ papers in international journals, book chapters, and conference proceedings.