Expert Systems With Applications 221 (2023) 119677

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Contextual combinatorial bandit on portfolio management

He Ni [a], Hao Xu [b,*], Dan Ma [a], Jun Fan [c]

[a] Tailong Finance School, Zhejiang Gongshang University, HangZhou, 310018, Zhejiang, China
[b] Fintech Research Centre, Zhejiang Lab, HangZhou, Zhejiang, China
[c] School of Business Administration, Zhejiang Gongshang University, HangZhou, 310018, Zhejiang, China

ARTICLE INFO

Keywords:
Bandit learning
Portfolio management
LinUCB
Ensemble learning

ABSTRACT

Portfolio optimization is a classic problem in finance, and assumptions are generally accepted about the stochastic dynamics of prices and market information. The Linear Upper Confidence Bound (LinUCB) algorithm based on bandit learning is a typical reinforcement learning approach that can be a data-driven approach to the portfolio problem. Most investors may have inconsistent preferences for different assets; therefore, the reward function needs to be designed to balance exploration (allocating part of the investment to searching among other alternative portfolios) and exploitation (concentrating the investment on the historically best portfolio). The study consists of developing a two-stage investing strategy that uses the supervised adaptive decision tree approach to build a pool of candidate portfolios and the reinforcement LinUCB algorithm to derive a portfolio update rule with respect to a specific utility function. The two-stage strategy has been tested on CSI300 constitute stocks listed on the Shanghai Stock Exchange. The accumulated returns, as well as other performance measures (e.g., maximum drawdown, Sharpe ratio, and information ratio) at different testing periods achieved by the model, can well outperform the benchmark index.

1. Introduction

Due to the complexity, nonlinearity, and chaos inherent in the asset pricing process, portfolio management is considered a challenging task. Investors have a huge selection of various assets to choose from. Significant facts must also be considered. The majority of them have been shown to have some effects on both trade volume and price volatility. Another deciding aspect of portfolio management is risk preference. Making a thoughtful and informed investment is, therefore, quite difficult. A decision algorithm needs to be optimized to select among several competing options. Each option is associated with a certain context (historical performance or profile) and reward (possible profit). When talking to a portfolio advisor, an investor would ask about things like how risky different assets are and how much the advisor knows about a specific industry. The portfolio advisor should respond with a pool of candidate assets. If the idea works and the investor agrees, both parties receive some reward (profits). As a result, the sequential decision-making challenge of portfolio selection—a process of reinforcing—can be related to a variety of real-world issues, including news recommendations, clinical trials, and advertising. In such a process, learners are faced with a trade-off between taking bold actions (e.g., try stocks with higher volatility) to acquire new knowledge and using existing knowledge conservatively (e.g., keep those historically good performing stocks) in each trial, which is often called the exploration–exploitation dilemma (Lavie & Rosenkopf, 2006). This dilemma originated from the Multi-armed Bandit(MAB) problem[1] (May et al., 2012; Scott, 2010), which is generally known as a single-state Markov decision process (Fianu & Davis, 2018).

According to the Efficient Market Hypothesis (EMH) (Awiagah & Choi, 2018), stock price prediction is impossible, and the distribution of stock prices is random and independent. Many studies, however, disprove the EMH by demonstrating the long-term memory of stock values. In the long run, active strategies can outperform passive strategies; that is, excessive returns can be achieved, or a winner group (e.g., a group of stocks that outperform the benchmark index on average) can exist. For instance, Chourmouziadis and Chatzoglou (2016) proposes an algorithm based on fuzzy logic combined with technical analysis. The profitability of their model was well above the benchmark at the Athens Stock Exchange. Most research focuses primarily on developing an investment portfolio using historical data. However, from a statistical point of view, the results of many previous studies were different under different conditions, and their long-term effects are still unknown. Historical data might not be as helpful as the most recent data, particularly in a turbulent market. Too much information

* Corresponding author.
E-mail addresses: nihe@zjgsu.edu.cn (H. Ni), xuhao.econ@outlook.com (H. Xu), feimodan@zjgsu.edu.cn (D. Ma), fj@zjgsu.edu.cn (J. Fan).

[1] The MAB can be thought of as a family of learning strategies, and the LinUCB used in this work is one of such strategies.

https://doi.org/10.1016/j.eswa.2023.119677
Received 25 March 2022; Received in revised form 10 January 2023; Accepted 5 February 2023
Available online 20 February 2023
0957-4174/

may incur overfitting, worsening the out-of-sample performance of a predicting model. Investors in the market are consistently overwhelmed with excessive information and a vast number of choices. They should be able to combine the expected returns based on past information and the uncertainty based on current information. This study aims to use the LinUCB algorithm developed by Auer (2002) to design a strategy that can dynamically form some optimal portfolios. The strategy will change the assets in the portfolio based on how the market is doing. This means that the portfolio that was chosen and worked before could always be changed. The estimated historical returns predicted by ridge regression are combined with an unclear margin of extra returns predicted by LinUCB utilizing real-time pricing information to get the total return. An empirical hyper-parameter will help balance the two parts of the returns (Garivier & Moulines, 2011). A system like the one advocated for is consistent with the well-known anticipated utility theory (Abdellaoui, 2002), which favors investments with maximum expected utility over those with the largest expected returns. Risk-adjusted real-time (e.g., online) returns can be estimated by the linear UCB algorithm. The algorithm can serve different investors and continuously update itself during online learning, which means that the algorithm will gradually know the market and investors, beside these, its recommendations' overall performance will also improve.

A pool of potential portfolio candidates must be chosen in advance to support a sequence of portfolio selections using the LinUCB algorithm. The stock market has flawless historical data, and there are significant correlations between the stock returns and the data/features from the stock price and the financial statements of the listed companies (O'Hara et al., 2000). Therefore, we use a popular ensemble learning algorithm (Liao et al., 2014), a decision tree, to recursively form a group of portfolios, each meeting two requirements: (1) the portfolio's constitute stocks are closely correlated in terms of relevant features,[2] and (2) the performance in terms of stock return and volatility of stocks in a candidate portfolio is similar. The decision tree model is trained to produce a large number of portfolio candidates (i.e., a portfolio means a group of stocks) with better performance than the market average. The term *homogeneous* in this context refers to the idea that stocks within a portfolio have more in common with one another (i.e., they should have the same fundamentals as those listed in Table 1 and exhibit similar performance in terms of historical return and volatility) . In contrast, stocks in different portfolios should not be all that similar in terms of return, volatility, and other fundamentals.

Meanwhile, the performance of the selected portfolio should be better, on average, than that of the benchmark, for instance, the local stock index. Decision trees are often used as base learners in ensemble learning for classification or regression (Chen, 2011; Sun et al., 2018). Their success can be explained by the fact that they can make representations that are easy for both experts and non-experts to understand and interpret. In the case of the stock market, data may suffer from imperfections . For example, features of listed companies can be a mixture of quantitative and qualitative. A decision tree can easily handle features of different natures and find some relatively purified portfolios by aggregating splits based on a decrease in information entropy.

Therefore, the creation of a decision-tree or CART bagging in particular (will be explained in more detail in the methodology section), based candidate portfolio pool, and a LinUCB learning-based portfolio selection procedure make up the entire system that this study has presented. Historically, better-performing stocks have been grouped into different portfolios by some of their features believed to be correlated with their future performance . These pre-selected portfolio candidates are chosen one at a time based on both their past performance and current information, as well as on the risk preferences of investors. The system is expected to automatically provide suitable recommendations to either institutional traders or individual investors according to the LinUCB algorithm. The strategy generated by the proposed model can be more aggressive than typical mean–variance analysis since it takes into account the unidentified risk associated with volatile stocks.

The rest of the paper is organized as follows. Section 2 reviews the relevant research and summarizes our contribution to the literature. Section 3 presents the methodology, describing how decision tree and LinUCB bandit learning are adapted and combined to form a two-stage model that could learn to fit investors' preferences. Section 4 presents the results of our testing experiments on stocks listed on CSI 300. Section 5 comments on the direction of future work and offers conclusions.

[2] The features would contain information related to stock returns and their fundamentals. The economic implications of selected features would be explained in details in Data and Experiments section.

Table 1
List of features that are used in forming the portfolio.

Name	Economic implications	Source literature
QRR	Quantity Relative Ratio:(Total number of lots traded/Current cumulative open time (min))/Average volume per minute over the past 5 days	Ge et al. (2016)
DV-TTM	Dividend Yield Ratio (Trailing Twelve Months): Dividends per share/Current share price for the last 4 quarters	Phan et al. (2015), Lemmon and Nguyen (2015)
NPG	Net Profit Growth: The profit that the company generates after accounting for all its expenses	Yadav et al. (2022)
EM	Equity Multiplier: The portion of a company's assets that is financed by stockholder's equity	Karolyi and Wu (2018)
PE	Price Earning Ratio: The ratio of a company's share price to its earnings per share.	Bagella et al. (2005)
NAVPS	Net Assets Value Per Share(year over year): Annual increase in net assets per share/Net assets per share at beginning	Cen et al. (2017), Mama (2018)
TR	Turnover Rate: Trading volume of last quarter/total shares outstanding (lots)	Xiao and Kong (2012)

*The correlation to investment return is an average value of correlation coefficients between return series and the indicator series. In case when two series are not of same frequency, the return series were sampled to facilitate the estimation.

2. Related literature and contribution

We focus on previous works that have influenced our research in this section. There has been a substantial amount of published research on the use of ensemble learning and reinforcement learning techniques in a variety of applications, including portfolio management.

2.1. Reinforcement learning: LinUCB and multi-armed bandit

For one class of adaptive optimal control problems, Sutton et al. (1992) first developed the reinforcement learning method (Q-learning) and showed its analytically proven efficiency. Almahdi and Yang (2017) put forth a coherent risk-adjusted performance objective function and a recurrent reinforcement learning method, demonstrating how the suggested portfolio trading system consistently outperforms hedge fund benchmarks. The majority of the latter research (Aboussalah & Lee, 2020; Dempster & Leemans, 2006; Gorse, 2011; Huo & Fu, 2017) heavily used reinforcement learning when developing decision-support systems for asset allocation and portfolio management.

An example of a typical reinforcement learning issue is the Multi-Armed Bandit (MAB), which is described as a sequential decision model that must continuously select one action from a group of actions, or *arms*. In their comprehensive evaluation of 1327 articles published since 2000, Silva et al. (2022) assess the relevance of MAB-based recommendation techniques to problems with recommender systems. However, asset allocation and other fields pertaining to investments have only very infrequently seen the application of LinUCB or MAB.[3] In a multi-armed bandit scenario, Sani et al. (2012) introduced risk aversion and defined the risk of a learning algorithm as the variability of the rewards sequence. Vakili and Zhao (2016) applied UCB and DSEE policies and defined a finer measure of regret in terms of a commonly used mean–variance risk measure in mathematical finance. Zimin et al. (2014) defined the LCB algorithm (the variant of UCB) that takes into account a risk measure that is a linear combination of the mean and the square root of the variance achieves desirable performance under specific circumstances. They also set the measure of goodness of an arm to be a function of the mean and the variance. An application of MAB on S&P 500 stock data was made by Huo and Fu (2017). In addition, they used social network analysis, using vertices to represent the stocks (nodes) and edges to show interactions between them. Regarding the centrality in graph theory, the K vertices with the smallest degree of periphery are selected for the candidate portfolios.

2.2. Ensemble learning: CART bagging

Due to its flexibility and efficiency in extracting meaningful information content from a vast range of features, bagging has been frequently used as a general-purpose learning method (Jordan et al., 2017; Shen et al., 2019; Yin, 2020) for portfolio management or asset allocation challenges. In Suzuki and Tanaka (2014), predicted values from the bagging method are combined to create a probability distribution. They obtain good accuracy with relatively little risk in their modeling of Japanese stocks. According to Jordan et al. (2017), when portfolio weights are regulated, bagging generally increases forecast accuracy and produces obvious economic improvements relative to the benchmark. Meanwhile, bagging has reportedly demonstrated excellent performance in producing reliable clustering solutions. Buhmann and Fischer (2003) use bagging to path-based clustering on a sizable set of image data. The potential alternative method put forth by Liu and Song (2017) enhanced the prediction of the S&P 500 index by integrating a few weak artificial neural networks with optimal bagging. According to Shi et al. (2009), the bagging ensemble significantly improves bankruptcy prediction. By integrating the advantages of boosting and bagging, Yang and Jiang (2016) presented a unique hybrid sampling-based clustering ensemble.

2.3. Contribution to literature

Due to the primary limitation that the cited research works only select the best single machine to play at each trial, they are not immediately applicable to our problem. The previously mentioned research acts as our source of inspiration as we carefully analyze the issue of asset allocation or portfolio management in the model. In the early stage, a selection of possible portfolios has been made to address this issue strategically and logically. Subsequently, these candidate portfolios are chosen to reflect the shifting market circumstances. Our study suggests a model that is marginally more aggressive than the mean–variance approach.

The novelty of our paper mainly related to the methodology, i.e., we chose candidate portfolios sequentially from a pre-selected basket instead of picking individual stocks. The developed LinUCB searching method is used to determine the sequence as well as the weights of constitute stocks, and bagging is used to build the basket.

Both methodological developments and algorithmic insights can be seen in our solution. We put into practice a context-based MAB algorithm with a linear utility function that takes asset volatility and return into account. Exploration and exploitation trade-offs naturally connect to the traditional return–risk balance in portfolio selection. Numerous utility functions have been documented in the literature as solutions to the problem. In our study, expected return represents exploitation, while volatility represents exploration. The LinUCB model helps us get high performance in a market that is uncertain and changing, without having to explicitly define a utility function, which is hard to do for most portfolio management problems.

3. Methodology

Bandit learning is a reinforcement learning algorithm that can deal with the exploration and exploitation dilemma. It has, therefore, natural applications to portfolio management issues (Gutowski et al., 2021). In this paper, we present a bandit algorithm for selecting a series of portfolios (i.e., a portfolio at a time, from a pool of candidate portfolios). The optimal portfolio strategy represents a combination of passive and active investment in line with a mean–variance trade-off or exploration–exploitation dilemma. The idea is to find a series of portfolios that can ensure sub-optimized returns based on their past performance while exploring new opportunities that might give high returns in the future. According to the UCB algorithm, each portfolio candidate, also known as an arm, should achieve higher average payoffs — i.e., exploit its past performance to select the arm that did best (Shen et al., 2015). Meanwhile, due to the imprecision in the estimation of payoffs for any portfolio, we could also explore by choosing portfolios with good potential to do well in the future. Obviously, exploration may increase short-term regret or mistakes by choosing a suboptimal portfolio . However, combined with exploitation, exploration can reduce long-term regret. generally, a balanced trade-off would surely outperform either pure exploitation or exploration.

Silva et al. (2022) summarized that bandit learning is typically used to recommend products or services, but its application to investment is still limited. Our study proposes a non-standardized MAB learning algorithm, as some unique features of stock portfolios make this application a bit different from ordinary bandit learning. The major distinctions are as follows:

1. The original MAB got its name from a gambling thought experiment in which a player has to choose from a number of slot machines with different payouts. These machines appear to perform independently of one another. The performances of stocks, on the other hand, are strongly linked because they all face the same systematic risk or market risk.
2. The portfolio management problem requires a group of sub-optimal assets at a time, whereas the original MAB learning simply selects the best choice at a time.
3. When MAB is used for product recommendations, unpicked products cannot receive customer feedback. Similarly, unpicked jackpot machines cannot be operated by anyone. However, the price of stocks and the fundamentals of companies are constantly changing and always available to the general public because stocks are traded by such a large number of individuals in a liquid market.

3.1. Tree based portfolio

The decision tree is commonly used in operations research and machine learning (Somvanshi et al., 2016). It is non-parametric and can handle both continuous and nominal variables. A decision tree can generate interpretable classification rules and is particularly attractive for this study because investors need an explicit explanation of certain investment decisions. The decision tree allows investors to explore

[3] Among 230 most relevant studies conducted by Silva et al. (2022), none of them is reported in the field of portfolio management or general investment.

features that could materially affect their decisions in allocating their assets. Its simple structure and inherent ease in interpreting a set of choices (based on several features) make it very popular in a broad range of applications (Chang et al., 2018; Xia et al., 2017). In the case of classification, each internal node represents a binary split on an attribute(e.g., whether the company's PE ratio is larger than 20); each branch represents the condition of the test; and each leaf node can be represented by a class label with conditional probabilities, or a group of samples with certain impurity measures, such as information entropy or the Gini index (Liu et al., 2017; Zheng & Wang, 2018).

We aim to find a number of winning/sub-optimal stock combinations/portfolio candidates. These portfolio candidates can be either mutually exclusive or overlapping, which means that professionals, who are supposed to make an experienced selection of outperforming assets from a common pool despite having different perspectives and considerations, should also reach some agreement by picking similar stocks. Therefore, the original algorithm of the decision tree needs to be modified. In any non-terminal node, it is used to perform a binary split with a specific attribute f_i and an associated split value v_{f_i}. In a traditional approach, data belonging to that node will be split into two leaf nodes without any overlap. In order to mimic the real scenario, the original splitting rule is modified to be more adaptive, as explained in the following three settings:

Splitting criterion Given that Table 1 identifies and specifies a total of nine characteristics, we need to build a method that can be utilized to select the feature that leads to the highest gain in purity (for example, the Gini index) prior to completing the splitting. Each portfolio has a discrete utility label to help with the Gini index computation. We suggest using the mode of stock quantiles to figure out the discrete utility label for a certain node. Detailed descriptions will be provided afterward .

Stopping criterion This criterion is used to prevent overtraining. Apart from the requirement of purity – that is, the Gini index calculated by the label value of stocks assigned to the node should be smaller than a threshold G^* – the stocks in the portfolio should achieve appropriate utility simultaneously. Since there is no benchmark to suggest the best achievable utility of a portfolio with a given number of stocks, we propose that any portfolio candidate have a minimum number min_{asset} of assets and a lower bound of historical average utility u_{lb}. A leaf node is considered terminal if the number of assets classified to that node is equal to or less than min_{asset} or their average utility is equal to or less than u_{lb}. The lower bound utility and minimum number of assets are determined empirically.

Overlapped splitting When the utility of the portfolio before splitting has already been satisfied, we should consider a non-binary split because successful investors may have some commonalities in their choice of investments. The assemblage of assets is of exceptional quality, according to the economic implications. Because the algorithm indicates that more splitting will make the utility go up, we split, but we keep some overlap. The amount of overlap should correspond to the utility level attained. We assume that the quantile of utility $q(u)$ measures the percentage of utilities less than u for the given period, and the level of the overlap can be controlled by $\xi(u) = 1 - q(u)$. Thus, the original split e.g., $\gamma > v$ and $\gamma < v$ — is replaced by $\gamma \geq \xi(u)v$ and $\gamma < (1 + \xi(u))v$

The classic decision tree algorithm is a family with several similar algorithms, such as ID3, C4.5, and CART (Sharma & Kumar, 2016). Since the data collected from the real-time market and financial statements is more likely to be continuous, and the distribution of the value of these continuous variables may always be far from normal distribution, ID3, as well as its extension, C4.5, may not be very suitable due to the high possibility of missing values and model overfitting. In this study, we apply CART (Ozturk et al., 2016) for choosing portfolio candidates. The end leaf nodes with low Gini values and higher utility can be treated as appropriate nodes with qualified portfolio candidates.

Each sample stock was assigned an utility score estimated by a linear sum of its past average return and associated volatility in the form of standard deviation. The setting of risk aversion coefficient and time horizon for utility estimation is explained in Section 4.2. Each utility score is further discretized according to its quantile and serves as the label for each stock (e.g., the stock whose past utility score is within best 10% of the stocks in the whole pool is labeled 1. The stock is labeled 2 if its utility is within the 10% – 20% range, etc.)

After training, the stocks are "grouped" by the end nodes of the trained CART. Conventionally, the performance of a portfolio as a whole cannot be calculated as the linear sum of its constituent stocks' individual performances because excellent stock performance may be offset by bad stock performance, resulting in an average performance for a portfolio with both good and bad stocks. However, for a group of good-performing stocks with similar fundamentals, we can assume their return patterns are highly correlated (i.e., the correlation coefficient is close to 1), and consequently the utility can be approximated by a linear sum. Therefore, we define the utility of the portfolio as the linear sum of the utilities of its constituent stocks . The losing nodes have a group of "underperforming" stocks that under-perform 50% of the best stocks, and have a high Gini index, which means there are very few better performing stocks in the group. Branches leading to such nodes are pruned during and after training for high computational efficiency. The winning nodes have a group of stocks with a lower-than-average label value (e.g., higher utility in the past) and a low Gini index for the group. The threshold value of the Gini index and label value is set empirically. The optimal sequence of decisions along a specific route can be found by starting from a successful end node and rolling backward to the root of the decision tree. In this study, we do not focus on the features and their values that can identify those winning stocks. The pool of features is chosen according to recommendations from the literature . This will be addressed in detail in the experiment section.

As mentioned, we need some overlap between different portfolio candidates, as it would not only address the agreements in selecting stocks among different investors but couldalso be useful in controlling transaction costs and slippage (Lo & Remorov, 2017). In addition to using a soft threshold value in each split at non-end nodes in each CART, we use multi-CART to form a bagging strategy that allows a parallel calculation to generate a number of homogeneous portfolio candidates. For the UCB algorithm, more candidates mean the algorithm has more options available. However, we ought to make sure all these options are satisfying. We, therefore, choose the best 30 portfolios for the candidates pool.

The flowchart of proposed tree based learning is shown in Fig. 1. In order to maintain high efficiency during the learning process, post-pruning (Ahmed et al., 2018) is performed on each non-terminal node.

3.2. *Contextual bandit searching*

A context-based UCB framework should be appropriate for addressing the challenges of online portfolio management. The optimality is defined with respect to the best achievable utility. The contextual information includes a desired return history of the portfolio(s) together with associated risk.

As one of the most popular extensions of the multi-armed bandit problem, the most basic example of contextual bandits is the Linear Upper Confidence Bound (LinUCB) algorithm, which has K arms that represent K portfolios and time steps $t = 1, 2, \ldots, M$, indicating that each portfolio k receives a return $R_k(t)$ at each time step $t = 1, 2, \ldots, M, k \in \{1, 2, \ldots, K\}$. The objective of bandit learning in this

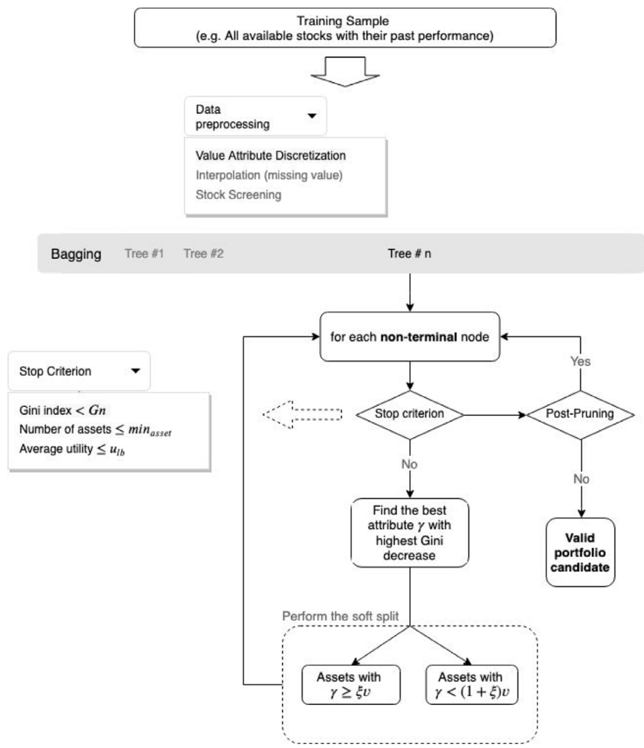

Fig. 1. The flowchart of proposed tree-based learning that generates portfolio candidates.

paper is to choose a series of portfolios, one for each time, with a minimized pseudo-regret after M periods defined as follows.

$$\frac{1}{M}\sum_{t=1}^{M}\left(R_{k_t^*}(t)-\frac{1}{K}\sum_{k=1}^{K}R_k(t)\right) \tag{1}$$

where k_t^* represents the best k among K portfolios at each time t. We also assume that the return $R_k(t)$ of each portfolio k linearly depends on d−dimensional features $X_{k,t}$, which can be assumed to be the returns of d component assets (i.e., different stocks) in portfolio k. The expected payoff of the portfolio returns can be shown as the linear combined returns of each asset in the past m days (i.e., the time interval used in this paper).

$$R_k(t) = E[r_k(t)|W_k] = \frac{1}{m}\left(\sum_{\tau=t-m}^{t} X_{k,\tau} W_k\right) \tag{2}$$

where W_k is a $d \times 1$ unknown weighting vector,[4] and the $X_{k,t}$ is an $m \times d$ return matrix that contains information about d component assets over the period m. Here, we assume that the weighting vector is time-invariant for a reasonably short period with a length of m, namely, the minimum forming period (MFP). The length of the MFP is adjustable according to market conditions and the preferences of investors. If the market conditions change dramatically, a short MFP would be appropriate. During volatile market circumstances, investors should not hold the same position for a long period of time. If the market is steady, as in the case of normal economic growth, investors should avoid frequently changing their positions, and, therefore, a longer MFP is preferable. However, the MFP should not be too long because market movements are not regular, and a long MFP means that huge compromises will have to be made to deal with the market's volatility. A short MFP can well reflect the recent changes in the market (at least for the chosen stocks).

Investors' risk preferences also vary and should be addressed in their portfolios. For example, an aggressive investor might want to review and/or update the portfolio more frequently than a relatively more conservative investor. Furthermore, due to risk preference, instead of maximizing the expected payoff, we should maximize utility, which includes both the expected return $R_k(t)$ and a risk-regulation term $\eta_\alpha Y_k$. Here, we assume that risk Y_k is independent for different portfolio k, although, in the long term, the stock prices will be subject to the same systematic/market risk, and in a relatively short period, the effects of diversification are not very significant. Following the setting of Zimin et al. (2014), which considers a risk measure that is a linear combination of the mean and variance, where both summands are of the same order. We replace $W_{k,t}^T \Lambda_k(t) W_{k,t}$ with $Y_k(t)W_{k,t}$, where $\Lambda_k(t)$ is the correlation matrix and the $\upsilon_k(t)$ are the values on its diagonal — the standard deviation of k stocks. This is a natural variant of the mean–variance optimization and is a continuous function of the mean and the variance. Here, $W_{k,t}$ is a fixed weight vector W_k over an MFP terminated at time t. The effects of diversification can be factored into the risk-regulation process.

$$\begin{aligned} U_{\alpha,k,t} = U[R_k(t), Y_k(t), \eta_\alpha] &= r_k(t)W_{k,t} - \eta_\alpha Y_k(t)W_{k,t} \\ &= \left(r_k(t) - \eta_\alpha Y_k(t)\right) W_{k,t} \\ &\stackrel{\text{def}}{=} D_{\alpha,k,t} W_{k,t} \end{aligned} \tag{3}$$

where η_α is a coefficient for the risk preference of investor α and is an empirical input. In most empirical applications (Bodnar et al., 2018; Zimin et al., 2014), the choice of risk preference is performed heuristically. In our study, since the nature of bandit learning is more aggressive than the risk aversion strategy, we fixed its value (i.e. $\eta_\alpha = 2$) before have tried a few different settings.[5] Since risk preference is fixed in our experiment, we therefore use $U_{k,t}$, $D_{k,t}$ instead of $U_{\alpha,k,t}$, $D_{\alpha,k,t}$. The expected payoff $R_k(t)$ in Eq. (2) is replaced by utility $U_{k,t}$. The pseudo-regret will accordingly be the difference between the actual averaged utility and the best attainable utility: $U_{k^*,t} = D_{k,t}W_{k^*,t}$ with the optimal $W_{k^*,t}$,

$$W_{k^*,t} = \arg\max_{W_{k^*,t}} \left(D_{k,t}W_{k,t}\right) \tag{4}$$

where $D_{k,t}$ is an $m \times d$ observation matrix representing the information matrix $X_{k,t}$, the optimal $W_{k^*}(t)$ with a unit limit $\sum W_k(t) = 1$ that could easily be estimated by linear programming or by using some empirical settings. In order to test the performance of our proposed model, we need the best attainable utility $U_{k^*,t}$ to calculate the largest pseudo-regret. At the beginning of learning, as each portfolio should have an initial weight setting for capital allocation, the $W_{k^*}(t)$ calculated by the data of the first MFP serves as the initial $W_k^*(0)$ for each k.

In fact, the performance of different assets could be very similar over a short period due to some common factors. For instance, two similar listed companies may respond similarly in their stock prices when they are influenced by the same impact. Their returns and changes of returns, as well as their risk preference, could be the same. Therefore, $D_{k,t}^T D_{k,t}$ may not be invertible. In order to avoid such an ill-posed problem, we use L_2 regularization (i.e., ridge regression).

[4] The weighting vector W_k for portfolio k may change subject to investors' trading activities. The change may happen aperiodically. During the short time period $t = 1, 2, \ldots, m$, the W_k is assumed to be a constant since frequent changes in investment weights would be inappropriate and incur higher transaction costs.

[5] We have tried a few different values of η_α (around the averaged value of 2). The experimental results show that the overall performance of the strategy with different risk reference settings can achieve satisfying returns against the benchmark index, although the averaged utility would be different. The purpose of this study is to verify that people with different risk attitudes can use this strategy.

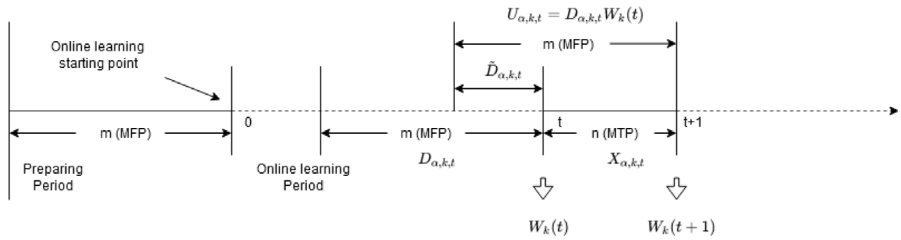

Fig. 2. The *time-line* alone which the proposed Contextual-Bandit Learning strategy chooses portfolio candidates.

Applying ridge regression to the given training input $(D_{k,t}, U_{k,t})$, where utility $U_{k,t}$ are the best possible utility conditioned on corresponding $D_{k,t}$ and weighting vector $W_k(t)$, we can define a pseudo-regret based loss function according to Eq. (3):

$$loss = (U_{k,t} - D_{k,t}W_k(t))^T(U_{k,t} - D_{k,t}W_k(t)) + \lambda W_k(t)^T W_k(t) \tag{5}$$

where λ is a Lagrange multiplier that gives an estimate of weighting vector

$$\hat{W}_k(t+1) = (D_{k,t}^T D_{k,t} + I_d)^{-1} D_{k,t}^T U_{k,t} \tag{6}$$

where I_d is a d−dimensional identity matrix, and $U_{k,t}$ is an $m \times 1$ vector. Eq. (3) can be treated as an estimation of utility $U_{k,t}$ of a selected portfolio k based on its current weighting vector $W_k(t)$ and market observation $D_{k,t}$. Therefore, the model input $U_{k,t}$ in Eq. (6) will contain some new information (e.g., future returns of component stocks in portfolio k). We assume that the new information is collected for n consecutive time steps after the point of training t, or we define n as the minimum testing period (MTP). Intuitively, the length of the MTP should be less than that of the MFP, $n < m$, as learning should be sufficient and use more information than testing. Therefore, the $m \times 1$ vector $U_{k,t}$ will contain all of the new information with the length of $n(n < m)$ steps (e.g., $D'_{k,t}$) together with some old information. For example, if $D_{k,t}$ is assumed to contain old information (e.g., $\tilde{D}_{k,t} \in D_{k,t}$) during a time period $\{t-m, t-m+1, \ldots, t-2, t-1\}$, and t is the starting point for testing, as well as for online learning, then $U_{k,t}$ will contain information during the time period $\{t+n-m, t+n-m+1, \ldots, t+n-2, t+n-1\}$.

The parameter estimation of ridge regression can be done by the Bayesian point estimate that the posterior distribution of the coefficient vector $p(W_k(t+1))$ should be Gaussian with mean $\hat{W}_k(t+1)$ and covariance $(D_{k,t}^T D_{k,t} + I)^{-1}$; so does the distribution of weighted utility.

$$\begin{gathered} p(W_k(t+1)) \sim N\left(\hat{W}_k(t+1), (D_{k,t}^T D_{k,t} + I_d)^{-1}\right) \\ p(X_k(t)W_k(t+1)) \sim N\left(X_k(t)\hat{W}_k(t+1), X_k(t)(D_{k,t}^T D_{k,t} + I_d)^{-1} X_k^T(t)\right) \end{gathered} \tag{7}$$

for any $\delta > 0$, the inequality follows could be a UCB for the expected payoff of portfolio k, with probability at least $1 - \delta$,

$$|X_k(t)\hat{W}_k(t+1) - E[U_{k,t}]| \le \beta\sqrt{X_k(t)(D_{k,t}^T D_{k,t} + I_d)^{-1} X_k^T(t)} \tag{8}$$

where $\beta = 1 + \sqrt{\ln(2/\delta)/2}$ is a constant,[6] which is a measure of optimism or how tight the UCB for the expected payoff of portfolio k.

The information contained in an $m \times d$ input matrix $D_{k,t}$ consists of portfolio returns, variance, and investors' risk preference during the last MFP, while the $n \times d$ matrix $X_k(t)$ has the information on recent returns, variance, and risk preference. The sample size n could be any value between 1 and m. The UCB portfolio selection strategy can be derived as follows:

$$k^* = \arg\max_{k \in K}\left(X_k(t)\hat{W}_k(t+1) + \beta\sqrt{X_k(t)(D_{k,t}^T D_{k,t} + I_d)^{-1} X_k^T(t)}\right) \tag{9}$$

Based on information $X_k(t)$ collected in time t, the optimal portfolio candidate k^* is selected in time $t+1$. The new portfolio will then be maintained for another n days, during which both the stocks in the portfolio k^* and stocks in other portfolios $k(k \neq k^*)$ will generate new returns and corresponding utilities. According to the proposed bandit learning algorithm, the parameter W, related upper bond values, and optimal k^* will also be re-estimated after each n times. Therefore, the algorithm is, indeed, an online learning approach (see Fig. 2).

The pseudo-code of contextual-bandit learning is illustrated as follows. The adjustable parameter η refers to the risk attitude of a given investor, and m, n are two time constants. In this study, we use $\eta = 2$, $m = 20$ and $n = 15$, which could generate fairly long-term good performance. In the experiment section, the backtesting[7] outcomes using data from the Chinese A share stock market have been displayed and analyzed.

Algorithm 1 Contextual-Bandit Learning Approach

```
Initialize:
Ŵ_k ← W*_k for each portfolio k                         {For the first m periods}
k* = arg max_{k∈K} D_k(t)W_k(t)
Main Loop:
while termination condition is not satisfied
for each k ∈ K
D_{k,t} = r_k(t) − ηυ_k(t)                              {η depends on investor}
X_k(t) ← Σ_{i=t}^{t+n} r_k(i) − ηυ_k(i)
U_{k,t} ← X_k(t) ∪ D̃_{k,t}                              {D̃ is last n-out-of-m previous D}
Ŵ_k(t+1) = (D^T_{k,t} D_{k,t} + I_d)^{-1} D^T_{k,t} U_{k,t}   {Update each candidate}
end for
k* = arg max_{k∈K} ( X_k(t)Ŵ_k(t+1) + β sqrt( X_k(t)(D^T_{k,t} D_{k,t} + I_d)^{-1} X^T_k(t) ) )
Record portfolio performance by candidate k*
t = t + 1
end while
```

3.3. Two-stage process

The model starts with a tree-based algorithm that is used to form a number of candidate portfolios based on various economic indicators and the market performance of individual stocks. The second stage of the model is to dynamically select a series of portfolios after each pre-fixed time interval based on LinUCB online searching. We update the pool of candidate portfolios each season (i.e., every three months) since the stocks we intend to invest in ought to have strong fundamentals. Every three months, their financial statements also provide

[6] The constant, which also refers to the Chernoff–Hoeffding Bound, is used to bound the probability that some function of many small random variables falls in the tail of their distributions or far from their expectations. The usefulness of the value has been empirically proved by many applications.

[7] The standard technique for determining how well a strategy or model would have performed ex-post is backtesting. Through the use of previous data, backtesting determines whether a trading technique is viable. If backtesting is successful, traders and analysts might feel confident using it in the future.

the most recent value for economic indexes. The most up-to-date value of economic indicators from their financial statements is also available every three months. In the second stage, the LinUCB learning approach is executed after every minimum testing period (MTP). The length of the MTP, as well as a few other parameters, will be valued on a trial-and-error basis and in terms of some empirical prior knowledge. The explanation in further detail can be found in Section 4's examination of our trials.

4. Data and experiments

4.1. Data and features

The economic prognosis of a particular industry, historical trade data on the assets, as well as thoroughly researched financial indicators in literature, are important aspects or features that contribute to the effectiveness of investments. These theoretically important features should be used in conjunction with the decision tree[8] to find high-performing stocks. Consistent with the literature, the features listed in Table 1 are included in our study.

These indicators are useful in measuring companies' short-term development and the trend of their stock prices. We investigate research reports produced by the top six securities in China: Guotai Junan Securities (www.gtja.com), Bohai Securities (www.ewww.com.cn), Soochow Securities (www.dwjq.com.cn), China Merchants Securities (www.newone.com.cn), Northeast Securities (www.nesc.cn), and Huatai Securities (www.htsc.com.cn). Among the seven influential indicators quoted in Table 1, four are financial quality indicators (quantity relative ratio, dividend yield ratio, net profit growth rate, and equity multiplier), and the other three are volume and price indicators (price earning ratio, net assets per share, turnover rate).

In the stock market, the most important indicator for investors is the stock return information, which is closely related to the follow-up market trend of the stock to a large extent. Volatility is another concern as it describes the risk of keeping expected returns. Therefore, we use the utility attributes defined in $E(r_k(t)) - \eta Y_k(t)$ as the label attribute of the sample set. Since the values of the daily return and its volatility over a given period are continuous, and the label attribute must be a discrete value, we need to discretize the utility attribute. We define the returns attribute by $r_k(t) = \log(P_k(t)/P_k(t-1))$, where $P_k(t)$ represents the closing price of stock k in period t. $E(.)$ and $Y_k(t)$ represent the average return and its standard deviation over 3 months. Since the value of the daily return is bounded in the Chinese stock market (i.e. the intraday trading of a given stock would be suspended for the rest of the day if its return reaches 10% or -10%.), the average return and standard deviation are also bounded. We, therefore, may rank the values of the utility of each stock in an ascending order and divide them into ten separate folders (i.e., $1, 2, \ldots, 10$) for their label attributes.

We build the portfolio and use a 15-day (MTP) adjustment scheme to update the weight online while backtesting. In the trial period from 2013 to 2020, the historical data will be backtested. We compare the model's performance with that of other benchmark portfolios in order to determine whether the model developed in this research is efficient and capable of mitigating the risk of market rotation . First, the CSI 300 index, which aims to reflect the overall performance of the China A-share market and consists of the 300 largest and most liquid stocks in terms of market capitalization.[9] Second, a historically optimal portfolio that performed the best in terms of returns and volatility during the previous MFP.

[8] The decision tree can also be used to determine how these important features relate to stock performance. This will be used in subsequent research because it is outside the purview of the current study.

[9] Since the index is composed according to the general performance of each constituent stock, we treat the value of the index as the value of the portfolio.

4.2. Modeling and results

According to the research design proposed by our algorithm, stocks in the candidates' pool are grouped/clustered (into several portfolios) by CART bagging. The chosen portfolios that satisfy the following criteria are kept. First, all of the stocks in the portfolio share a high degree of similarity in terms of both descriptive and label characteristics (e.g., level of utility). Second, each portfolio ought to be a winning combination when looking at past performance. That is, the stock portfolio with a bad track record should be pruned during the learning process. The maintained portfolio ought to have a low Gini index value, indicating that the majority of its stocks have similarly and consistently achieved profitable results . Third there is a predetermined amount of stock overlap/dislocation between portfolios.[10] This reduces some transaction costs and the slippage effect and gives the portfolios a certain amount of stickiness. The CSI 300's component stocks are used to create the stock samples. The WIND database and statistics from the literature are used to determine the descriptive characteristics of each stock sample.

During the learning process, we use a dislocation/overlap ratio parameter $\xi = 0.02$ and the maximum depth of each CART = 5. One of the trained CARTs is shown in Fig. 3. From these trained CARTs, we could find several portfolios with a lower Gini index value and a leading label larger than 5, which means that the majority of constituent stocks in the portfolio are outperforming (than the market average) stocks according to their recent utility.

Fig. 4 displays one of the outputs from a CART tree. Eight portfolios, each of which has a qualified Gini index and utility, were created by eight leaf nodes and are included in the output. The histograms in Fig. 4 show how the labels for each of the eight portfolios' stocks are spread out. The labels range from 1 to 10. If one label in a distribution has a higher proportion than all of the other labels in the distribution, we can always identify the leading label (i.e., the mode of labels). In a distribution or histogram with two leading labels, we randomly select one. In Fig. 4, the y-axis displays the number of stocks for each label, and the x-axis displays the labels' values, which range from 1 (the lowest percentile of utility) to 10 (the greatest percentile of utility). For instance, in Fig. 4, the portfolio D contains 16 stocks that have been labeled by 7 different values, and the stocks being labeled 7 have a larger amount/proportion than the other four (i.e., 1, 5, 6, 8, 9, 10). Therefore, we label portfolio D with 7. Likewise, in Fig. 4, we can label portfolio A with value of 7, portfolio B with 4, portfolio C with 8, portfolio E with 10, portfolio F with 8, portfolio G with 2 and portfolio with by 5. All candidate portfolios can, thus, be classified by a leading label. Meanwhile, because the constituent stocks will have similar values in the descriptive attributes defined in Table 1, as they have been classified into one leaf node of a CART, we believe these stocks should perform similarly as they have similar fundamentals.

Portfolio candidates for contextual bandit searching are then considered as the portfolios chosen by the ensemble CART. Fig. 5 illustrates the backtesting findings for the data from 2013 to 2020, which have been contrasted with the performance of the CSI 300 index. The setting of the risk-aversion parameter η_α will not significantly impact on the results of backtesting, though a lower value (less risk-tolerant) will result in less excess performance against the CSI 300, on average. The literature suggests the value of the Chernoff–Hoeffding Bound β, and the MTP is based on trial-and-error, as a large MTP will be less sensitive to market change, while a small MTP might incur unnecessary transaction costs by frequent trading.

There are different ratios/statistics/measures that investors use to measure the performance of their investments. We use three of the most popular measures : the Sharpe Ratio, the information ratio, and the maximum drawdown.

[10] The same stocks might exist in different portfolios.

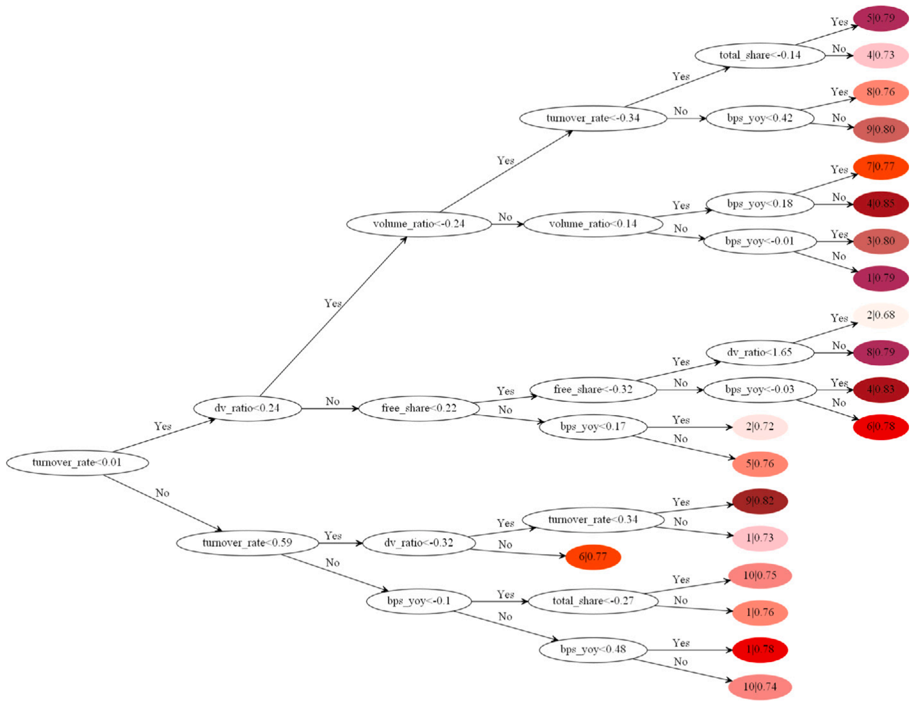

Fig. 3. The structure of a learned tree by CART algorithm: Each terminal leaf represents a portfolio; the digit on the left shows the leading label, and the number on the right is the Gini index (the darkness is proportional to the value of Gini index); The leading label and Gini index together show the quality of the portfolio.

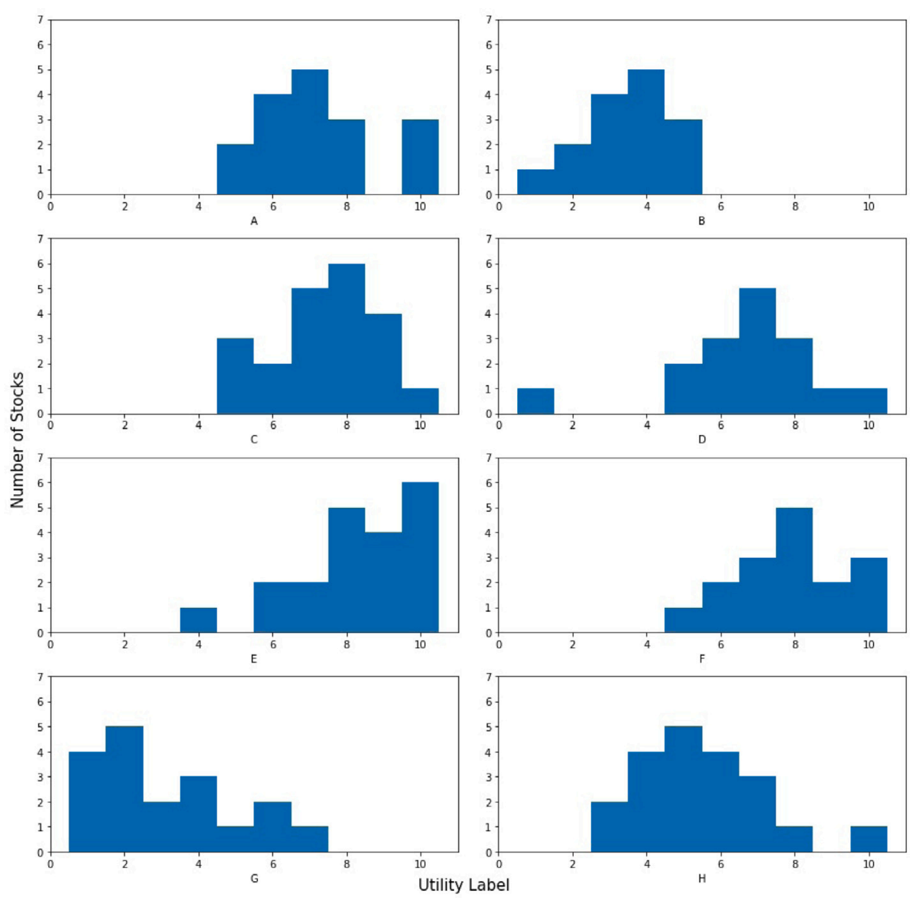

Fig. 4. Histogram of stock labels of *eight* portfolios. X-axis: value of labels ranked by percentile of utility (range from 1 to 10); Y-axis: number of stocks for each label.

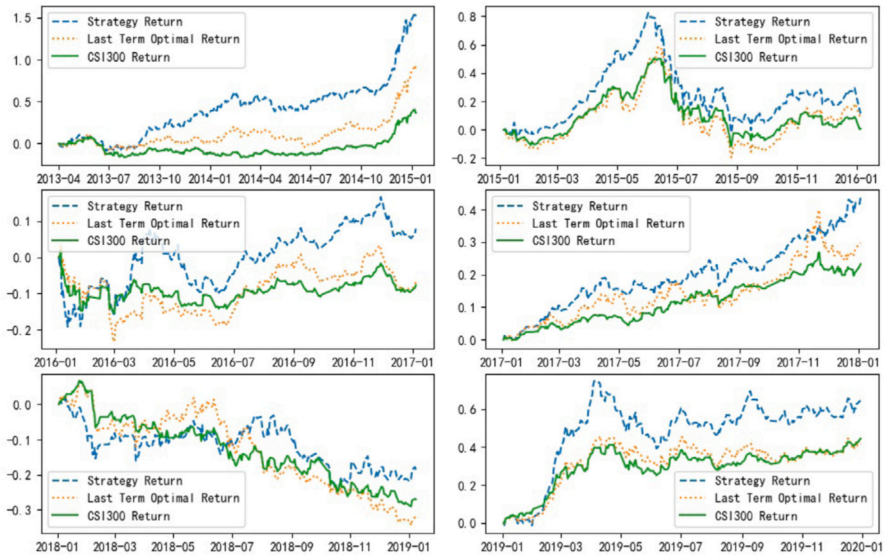

Fig. 5. Returns of the strategy selected portfolio, CSI300 index returns, and the returns of the best-performing portfolio of the last term in different years: The difference in performance between the strategy return and the last-term optimal returns represents the effectiveness of the bandit learning, which does make a better choice than the best-performing portfolio in the last term.

By using these three popular tests along with the return, shown in Table 2 and Fig. 5, we measure the performance of the benchmark, the proposed model, and the so-called last-term optimal return (i.e., At every decision time, instead of using LinUCB, we select the portfolio that achieves the highest utility in the last minimum form period.).

It can be seen that, on average, the proposed model can outperform the market average in the long run. From 2013 to 2014, we find that the net value of the learned portfolio could be as high as 1.461, and the income results show a state of continuous raising with a few small-scale fluctuations. Before June 2013, performance of the model was a bit weaker than that of the CSI 300 index. Thereafter, the learned model gradually started to show persistently better net income than the CSI 300 showed. From 2015 to 2016, the backtesting results show that the proposed model kept excessive returns compared to the CSI 300. From June 2015, due to the recession in the Chinese capital market, market returns showed a dramatic decline, while the proposed model resisted the risk of market turbulence. From 2017 to 2018, the backtesting results showed that the highest net value achieved by the algorithm was close to 40% — much better than the index markets' returns. The results from 2018 to 2019 showed that after the A-share market fell from the end of 2018 to the beginning of 2019, the indexes lost nearly 1/3 of their past values, causing investors to suffer significant losses. Although the algorithm proposed in this paper had some drawdowns, its total loss was much less than that of its benchmark. During the whole backtesting period, the adaptive bandit algorithm completes the portfolio formation and weighting, as well as the classic exploration and exploitation dilemmas. The algorithm achieves a good balance between profit and risk.

According to the overall backtesting evaluation, the cumulative yield of the selected portfolios can reach 2.630, with a 37.5% annual yield. The detailed results shown in Table 2 indicate that the proposed algorithm can not only achieve better returns but also provide effective risk control. The balanced combination by the algorithm can ensure excellent cumulative returns over various market periods.

We recorded the chosen portfolio and the corresponding weights in each learning period. The histograms of the portfolio returns and the highest possible returns are shown in Fig. 6. The blue column indicates the actual returns on the portfolio picked by each learning period of the algorithm, while the orange column shows the highest potential portfolio return after each selection. It is clear that the chosen portfolio initially underperformed (compared to the best-possible portfolio, which is the greatest attainable portfolio). This poor performance results from how the reinforcement learning algorithm is designed. We can anticipate higher performance as the algorithm progresses to a later phase.

Table 2
The performance of the strategy selected portfolios against other benchmarks.

	Ann. Return	Max. DD.(%)	Sharpe Ratio(%)	Info. Ratio(%)
	2013–2014			
Portfolio	0.731	17.84	2.256	2.442
Last Opt.	0.420	21.09	1.451	1.117
CSI300	0.180	22.28	0.912	–
	2015			
Portfolio	0.261	46.37	0.719	0.891
Last Opt.	0.177	49.05	0.549	0.598
CSI300	0.079	41.27	0.333	–
	2016			
Portfolio	0.052	18.77	0.245	1.148
Last Opt.	−0.093	25.78	−0.378	0.099
CSI300	−0.093	18.08	−0.528	–
	2017			
Portfolio	0.421	6.35	2.535	1.659
Last Opt.	0.279	1.85	1.495	0.555
CSI300	0.208	6.61	1.872	–
	2018			
Portfolio	−0.211	26.62	−0.923	0.687
Last Opt.	−0.326	37.69	−1.649	−0.357
CSI300	−0.280	32.98	−1.694	–
	2019			
Portfolio	0.646	20.42	2.052	1.159
Last Opt.	0.418	11.34	1.551	−0.078
CSI300	0.445	11.50	1.986	–

*Portfolio means the return of strategy selected portfolios by the proposed bandit learning. Last Opt. refers to the last-term optimal return, which is the return of the portfolio with the best performance (i.e., utility) in the last term. CSI300 is the CSI300 index return.

5. Conclusions and future works

This paper proposes a portfolio selection model based on an ensemble learning algorithm and a reinforcement learning algorithm.

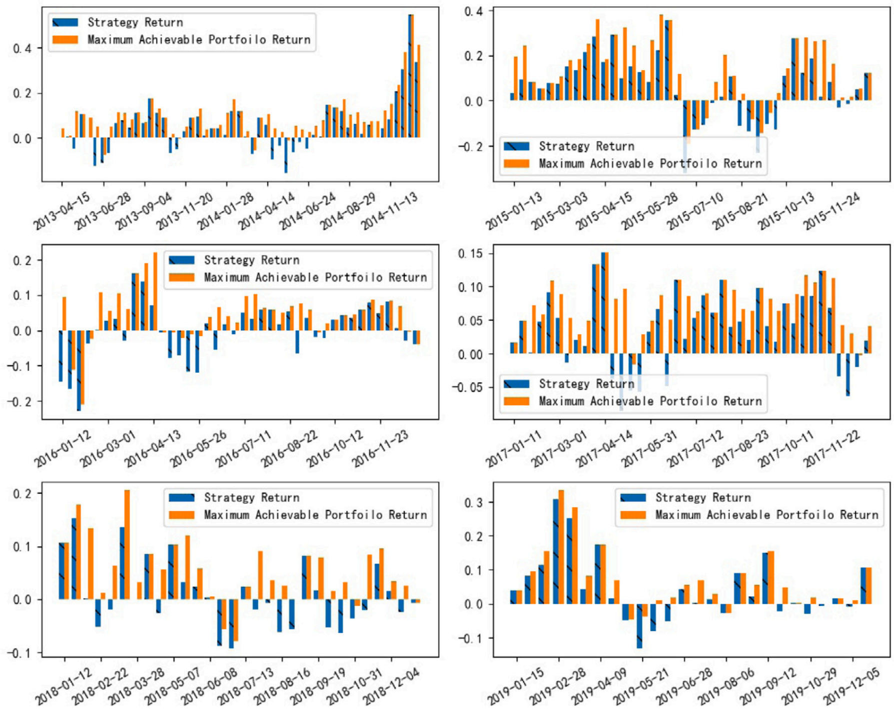

Fig. 6. Returns of the strategy returns and the maximum achievable portfolio returns in different years.

The efficiency of the proposed two stage model is evaluated using actual historical Chinese A-share stock market data. The following is a summary of the model's benefits and contributions .

Previous research on portfolio management is used to adopt a risk-averse utility, which presupposes that utility should be negatively impacted by return uncertainty. We believe that individuals can profit from exploring a small number of stocks with increased uncertainty regarding bandit learning. The algorithm keeps a sizable portion of investors' utility using the conventional definition of risk aversion while keeping a tiny portion of investors' utility using the risk-seeking definition.

The proposed pairing of two machine learning techniques can significantly outperform the market average, and this kind of risk-taking is effective in a variety of market scenarios. Besides, using nonlinear mapping or transfer in many applications of a machine learning algorithm on stock analysis may inevitably lead to a distortion of the original factors' or features' meaning. Such changes might have favorable outcomes in terms of average performance over the course of a particular testing period. Institutional investors, however, typically favor a model that can be understood. This study creates a pool of potential portfolio candidates using a decision tree and bandit selection. These arrangements made sure the final choice was completely interpretable.

Since portfolio management based on bandit learning is relatively new, there is still a lot to explore in the future. For example, we should take into account investor comments, adjustments to the candidate portfolio pool, the length of the minimum forming and testing periods, among others. The two stages might interact more. Thus how well bandit learning performs should have an impact on how best to identify a portfolio. For instance, the candidate pool should be modified by adding more candidate (lucky) portfolios if one candidate (lucky) portfolio has been chosen through bandit learning quite frequently. More work is needed to make the model more dynamic and adaptable to the fluctuating market environment. The model should be further developed to account for the fluctuating risk preferences of various investors.

CRediT authorship contribution statement

He Ni: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing. **Hao Xu:** Conceptualization, Methodology, Formal analysis, Investigation, Supervision. **Dan Ma:** Conceptualization, Methodology, Investigation. **Jun Fan:** Conceptualization, Supervision, Writing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: He Ni reports financial support was provided by Zhejiang Province Natural Science Foundation. Hao Xu reports financial support was provided by Zhejiang Province Natural Science Foundation.

Data availability

Data will be made available on request.

Acknowledgments

This research was financially supported by the Zhejiang Provincial Natural Science Foundation (LY20G010002, LQ22G030021), the Ministry of Education Humanities and Social Science Research Project (22YJA790042), the characteristic & preponderant discipline of key construction universities in Zhejiang province (Zhejiang Gongshang University- Statistics), Zhejiang Gongshang University "Digital+" Disciplinary Construction Management Project (SZJ2022B001) and the Tailong Finance School. The authors would also like to thank the data collection service provided by Hangzhou Woma Information Tech., as well as significant support from postgraduate students Mr. Yilai Liu, Miss Yan Fang and Mr. Zhaoyu Chen.

References

Abdellaoui, M. (2002). A genuine rank-dependent generalization of the von Neumann–Morgenstern expected utility theorem. *Econometrica*, *70*(2), 717–736.

Aboussalah, A. M., & Lee, C.-G. (2020). Continuous control with stacked deep dynamic recurrent reinforcement learning for portfolio optimization. *Expert Systems with Applications*, *140*, Article 112891.

Ahmed, A. M., Rizaner, A., & Ulusoy, A. H. (2018). A novel decision tree classification based on post-pruning with Bayes minimum risk. *PLoS One*, *13*(4), Article e0194168.

Almahdi, S., & Yang, S. Y. (2017). An adaptive portfolio trading system: A risk-return portfolio optimization using recurrent reinforcement learning with expected maximum drawdown. *Expert Systems with Applications*, *87*, 267–279.

Auer, P. (2002). Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, *3*, 397–422.

Awiagah, R., & Choi, S. S. B. (2018). Predictable or random?–A test of the weak-form efficient market hypothesis on the Ghana stock exchange. *Journal of Finance and Economics*, *6*(6), 213–222.

Bagella, M., Becchetti, L., & Adriani, F. (2005). Observed and "fundamental" price–earning ratios: A comparative analysis of high-tech stock evaluation in the US and in Europe. *Journal of International Money and Finance*, *24*(4), 549–581.

Bodnar, T., Okhrin, Y., Vitlinskyy, V., & Zabolotskyy, T. (2018). Determination and estimation of risk aversion coefficients. *Computational Management Science*, *15*(2), 297–317.

Buhmann, J. M., & Fischer, B. (2003). Bagging for path-based clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(11), 1411–1415.

Cen, L., Wei, K. J., & Yang, L. (2017). Disagreement, underreaction, and stock returns. *Management Science*, *63*(4), 1214–1231.

Chang, Y.-C., Chang, K.-H., & Wu, G.-J. (2018). Application of extreme gradient boosting trees in the construction of credit risk assessment models for financial institutions. *Applied Soft Computing*, *73*, 914–920.

Chen, M.-Y. (2011). Predicting corporate financial distress based on integration of decision tree classification and logistic regression. *Expert Systems with Applications*, *38*(9), 11261–11272.

Chourmouziadis, K., & Chatzoglou, P. D. (2016). An intelligent short term stock trading fuzzy system for assisting investors in portfolio management. *Expert Systems with Applications*, *43*, 298–311.

Dempster, M. A., & Leemans, V. (2006). An automated FX trading system using adaptive reinforcement learning. *Expert Systems with Applications*, *30*(3), 543–552.

Fianu, S., & Davis, L. B. (2018). A Markov decision process model for equitable distribution of supplies under uncertainty. *European Journal of Operational Research*, *264*(3), 1101–1115.

Garivier, A., & Moulines, E. (2011). On upper-confidence bound policies for switching bandit problems. In *International conference on algorithmic learning theory*. Springer.

Ge, L., Lin, T.-C., & Pearson, N. D. (2016). Why does the option to stock volume ratio predict stock returns? *Journal of Financial Economics*, *120*(3), 601–622.

Gorse, D. (2011). Application of stochastic recurrent reinforcement learning to index trading. In *19th European symposium on artificial neural networks*. ESANN.

Gutowski, N., Amghar, T., Camp, O., & Chhel, F. (2021). Gorthaur-EXP3: Bandit-based selection from a portfolio of recommendation algorithms balancing the accuracy-diversity dilemma. *Information Sciences*, *546*, 378–396.

Huo, X., & Fu, F. (2017). Risk-aware multi-armed bandit problem with application to portfolio selection. *Royal Society Open Science*, *4*(11), Article 171377.

Jordan, S. J., Vivian, A., & Wohar, M. E. (2017). Forecasting market returns: bagging or combining? *International Journal of Forecasting*, *33*(1), 102–120.

Karolyi, G. A., & Wu, Y. (2018). A new partial-segmentation approach to modeling international stock returns. *Journal of Financial & Quantitative Analysis*, *53*(2), 507–546.

Lavie, D., & Rosenkopf, L. (2006). Balancing exploration and exploitation in alliance formation. *Academy of Management Journal*, *49*(4), 797–818.

Lemmon, M. L., & Nguyen, T. (2015). Dividend yields and stock returns in hong kong. *Managerial Finance*, *41*(2), 164–181.

Liao, J.-J., Shih, C.-H., Chen, T.-F., & Hsu, M.-F. (2014). An ensemble-based model for two-class imbalanced financial problem. *Economic Modelling*, *37*, 175–183.

Liu, J., Lin, Y., Lin, M., Wu, S., & Zhang, J. (2017). Feature selection based on quality of information. *Neurocomputing*, *225*, 11–22.

Liu, H., & Song, B. (2017). Stock trends forecasting by multi-layer stochastic ANN bagging. In *2017 IEEE 29th international conference on tools with artificial intelligence* (pp. 322–329). Los Alamitos, CA, USA: IEEE Computer Society.

Lo, A. W., & Remorov, A. (2017). Stop-loss strategies with serial correlation, regime switching, and transaction costs. *Journal of Financial Markets*, *34*, 1–15.

Mama, H. B. (2018). Innovative efficiency and stock returns: Should we care about nonlinearity? *Finance Research Letters*, *24*, 81–89.

May, B. C., Korda, N., Lee, A., & Leslie, D. S. (2012). Optimistic Bayesian sampling in contextual-bandit problems. *Journal of Machine Learning Research*, *13*(1), 2069–2106.

O'Hara, H. T., Lazdowski, C., Moldovean, C., & Samuelson, S. T. (2000). Financial indicators of stock price performance. *American Business Review*, *18*(1), 90–100.

Ozturk, H., Namli, E., & Erdal, H. I. (2016). Modelling sovereign credit ratings: The accuracy of models in a heterogeneous sample. *Economic Modelling*, *54*, 469–478.

Phan, D. H. B., Sharma, S. S., & Narayan, P. K. (2015). Stock return forecasting: some new evidence. *International Review of Financial Analysis*, *40*, 38–51.

Sani, A., Lazaric, A., & Munos, R. (2012). Risk-aversion in multi-armed bandits. *Advances in Neural Information Processing Systems*, *25*.

Scott, S. L. (2010). A modern Bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*, *26*(6), 639–658.

Sharma, H., & Kumar, S. (2016). A survey on decision tree algorithms of classification in data mining. *International Journal of Science and Research (IJSR)*, *5*(4), 2094–2097.

Shen, W., Wang, J., Jiang, Y.-G., & Zha, H. (2015). Portfolio choices with orthogonal bandit learning. In *Twenty-fourth international joint conference on artificial intelligence*.

Shen, W., Wang, B., Pu, J., & Wang, J. (2019). The Kelly growth optimal portfolio with ensemble learning. In *Proceedings of the AAAI conference on artificial intelligence, vol. 33, no. 01* (pp. 1134–1141).

Shi, L., Xi, L., Hu, X., & Ma, X. (2009). Bagging of artificial neural networks for bankruptcy prediction. In *2009 international conference on information and financial engineering* (pp. 154–156). Los Alamitos, CA, USA: IEEE Computer Society.

Silva, N., Werneck, H., Silva, T., Pereira, A. C., & Rocha, L. (2022). Multi-armed bandits in recommendation systems: A survey of the state-of-the-art and future directions. *Expert Systems with Applications*, *197*, Article 116669.

Somvanshi, M., Chavan, P., Tambade, S., & Shinde, S. (2016). A review of machine learning techniques using decision tree and support vector machine. In *2016 international conference on computing communication control and automation* (pp. 1–7). IEEE.

Sun, S., Wang, S., & Wei, Y. (2018). A new multiscale decomposition ensemble approach for forecasting exchange rates. *Economic Modelling*, *81*, 49–58.

Sutton, R. S., Barto, A. G., & Williams, R. J. (1992). Reinforcement learning is direct adaptive optimal control. *IEEE Control Systems Magazine*, *12*(2), 19–22.

Suzuki, T., & Tanaka, K. (2014). Mean-variance portfolio model modified by nonlinear bagging predictors. *Journal of Signal Processing*, *18*(6), 283–290.

Vakili, S., & Zhao, Q. (2016). Risk-averse multi-armed bandit problems under mean-variance measure. *IEEE Journal of Selected Topics in Signal Processing*, *10*(6), 1093–1111.

Xia, Y., Liu, C., Li, Y., & Liu, N. (2017). A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring. *Expert Systems with Applications*, *78*, 225–241.

Xiao, X., & Kong, L. (2012). An empirical study on the relationship between turnover rate and stock returns in Chinese stock market. *Advances in Information Technology & Management*, *2*(1), 239–245.

Yadav, I. S., Pahi, D., & Gangakhedkar, R. (2022). The nexus between firm size, growth and profitability: new panel data evidence from asia–pacific markets. *European Journal of Management and Business Economics*, *31*(1), 115–140.

Yang, Y., & Jiang, J. (2016). Hybrid sampling-based clustering ensemble with global and local constitutions. *IEEE Transactions on Neural Networks and Learning Systems*, *27*(5), 952–965.

Yin, A. (2020). Equity premium prediction and optimal portfolio decision with Bagging. *The North American Journal of Economics and Finance*, *54*, Article 101274.

Zheng, K., & Wang, X. (2018). Feature selection method with joint maximal information entropy between features and class. *Pattern Recognition*, *77*, 20–29.

Zimin, A., Ibsen-Jensen, R., & Chatterjee, K. (2014). Generalized risk-aversion in stochastic multi-armed bandits. arXiv preprint arXiv:1405.0833.