Computational Economics (2022) 60:861–882
https://doi.org/10.1007/s10614-021-10169-8

Exploring Statistical Arbitrage Opportunities Using Machine Learning Strategy

Baoqiang Zhan[1] · Shu Zhang[2] · Helen S. Du[2] · Xiaoguang Yang[3]

Accepted: 17 July 2021 / Published online: 19 November 2021

Abstract

Arbitrage opportunity exploration is important to ensure the profitability of statistical arbitrage. Prior studies that concentrate on cointegration model and other predictive models suffer from various problems in both prediction and transaction. To prevent these problems, we propose a novel strategy based on machine learning to explore arbitrage opportunities and further predict whether they will make a profit or not. The experiment is conducted in the context of Chinese financial markets with high-frequency data of CSI 300 exchange traded fund (ETF) and CSI 300 index futures (IF) from 2012 to 2020. We find that machine learning strategy can explore more arbitrage opportunities with lower risks, which outperforms cointegration strategy in different aspects. Besides, we compare different algorithms and find that LSTM achieve better performance in predicting the positive arbitrage samples and obtaining higher ROI and Sharpe ratio. The profitability of machine learning strategy validate the mean reversion and price discovery function of asset price between spot market and futures market, which further substantiate the market efficiency. Our empirical results provide practical significance to the development of quantitative finance.

Keywords Statistical arbitrage · Cointegration · Machine learning · Opportunities exploration

✉ Baoqiang Zhan
2111708031@mail2.gdut.edu.cn

[1] School of Management, Harbin Institute of Technology, Harbin, China

[2] School of Management, Guangdong University of Technology, Guangzhou, China

[3] Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China

1 Introduction

With the rapid development of quantitative finance in today's stock markets, statistical arbitrage, one of the most popular way of algorithmic trading, has been widely used in both academia and industry (Chaboud et al., 2014). Originated from pairs trading, statistical arbitrage is a kind of market neutral strategy used to obtain excess profits within controllable risks. Specifically, investors can select a pair of assets from different markets which have high relation in historical prices, and then set up a series of trading rules for machines. Following the trading rules, the machine buys the declining-price assets and sell the increasing-price assets automatically if the price spread of paired assets deviates from historical averages and expands beyond the preset threshold (Gatev et al., 2006). Hence, the profits are made when the price spread returns to long term equilibrium.

In general, statistical arbitrage trading lies in two principles. The first one is mean reversion, which refers to a tendency of asset prices to fluctuate around the long term average and return to a trend path (Balvers et al., 2000). The principle of mean reversion guarantees the random walk and stationary trend of price spread between two assets and thus creating the arbitrage opportunities (Balvers et al., 2000; Garleanu & Pedersen, 2013). A large amount of evidence witnesses the mean reversion phenomena in stock markets (Fama & French, 1988; Chaudhuri & Wu, 2003), which at the mean time establishes the prerequisite and foundation for statistical arbitrage. The second principle is price discovery, which often straddles both the spot and futures markets and reveals the relative efficiency between these two markets (Chen & Gau, 2010). Price discovery largely reflects to what extend the information incorporated into prices, making it possible to explore the arbitrage opportunities, especially in the context of high frequency trading (Chakravarty et al., 2004; Brogaard et al., 2014).

Undoubtedly the introduction of short sales mechanism and the boosting of computing power facilitate the implement of statistical arbitrage. Yet it's still difficult to make profits from statistical arbitrage. The non-arbitrage principles shows that statistical arbitrage opportunities are transient and cannot last for long (Ross 1976). Though arbitrage opportunities are caused by markets frictions and generated from the prices deviations, they could also be easily erased by markets' self-restoring function, which makes them hard to explore. An extensive previous studies employ cointegration strategy to investigate the prices deviations for the purpose of exploring statistical arbitrage opportunities (Tsay, 1998; McMillan & Speight, 2006; Huck, 2015; Papantonis & Biktimirov, 2016). Nevertheless, trading delay and threshold determination error become the main obstacles in cointegration strategy. Regarding trading delay, basic cointegration model merely considers about timing problem. The fixed threshold is set in the model while the deal could only be made after the prices reach the threshold. There has a high probability to cause order execution delay and fail in the arbitrage (Abreu & Brunnermeier, 2002). As for threshold determination error, it's an essential component of algorithmic trading. However, the dynamic volatility of risks increases the difficulty of threshold estimation (Chiu & Wong, 2015). A

rational and precise threshold is dominant to arbitrage. Therefore, the threshold determination error affect the return of arbitrage to a greater or lesser extent.

Apart from cointegration strategy, there are also a lot of studies focus on statistical arbitrage using machine learning techniques, such as Kalman filter, neural networks, random forests and so on (De Moura et al., 2016; Krauss et al., 2017). Most of these studies try to predict the assets prices directly, using the prediction results to arbitrage and skipping the timing problem of the trading. However, price prediction is not the best choice. The efficient market hypothesis have already pointed out that assets prices are influenced by a variety of factors and appear in a random pattern, which makes it cannot be predicted with a high accuracy (Fama, 1970; Qian & Rasheed, 2007). Whereas a successful statistical arbitrage needs to be predicted precisely, predicting the occurrence of statistical arbitrage opportunities would be a better choice instead.

So, is it possible to predict the statistical arbitrage opportunities using machine learning strategy? If it is possible, would there be any differences in the performance of arbitrage Return of Investment (ROI) comparing the cointegration strategy with machine learning strategy? These are the main two questions that we are focus on this paper. To solve these two questions, we follow the framework of cointegration model and translate the arbitrage opportunity exploration problem into a prediction problem. Then we adopt a series of machine learning models (e.g. logistic regression, support vector machine, XGBoost, CNN and LSTM) to predict the trading signals, which include opening signal, closing signal and stop-loss signal. A complete signal cycle are regarded as a statistical arbitrage opportunities in our study. Therefore, the motivation that we propose the machine learning strategy in this study is to explore the statistical arbitrage opportunities as much as possible for the purpose of obtaining the highest low-risk profits and exceeding the traditional cointegration strategy in high-frequency trading.

The contributions of this paper are twofold. First, we develop a framework of arbitrage opportunities exploration using machine learning strategy. Based on this framework, we can predict the occurrence of arbitrage opportunities ahead of time, which provides essential buffer time for arbitrage trading. Second, we compare the performance of cointegration strategy and machine learning strategy from different aspects, including prediction accuracy and arbitrage profits. Our results shows that machine learning strategy have a lower risk and higher return than cointegration strategy. The results of this paper validate the sustainable profitability of statistical arbitrage and the practical feasibility of machine learning strategy, which also provide existence evidence of market efficiency and implications for quantitative finance.

The remainder of this paper is organized as follows. Section 2 presents the literature review of this study. Section 3 formulates the problem and display the framework of machine learning strategy. Section 4 illustrates the experimental performance of statistical arbitrage of both cointegration strategy and machine learning strategy. Finally the discussion and conclusions are summarized in Sect. 5.

2 Literature Review

2.1 Related Studies in Statistical Arbitrage

Hogan et al. (2004) first define statistical arbitrage as a zero cost, self-financing trading strategy in mathematics. More intuitively, statistical arbitrage is regarded as a risk-neutral transaction behavior, which pursues profit maximization without any preference of for risk seeking or risk averse. Current literature has proved the profitability of statistical arbitrage. For instance, Baker and Savasoglu (2002) state that a diversified strategy considering arbitrage positions produces an abnormal return of 0.6–0.9% per month over the period from 1981 to 1996. Broussard and Vaihekoski (2012) also find that the annualized return of pairs trading can be as high as 12.5% and the profits are not related to market risk. As for the risk, arbitrage alleviates the investment restrictions and improve the transfer of risk among investors (Basak & Croitoru, 2006).

However, the competition are increasingly fierce as more and more arbitrageurs surging into the markets, which reduces the profitability of statistical arbitrage, albeit at a declining rate (Do & Faff, 2010). Besides, Kozhan and Tham (2012) point out that arbitrage can be a very risky business because the execution risk increases as the number of competing arbitrageurs increases. Attari et al. (2005) also confirms that the strategic trading in markets with large arbitrageurs produce significant price distortions and increase price manipulation, leading the arbitrageurs with financial constraints to make less profits in markets. Therefore, for most individual arbitrageurs without enough capital, the trading strategy is relatively more important. To some extent, the efficiency of trading strategy determines the possibility of successful arbitrage and profits making (Neely & Weller, 2013; Balvers et al., 2020) find that parametric contrarian investment strategies with fully exploit of mean reversion across national indexes outperform buy-and-hold and standard contrarian strategies. Chiu and Wong (2015) also design the optimal dynamic trading strategy through cointegration analysis and find their strategy generates higher Sharpe ratio that Black-Scholes model. Yet most previous studied are limited to the traditional econometric models (Ahn et al., 2002), whereas more efficient models are needed considering the gradually prevalence of algorithmic trading in the current market.

2.2 Machine Learning Enhanced High Frequency Trading

Given the arbitrage strategy, the cointegration model is frequently concerned in previous studies (Jorda & Taylor, 2012; Chiu & Wong, 2015). Such kind of traditional way of arbitrage is losing it effect in exploring statistical arbitrage opportunities. With the fast development in FinTech (Financial Technology) domain, using machine learning technologies to solve the financial problems becomes ubiquitous. For instance, Krauss et al. (2017) integrate the neural networks, gradient-boosted-trees and random-forests to predict the probabilities of arbitrage return on S&P 500. Huck (2019) considers around 600 predictors and use three

methods (e.g. random forest, deep belief networks, elastic net regression) to test the feature importance in contributing the prediction of excess return. Though hundreds of factors are informative enough, they also cause noise in predicting the trading signals. Schnaubelt et al. (2020) extract the relationship between features and subsequent returns using feature engineering in a low signal-to-noise setting, and find that the selected features are more interpretable in arbitrage.

The use of machine learning models in statistical arbitrage makes a signification step forward in algorithmic trading. However, most of these studies try to predict the asset price and return directly, which causes the threshold determination errors when implementing the trading strategy. Since the asset mispricing results in price discrepancy and generates the statistical arbitrage opportunities (Schultz & Shive, 2010), predicting the occurrence of arbitrage opportunities would be a better choice instead. Given the circumstance of increasing arbitrageurs and fierce competition in the market, exploring the arbitrage opportunities faster and more accurate is quite essential to make profits in algorithmic trading. Therefore, an efficient machine learning strategy of statistical arbitrage is required. Yet it still remains as a gap in existing literature.

3 The Framework of Statistical Arbitrage

In this section, we propose the framework of statistical arbitrage. First we introduce the cointegration strategy, and illustrate the structure of machine learning strategy, especially the component of how to predict the arbitrage opportunities. Figure 1 shows the framework of this study.

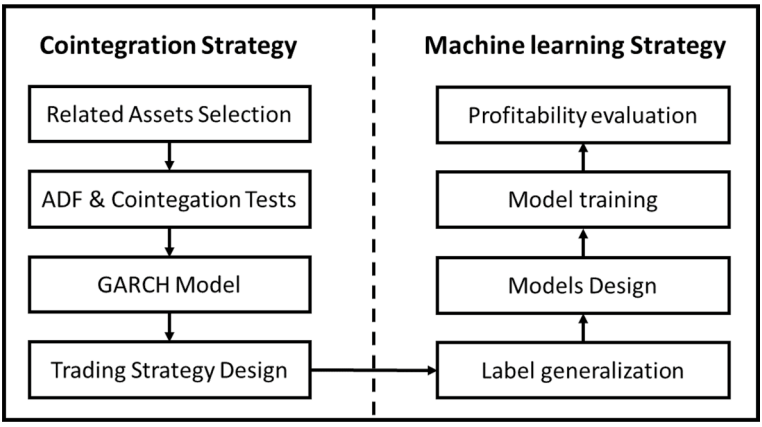

Fig. 1 The framework of statistical arbitrage

3.1 Cointegration Strategy

Based on mean reversion principle, cointegration strategy need to first select two assets whose prices have moved together historically (Gatev et al., 2006). Then, ADF test and cointegration test are carried out to exam the stationarity and cointegration relation of two assets. The third step is to analyze the dynamic volatility according to the spread series of two assets. Finally, determining the arbitrage threshold and generating the trading signals for opening, closing and stop-loss.

3.1.1 Related Assets Selection

We denote the two assets price as $\{E_t\}$ and $\{F_t\}$, where $t \in [0, T]$. In statistical arbitrage, a significant relationship between $\{E_t\}$ and $\{F_t\}$ is required. Therefore, we use the correlation coefficient r to measure the similarity of historical price trend of two assets.

$$r = \frac{\sum_{t=1}^{n} (E_t - \bar{E})(F_t - \bar{F})}{\sqrt{\sum_{t=1}^{n} (E_t - \bar{E})^2 \sum_{t=1}^{n} (F_t - \bar{F})^2}} \tag{1}$$

where $\bar{E}$ and $\bar{F}$ mean the average of $\{E_t\}$ and $\{F_t\}$ respectively. The greater of r indicates a higher similarity and relationship between these two assets. Only two assets with high correlation coefficient would be selected.

3.1.2 ADF test and cointegration test

Augmented Dickey-Fuller (ADF) test is a classical method to examine the stationarity of residual series between two assets price series. Cointegration test is used to check the long term equilibrium and stability of two assets price series (Kao, 1999). Considering the regression model (2), which is a $AR(p)$ process:

$$y_t = \sum_{i=1}^{p} \phi_i y_{t-i} + \omega_t \tag{2}$$

We test the statistics of $\tau = \hat{\rho}/S(\hat{\rho})$, where $\rho = (\phi_1 + \phi_2 + \cdots + \phi_p - 1)$ and $S(\cdot)$ denotes the standard error. If statistic τ is not significant for series $\{y_t\}$ but significant for the differential series $\{\Delta y_t\}$ instead, it means that $\{y_t\}$ is non-stationary and is integrated of order 1.

For cointegration test, we regress the two assets price series:

$$\ln F_t = \alpha + \beta \ln E_t + \varepsilon_t \tag{3}$$

We test the stationarity of residual series $\{\varepsilon_t\}$, which is also regarded as the price spread series. There would exist cointegration relationship between $\{\ln F_t\}$ and $\{\ln E_t\}$ if $\{\varepsilon_t\}$ is stable.

3.1.3 Dynamic Volatility Analysis Based on GARCH Model

Measuring the dynamic volatility of price spread is a key step to ensure the implement of statistical arbitrage as the arbitrage opportunities are explored in the price spread interval. Generally, the volatility of residual series is stable in most periods, yet it also generates heterogeneity sometimes, showing a combined effect. Such kind of effect is informative. Therefore, we adopt Generalized Autoregressive Conditional Heteroscedasticity (GARCH) model to capture the informative volatility of the residual series. The GARCH model is an extension of model (3):

$$\begin{cases} \ln F_t = \alpha + \beta \ln E_t + \varepsilon_t \\ \varepsilon_t = \sqrt{\sigma_t^2} \cdot e_t \\ \sigma_t^2 = \varphi + \sum_{i=1}^{q} \theta_i \epsilon_{t-i}^2 + \sum_{j=1}^{p} \gamma_i \sigma_{t-i}^2 \end{cases} \tag{4}$$

where p and q are autocorrelation and partial-correlation orders of heteroscedasticity. Based on $GARCH(p, q)$ model, we can calculate the dynamic volatility $\left\{\sigma_t^2\right\}$ and use it to determine the trading threshold.

3.1.4 Trading strategy design

The trading signals, including opening, closing and stop-loss, are determined based on the dynamic volatility of price spread. The dynamic volatility indicates the degree that price spread deviated from the long-term average. According to mean reversion, the price spread will finally converge to its average, from where the arbitrage profits are made. Let $\left\{mspread_t\right\}$ represents the decentralized price spread. Specifically,

$$mspread_t = \varepsilon_t - \bar{\varepsilon} - c \tag{5}$$

where $\bar{\varepsilon}$ is the average of price spread, and c is the trading cost. We set the trading costs here to further constrain the arbitrage interval. Table 1 displays the trading rules.

As illustrated in Table 1, λ_1, λ_2 and $\lambda_3 (0 < \lambda_2 < \lambda_1 < \lambda_3)$ are set as the thresholds for opening, closing and stop-loss, respectively. Here, we define a complete trading interval as an arbitrage opportunity since the statistical arbitrage behavior is only conducted after receiving trading signals. Figure 7 shows the trading interval of

Table 1 Trading rules setting

Trading interval	Trading strategy	Trading signal
$\lambda_1\sigma_t < mspread_t < \lambda_3\sigma_t$	Buying the assets with lower price and shorting the other	Opening
$-\lambda_3\sigma_t < mspread_t < -\lambda_1\sigma_t$	Buying the assets with higher price and shorting the other	Reverse opening
$-\lambda_2\sigma_t < mspread_t < \lambda_2\sigma_t$	Closing the current positions	Closing
$\lvert mspread_t \rvert > \lambda_3\sigma_t$	Closing the current positions	Stop-loss

statistical arbitrage. Besides, to prevent the threshold determination error, we optimize these three thresholds by evaluating the ROI under each situation. The optimization results are shown in Fig. 6.

3.2 Machine Learning Strategy

Compared with co-integration model, machine learning model is superior in its end-to-end structure that fully captures the latent feature of price spreads and makes prediction, regardless of threshold determination errors and other problems. On the basis of co-integration model, we first state clearly of the prediction problem and label the data for detecting the arbitrage opportunities. Then different machine learning algorithms (e.g. Logistic Regression, SVM, XGBoost, CNN and LSTM) for data training and prediction are introduced. Finally we use the metrics to evaluate the predictive performance and earnings performance of machine learning strategy.

3.2.1 Arbitrage Opportunities Labels

Based on supervised learning framework, machine learning algorithms requires the data input and output. Let the input $\{x_t\}$ represents the concatenate set of prices series of assets $\{E_t\}$ and $\{F_t\}$, that is $\{x_t\} = \{E_t, F_t\}$. Let the output $\{y_t\}$ denotes the detecting result of arbitrage opportunities, shown as follows.

$$y_t = \begin{cases} 1, & mspread_t \in \left(-\lambda_3\sigma_t, -\lambda_1\sigma_t\right) \cup \left(\lambda_1\sigma_t, \lambda_3\sigma_t\right) \\ 0, & mspread_t \in \left(-\lambda_2\sigma_t, \lambda_2\sigma_t\right) \\ -1, & mspread_t \in \left(-\infty, -\lambda_3\sigma_t\right) \cup \left(\lambda_3\sigma_t, \infty\right) \end{cases} \tag{6}$$

In Eq. (6), the output $\{y_t\}$ has three labels, which are corresponding to three trading signals of opening, closing and stop-loss. Specifically, label "1" indicates the occurrence of arbitrage opportunity. Therefore, it's recommended to take opening actions. Label "0" means that there has no arbitrage opportunities and closing actions should be taken. As for label "− 1", the $mspread_t$ is far beyond the threshold, implying that the risk is relatively high. It would lead to losses if the arbitrage is conducted at this moment and the stop-loss behavior has to be taken. Using the labels to indicate the trading signals of opening, closing and stop-loss, the arbitrage opportunities exploration are translated into a multi-classification problem, which provides a practical setting for machine learning strategy in statistical arbitrage.

3.2.2 Machine Learning Algorithm

Arbitrage opportunities exploration is a multi-classification task. For such kind of task, prior studies have used various supervised learning algorithms, which can be divided into three main streams. The first stream is traditional machine learning models, such

as logistic regression (Barboza et al., 2017) and support vector machine (Patel et al., 2015). These models are mainly used as baseline models because of their simple structure and fast computing. However, such characteristics also limit their performance in prediction accuracy. The second stream is ensemble tree model. XGBoost (Extreme gradient boosting method), one of the most typical models in ensemble methods, is widely used in different tasks (Basak et al., 2019; Chatzis et al., 2018). Based on the idea of boosting, XGBoost constructs a strong learner by weighted aggregating the weak learners. The mechanism of dropping the learners with bad performance and retaining those with good performance has decreased the training loss and has improved the accuracy to a great extent for XGBoost. Neural networks are the third main stream of machine learning technologies. At present, there are a variety of variants of networks, such as CNN, RNN and LSTM. Long-Short Term Memory (LSTM) is particularly designed to deal with time series data. Therefore, we mainly focus on the LSTM networks in this study.

The basic unit of LSTM is composed of a memory cell c_t and the three gates, which refer to update gate Γ_μ, forget gate Γ_f and output gate Γ_o. Basically, the memory cell is used as a transfer station to store the information from the last state and pass it to the next state. The three gates have different functions and effects. For instance, update gate is used to choose and update the important information, forget gate is used to filter and forget the insignificant information and output gate is used to aggregate the useful preprocessed information and pass it to the next memory cell. Let's denote x_t and a_t as the input and output of the memory cell c_t. The internal structure of LSTM unit is shown in Fig. 2.

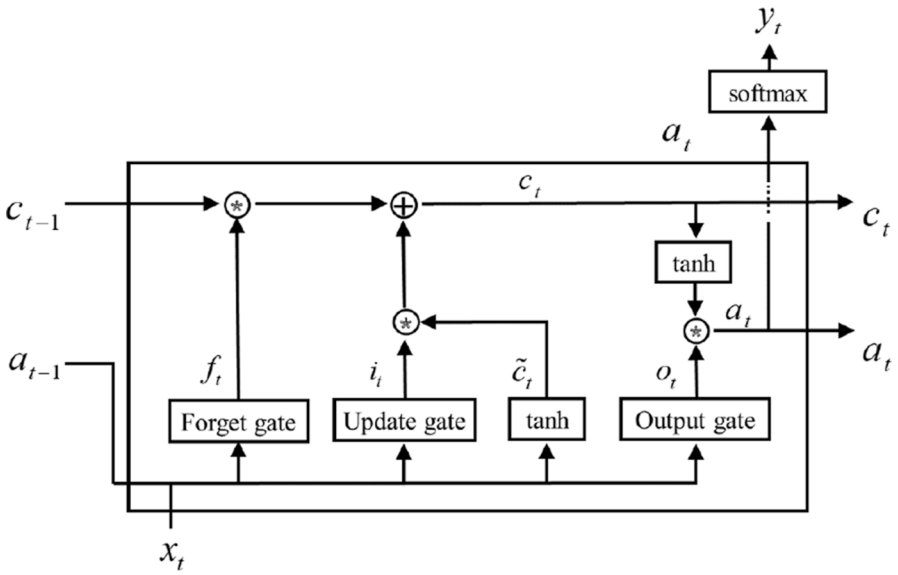

Fig. 2 Internal structure of LSTM unit

3.2.3 Evaluation Metrics

The main focus of our study is arbitrage opportunity exploration and arbitrage profit analysis. Therefore, we adopt two types of metrics to evaluate the performance of machine learning strategy. First, we adopt the widely accepted classification metrics: accuracy, precision, recall and f-measure to evaluate the predictive power of the machine learning model. Second, we use the return of investment (ROI) and Sharpe ratio to evaluate the profitability of machine learning strategy.

The accuracy, precision, recall and f-measure are defined as follows.

$$accuracy = \frac{tp + fn}{tp + tn + fp + fn} \tag{7}$$

$$precision = \frac{tp}{tp + fp} \tag{8}$$

$$recall = \frac{tp}{tp + fn} \tag{9}$$

$$f1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{10}$$

where tp and fp are the positive samples predicted as true and false respectively, tn and fn are the negative samples predicted as true and false respectively. The *precision* and *recall* measure the correctness and coverage of prediction results. The $f1$ is the harmonic mean of *precision* and *recall* while the *accuracy* is the overall performance measure of the model.

The ROI and Sharpe ratio are defined as follows.

$$r_t = \Delta E_t + \Delta F_t \tag{11}$$

$$sr = \frac{\bar{r} - r_f}{\hat{\sigma}_r} \tag{12}$$

where ΔE_t and ΔF_t is first order difference of price at time t and $t-1$ for two assets, which measures the incremental profits of arbitrage at time t. Besides, $\bar{r}$ and $\hat{\sigma}_r$ indicate the average profits and risks of the trading periods. r_f is the risk-free return rate. Sharpe ratio measures the extent to which the return can be obtained at a certain risk (Balvers et al., 2000).

4 Experiment

Following the statistical arbitrage framework of cointegration strategy and machine learning strategy, the experiment is conducted and the empirical results are displayed in this section. At the beginning, we introduce the whole dataset and its descriptive performance, as well as the data preprocessing. Then we implement the cointegration strategy and machine learning strategy simultaneously. Finally we compare the difference between these two strategies through profit analysis and explain the cause of return of statistical arbitrage.

4.1 Dataset

The dataset includes the high frequency price series of two assets in Chinese financial markets. One is Chinese Security Index (CSI) 300 index futures from futures markets, which is denoted as $\{F_t\}$. The other is Huatai-PineBridge CSI 300 index Exchange Traded Fund (ETF) from spot markets, denoted as $\{E_t\}$. The time range covers from May 28th, 2012 to Dec 30th, 2020, nearly 8 trading years. Therefore, the whole dataset contains 100,512 samples, which are all 5-min level high frequency data. Figure 3 plots the price series of these two assets.

The subject matter of these two assets are CSI 300 stocks. Therefore, the trends of price series are quite similar in Fig. 3. We calculate the correlation coefficient using Eq. (1), which is highly up to 99.33%. This is showing that these two assets are highly correlated, which satisfies the prerequisite of cointegration strategy.

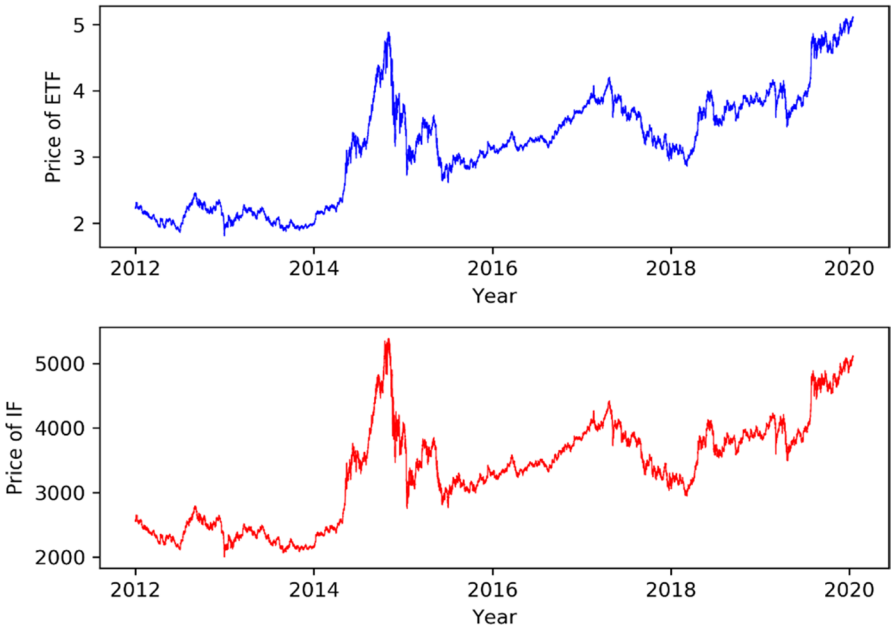

Fig. 3 Price series of asset $\{E_t\}$ and$\{F_t\}$

Table 2 Descriptive Statistics of asset $\{E_t\}$ and $\{F_t\}$

Year	Exchange traded fund $\{E_t\}$					Index futures $\{F_t\}$				
	N	Min	Max	Mean	SD	N	Min	Max	Mean	SD
2012	7200	1.866	2.316	2.069	0.103	7200	2113.2	2653.4	2348.316	122.498
2013	11,424	1.809	2.463	2.182	0.110	11,424	1999.6	2796.4	2438.733	146.639
2014	11,760	1.877	3.239	2.159	0.296	11,760	2066.2	3590.4	2372.833	326.118
2015	11,712	2.725	4.884	3.599	0.495	11,712	2755.2	5389.8	3894.012	584.398
2016	11,712	2.612	3.445	3.019	0.154	11,712	2765.2	3650.0	3204.218	149.583
2017	11,712	3.090	4.076	3.489	0.257	11,712	3266.0	4268.4	3663.448	252.050
2018	11,664	2.886	4.202	3.466	0.338	11,664	2956.8	4420.0	3599.627	379.615
2019	11,712	2.862	4.045	3.673	0.261	11,712	2948.0	4127.6	3748.926	253.481
2020	11,616	3.449	5.112	4.348	0.477	11,616	3489.2	5118.4	4365.093	454.874
All	100,512	1.809	5.112	3.159	0.003	100,512	1999.6	5389.8	3336.036	764.817

4.1.1 Descriptive Performance

The descriptive statistics are shown in Table 2. To show more details of the data, we statistic the data by years.

We find an upward trend in both assets on the whole. However, there is a dramatic decline in 2016, showing a great recession of the market. The overall statistics are consistent with the trend plotted in Fig. 3. Besides, the scale of F_t is much larger than E_t. Therefore, we use logarithmic form to eliminate the scale difference of E_t and F_t in the following models.

4.1.2 Data Structure

For the sake of model validation and generalization, the whole dataset is divided into training set and test set. Specifically, the data from 2012 to 2017 is used for training while the data from 2018 to 2020 is used as test set. Moreover, we apply a rolling-window prediction approach to fully capture the temporal relation of the asset prices. Since our data is 5-min level high frequency price series, we use the previous 12 price values $\{x_{t-12}, x_{t-11}, \ldots, x_{t-1}\}$ to predict the next label $\{y_t\}$. The window widths are 12 intervals of 5-min, which equals an hour long. Figure 4 shows the rolling window design of dataset.

The intuitive understanding of rolling window approach is that we are using the price trend of past an hour to predict the status of next 5 min. Therefore, we can extract the latent temporal characteristics from the assets price, and use it to predict the arbitrage opportunities, which highly rely on the price trend.

4.2 Cointegration Analysis

In this part, we first conduct the cointegration analysis on two assets. The cointegration analysis is composed of three parts. First is the ADF and cointegration

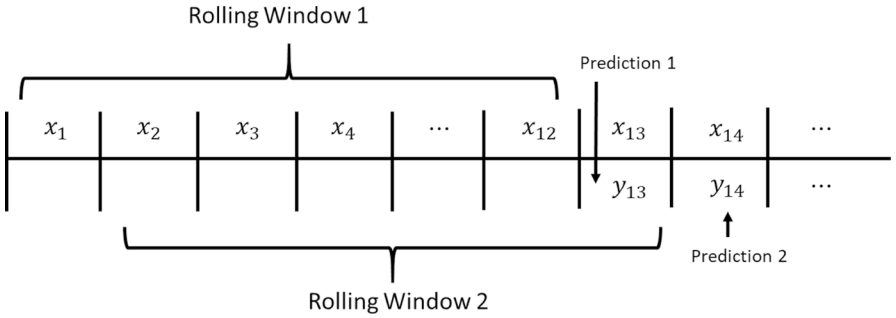

Fig. 4 Rolling window Procedure

test. Second is the dynamic volatility analysis based on GARCH model. Finally is the threshold optimization and trading interval partition.

4.2.1 ADF Test and Cointegration Test

As we mentioned before, ADF test is used to check the existence of unit root and examine the stationarity of prices series. Meanwhile, cointegration test is employed to examine the long term equilibrium relations of two assets. Table 3 shows the ADF test results.

As shown in Table 3, we conduct ADF test on price series of $\ln E_t$, $\ln F_t$, $\Delta\ln E_t$, $\Delta\ln F_t$, where ln denotes the logarithm and Δ means first order differential operation. Apparently, we can see that the P values of $\ln E_t$ and $\ln F_t$ are not significant while that of $\Delta\ln E_t$ and $\Delta\ln F_t$ are significant at 1% level. This indicates that the price series $\ln E_t$ and $\ln F_t$ are stationary series and are integrated of order one.

As for cointegration test, we first build the regression model on $\ln E_t$ and $\ln F_t$ and estimate the coefficients as follows.

$$\ln F_t = 7.100 + 0.883\ln E_t + \varepsilon_t \tag{13}$$

Then we test the stationarity of residual series $\{\varepsilon_t\}$ using ADF test again. The ADF test statistic is -4.243 (significant at 1% level), which indicates that $\{\varepsilon_t\}$ is stationary. Thus, there is significant cointegration relationship between $\ln E_t$ and $\ln F_t$, where the cointegration coefficient is 0.883.

Table 3 ADF test

Price series	ADF statistic	Test critical values			P values	Stationary
		1% level	5% level	10% level		
$\ln E_t$	− 0.568	− 3.430	− 2.861	− 2.567	0.875	No
$\ln F_t$	− 0.974	− 3.430	− 2.861	− 2.567	0.765	No
$\Delta\ln E_t$	− 115.328	− 3.430	− 2.861	− 2.567	0.000***	Yes
$\Delta\ln F_t$	− 118.548	− 3.430	− 2.861	− 2.567	0.000***	Yes

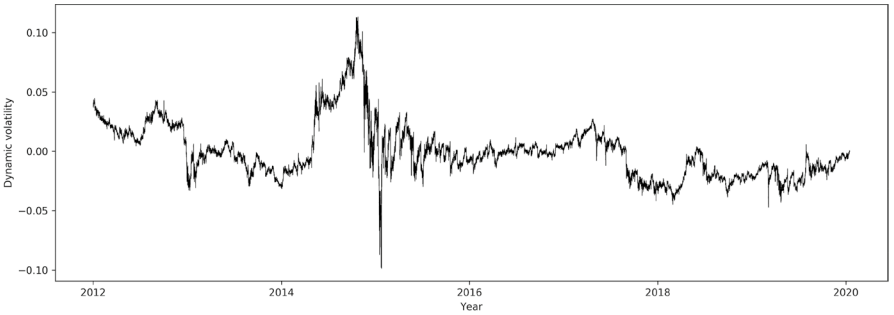

Fig. 5 The residual trend of asset $\ln E_t$ and $\ln F_t$

4.2.2 GARCH model

Before calculating the dynamic volatility using GARCH model, we have to test whether the residual series $\{\varepsilon_t\}$ has conditional heteroscedasticity. We plot the residual series $\{\varepsilon_t\}$ in Fig. 5.

From Fig. 5, we observe an extreme fluctuation and combined effect during 2014–2015. In other years, the fluctuations are relatively small, which implies that conditional heteroscedasticity exists in $\{\varepsilon_t\}$.

Therefore, we estimate the orders of autocorrelation and partial correlation and build the $GARCH(p, q)$ model to calculate the conditional heteroscedasticity $\{\sigma_t\}$. The orders of autocorrelation and partial correlation are estimated through ARCH-LM test, whose results are shown in Table 4.In table, we use the Akaike Info Criterion (AIC) and Hannan-Quinn (HQ) criterions to determine the orders of autocorrelation and partial correlation. As shown, the AIC and HQ are smallest when p and q are both equal 1. Therefore, we estimate the $GARCH(1,1)$ model and obtain:

$$\sigma_t^2 = 0.0000001 + 0.453\varepsilon_{t-1}^2 + 0.548\sigma_{t-1}^2 \tag{14}$$

Table 4 Autocorrelation and partial correlation orders estimation

p	q	AIC	HQ
1	1	− 6.200456	− 6.20037
2	1	− 6.201637	− 6.201522
1	2	− 6.201103	− 6.200988
2	2	− 6.202379	− 6.202235
3	1	− 6.201967	− 6.201823
3	2	− 6.202454	− 6.202282
3	3	− 6.200657	− 6.200456
1	3	− 6.201363	− 6.201219
2	3	− 6.202443	− 6.202271

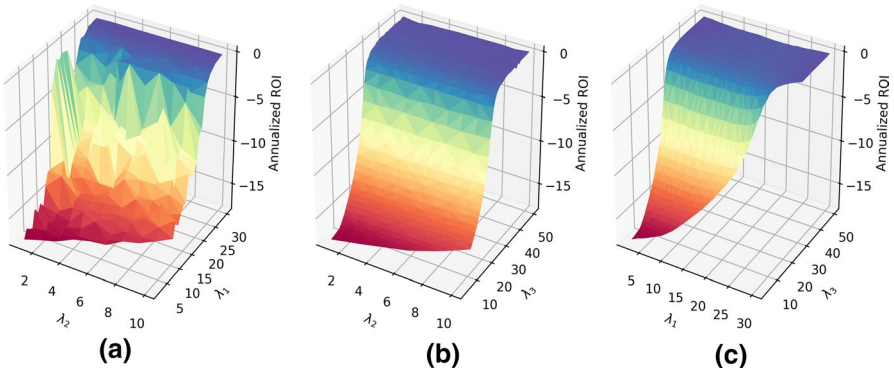

Fig. 6 Arbitrage Annualized ROI

From the regression model (14), we obtain the dynamic volatility (conditional heteroscedasticity), which is also used to determine the arbitrage trading signals.

4.2.3 Threshold optimization

An essential problem that we mentioned above is threshold determination. As Table 3 shown, the thresholds of λ_1, λ_2 and $\lambda_3 (0 < \lambda_2 < \lambda_1 < \lambda_3)$ are corresponding to the trading signal, closing signal and stop-loss signal. Appropriate and accurate thresholds contribute to the maximization of arbitrage profits. Therefore, to prevent the threshold determination error, we use the ROI analysis to optimize λ_1, λ_2 and λ_3. Concretely, we set the ranges for each threshold as: $\lambda_2 \in [0,10]$, $\lambda_1 \in \left[\lambda_2, 30\right]$ and $\lambda_3 \in \left[\lambda_1, 50\right]$. Then we try each value in threshold ranges iteratively and calculate the annualized ROI according to the Eq. (10). Figure 6 displays the arbitrage annualized ROI with different thresholds.

In Fig. 3, we find that the annualized ROI rises with the increasing thresholds of λ_1 and λ_3. However, there shows little changes when λ_2 increase, which indicates that the opening signals and stop-loss signals are more important factors in deciding the profits of arbitrage, compared with the closing signals. In the end, we choose a set of optimal thresholds for different signals, which is $\lambda_2 = 10, \lambda_1 = 11$ and $\lambda_3 = 50$. The arbitrage annualized ROI with this set of thresholds is 20.91%. Based on the optimal thresholds, we determine the trading interval with dynamical changes over time. Moreover, we also calculate the *mspread* series, in which we set the trading cost rate $c = 0.05$, consistent with the real market. The *mspread* and different trading intervals are displayed in Fig. 7.

As illustrated in Fig. 7, *mspread* is moving through the trading intervals. Once the *mspread* reach the different trading signals, different arbitrage actions are taken accordingly. The trading rules are shown in Table 1. Besides, we observe that the trading intervals in 2014–2015 are much larger than that in other periods. In actual, the rising trend in Chinese financial markets in 2014–2015 has greatly amplified the price volatility, which also creates plentiful arbitrage opportunities. However, the subsequent fallen trend in 2015–2017 shrinks the market imbalanced and erased

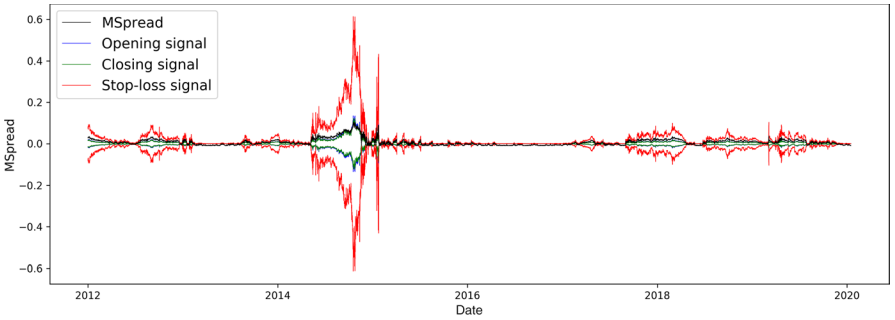

Fig. 7 Trading intervals and arbitrage opportunities

most of the arbitrage opportunities. Therefore, the occurrence of arbitrage opportunities largely depends on market performance. It's effective to detect the arbitrage opportunities using dynamic volatility to measure the trading intervals with optimized thresholds.

4.3 Machine Learning Prediction Results

As for machine learning methods, we mainly focus on Logistic regression, XGBoost, CNN and LSTM, which are used to train the relationship between prices series $\{x_t\}$ and arbitrage opportunities $\{y_t\}$. Specifically, the whole dataset covers 100,512 samples from 2012 to 2020. 65,520 samples from 2012 to 2017 are used as training set while 34,992 samples from 2018 to 2020 are used as test set. The training performance are shown in Fig. 8.

In Fig. 8, we check the overall accuracy of different methods and compare the precision, recall and f-measure of different labels which represent no arbitrage, positive arbitrage and negative arbitrage, respectively. Particularly, we care most about the positive arbitrage since it decides the upper limit of the arbitrage profits. Apparently, the precision, recall and f-measure of positive arbitrage have best performance when using LSTM. LSTM works well in extracting time series features and exploring the arbitrage opportunities. CNN is more sensitive to the non-arbitrage samples because the precision, recall and f-measure are higher when it comes to no arbitrage situation. Logistic regression performs relatively balanced in different labels. However, all the measures do not perform well when using XGBoost. This shows that XGBoost is more difficult to explore the arbitrage opportunities compared with other models. Generally, the different algorithmic mechanism causes the different performance of the models. Training performance is not enough to prove the superiority of the models. Therefore, we further conduct ROI evaluation on different models.

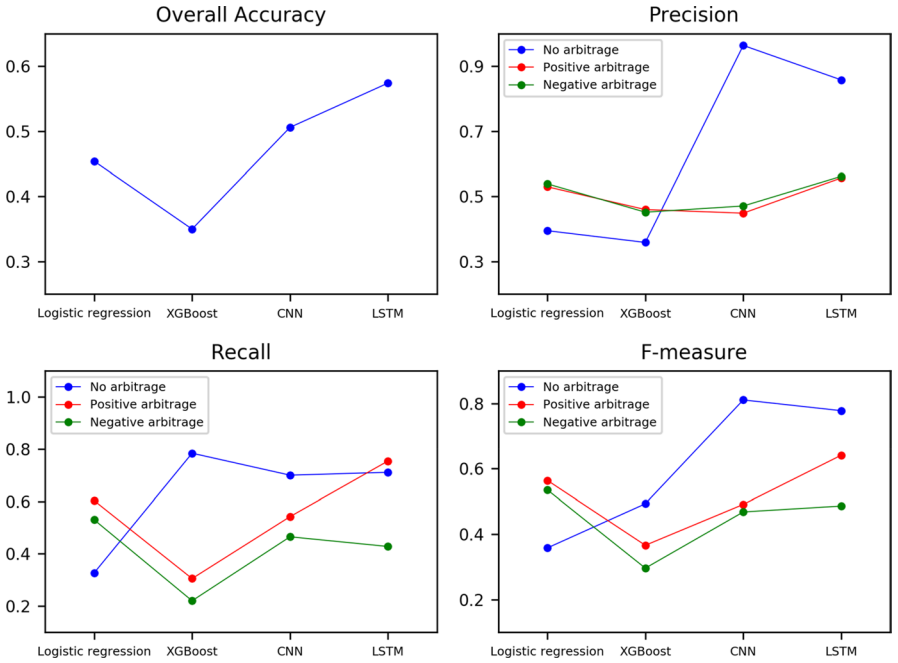

Fig. 8 Prediction performance

4.4 Profits Evaluation

We use the test set (34,992 samples from 2018 to 2020) to simulate statistical arbitrage trading and conduct the profits analysis based on both cointegration strategy and machine learning strategy. We mainly calculate the ROI and Sharpe ratio, which are shown in Table 5.

In table, we can see that cointegration strategy and CNN strategy deliver negative annualized ROI of − 9.7 and − 24.7%, respectively. In contrast, the annualized ROI

Table 5 ROI and Sharpe ratio results

	Cointegration	Logistic Regression	XGBoost	CNN	LSTM
Annualized ROI	− 0.097	0.320	2.688	− 0.247	5.134
Annualized Sharpe ratio	− 11.290	32.731	283.095	− 27.194	541.640
Times of positive arbitrage	13,037	14,838	8635	15,727	17,651
Times of negative arbitrage	12,700	12,496	6178	12,548	9672
Positive average ROI (%)	0.218	0.215	0.248	0.195	0.214
Negative average ROI (%)	− 0.226	− 0.248	-0.216	− 0.251	− 0.231
Positive average Sharpe ratio	0.325	0.231	0.281	0.262	0.317
Negative average Sharpe ratio	− 0.353	− 0.263	− 0.268	− 0.324	− 0.239

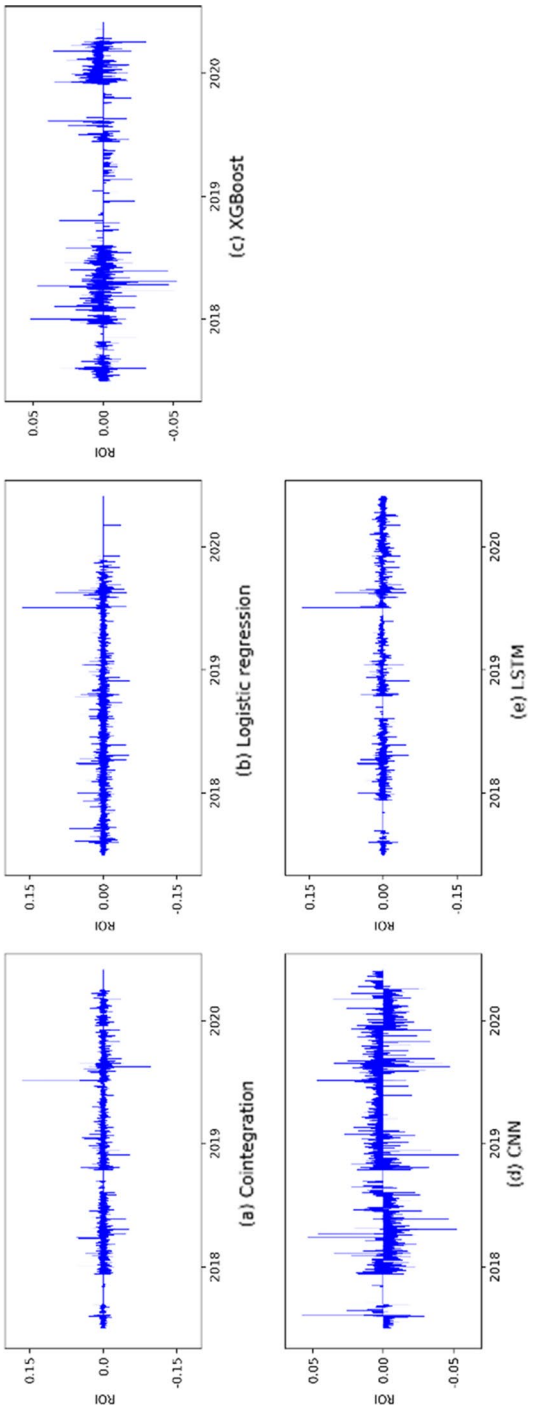

Fig. 9 Real-time ROI for different strategies

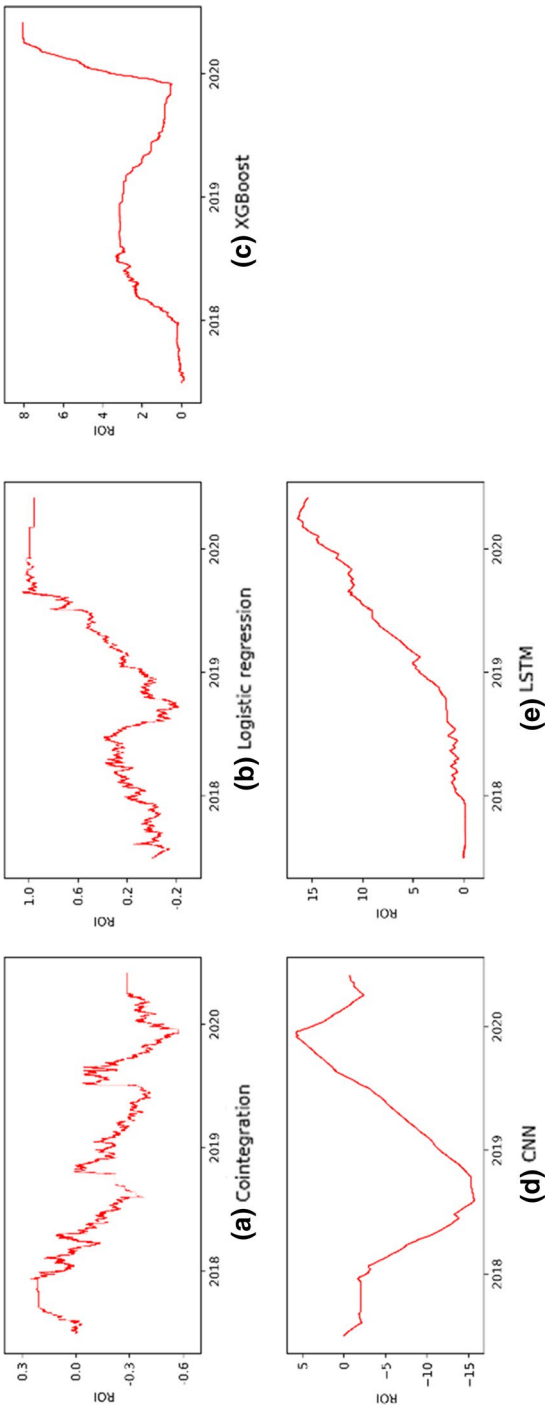

Fig. 10 Accumulated ROI for different strategies

of the other three strategies are all positive. During 2018–2020, LSTM has grabbed most opportunities for positive arbitrages, which are up to 17,651 times. CNN explores 15,727 times of positive arbitrage, yet it also encounters 12,548 times of negative arbitrage, leading to its' negative ROI. XGBoost ranks the second place of average and total ROI. Even though XGBoost does not explore too many arbitrage opportunities as other models, it achieves considerable profits, which indicating that it performs well in market timing

As for average ROI for real-time arbitrage, XGBoost has the highest positive average ROI and lowest negative average ROI. On the contrary, CNN has the lowest positive average ROI and highest negative average ROI. Considering Sharpe ratio, LSTM has the smallest negative Sharpe ratio while cointegration has the highest positive Shape ratio. We plot the real-time ROI in every time of arbitrage and accumulated ROI in Figs. 9 and 10

As we can see, the real-time ROI is fluctuated around 0 in Fig. 9. However, the volatility of CNN is relatively higher than other models, which explained the negative profits in arbitrage trading.

In Fig. 10, obviously that most strategies suffer a drawdown in 2018, especially for cointegration, logistic regression and CNN. As for XGBoost and LSTM, the drawdown shows very small compare to the subsequent growth of the accumulated ROI.

In summary, by comparing different strategies of arbitrage, we find that machine learning strategies performs better than cointegraion strategy. Specifically, LSTM gets the highest evaluations in both prediction and profits making. XGBoost explores the least arbitrage opportunities but achieving the highest positive average ROI. CNN is not good at dealing with sequential data, which results in its insufficient performance in ROI.

5 Conclusions

At present, statistical arbitrage using algorithmic trading becomes a mainstream trading techniques in quantitative finance domain. More and more investors including institutions and individuals pour in the market and star arbitraging, which makes the financial market increasingly competitive and hard to explore arbitrage opportunities. Therefore, finding an effective approach to explore arbitrage opportunities is quite necessary to ensure sufficient profits. In this study, we translate the arbitrage opportunities exploration problem into a multi-class prediction problem and propose machine learning strategy for statistical arbitrage. Besides, we re-implement the cointegration strategy and make comparison between these two strategies. Some findings are drawn from our empirical results. First, the price spreads of ETF and index futures are stationary, implying that mean reversion and price discovery function is still valid. Second, the GARCH model estimates the dynamic volatility of price spread, which significantly facilitate the determination of trading signals and trading interval. Regarding the training performance of machine learning algorithms, LSTM achieves best performance in overall accuracy and the different metrics in positive arbitrage samples, which is what we care most. CNN is more sensitive the non-arbitrage

samples and it achieve higher precision and f-measure in non-arbitrage samples. Logistic regression performs relatively balanced in different samples. Concerning the profits analysis, machine learning strategy outperform than cointegration strategy. Specifically, LSTM obtain highest annualized ROI and explore most arbitrage opportunities than other strategies. XGBoost has the maximum positive average ROI even though it explore the least arbitrage opportunities, which means that XGBoost performs better in arbitrage timing. The profitability of machine learning strategy further substantiate the market efficiency. To conclude, as a frontier domain in quantitative finance research, statistical arbitrage via machine learning strategy has greater potential and wider application than traditional strategy, which demands more attention and further study. In the future, we will attempt to integrate our strategy with more predictive factors and try to explain the reasons behind the arbitrage profits.

Funding This work was supported by National Natural Science Foundation of China (Nos. 71532013, 71431008 and 71572050).

Declarations

Conflict of interest All authors declare that they have no conflict of interest.

References

Abreu, D., & Brunnermeier, M. K. (2002). Synchronization risk and delayed arbitrage. *Journal of Financial Economics*, 66(2–3), 341–360

Ahn, D. H., Boudoukh, J., Richardson, M., & Whitelaw, R. F. (2002). Partial adjustment or stale prices? Implications from stock index and futures return autocorrelations. *The Review of Financial Studies*, 15(2), 655–689

Attari, M., Mello, A. S., & Ruckes, M. E. (2005). Arbitraging arbitrageurs. *The Journal of Finance*, 60(5), 2471–2511

Baker, M., & Savaşoglu, S. (2002). Limited arbitrage in mergers and acquisitions. *Journal of Financial Economics*, 64(1), 91–115

Balvers, R., Wu, Y., & Gilliland, E. (2000). Mean reversion across national stock markets and parametric contrarian investment strategies. *The Journal of Finance*, 55(2), 745–772

Barboza, F., Kimura, H., & Altman, E. (2017). Machine learning models and bankruptcy prediction. *Expert Systems with Applications*, 83, 405–417

Basak, S., & Croitoru, B. (2006). On the role of arbitrageurs in rational markets. *Journal of Financial Economics*, 81(1), 143–173

Basak, S., Kar, S., Saha, S., Khaidem, L., & Dey, S. R. (2019). Predicting the direction of stock market prices using tree-based classifiers. *The North American Journal of Economics and Finance*, 47, 552–567

Brogaard, J., Hendershott, T., & Riordan, R. (2014). High-frequency trading and price discovery. *The Review of Financial Studies*, 27(8), 2267–2306

Broussard, J. P., & Vaihekoski, M. (2012). Profitability of pairs trading strategy in an illiquid market with multiple share classes. *Journal of International Financial Markets, Institutions and Money*, 22(5), 1188–1201

Chaboud, A. P., Chiquoine, B., Hjalmarsson, E., & Vega, C. (2014). Rise of the machines: Algorithmic trading in the foreign exchange market. *The Journal of Finance*, 69(5), 2045–2084

Chakravarty, S., Gulen, H., & Mayhew, S. (2004). Informed trading in stock and option markets. *The Journal of Finance*, 59(3), 1235–1257

Chatzis, S. P., Siakoulis, V., Petropoulos, A., Stavroulakis, E., & Vlachogiannakis, N. (2018). Forecasting stock market crisis events using deep and statistical machine learning techniques. *Expert Systems with Applications, 112,* 353–371.

Chaudhuri, K., & Wu, Y. (2003). Random walk versus breaking trend in stock prices: Evidence from emerging markets. *Journal of Banking & Finance*, 27(4), 575–592

Chen, Y. L., & Gau, Y. F. (2010). News announcements and price discovery in foreign exchange spot and futures markets. *Journal of Banking & Finance*, 34(7), 1628–1636

Chiu, M. C., & Wong, H. Y. (2015). Dynamic cointegrated pairs trading: Mean–variance time-consistent strategies. *Journal of Computational and Applied Mathematics*, 290, 516–534

De Moura, C. E., Pizzinga, A., & Zubelli, J. (2016). A pairs trading strategy based on linear state space models and the Kalman filter. *Quantitative Finance*, 16(10), 1559–1573

Do, B., & Faff, R. (2010). Does simple pairs trading still work? *Financial Analysts Journal*, 66(4), 83–95

Fama, E. F. (1970). Efficient capital markets: Of theory and empirical work. *The Journal of Finance, 25*(2), 383–417.

Fama, E. F., & French, K. R. (1988). Permanent and temporary components of stock prices. *Journal of Political Economy, 96*(2), 246–273.

Gatev, E., Goetzmann, W. N., & Rouwenhorst, K. G. (2006). Pairs trading: Performance of a relative-value arbitrage rule. *The Review of Financial Studies*, 19(3), 797–827

Gârleanu, N., & Pedersen, L. H. (2013). Dynamic trading with predictable returns and transaction costs. *The Journal of Finance*, 68(6), 2309–2340

Hogan, S., Jarrow, R., Teo, M., & Warachka, M. (2004). Testing market efficiency using statistical arbitrage with applications to momentum and value strategies. *Journal of Financial Economics, 73*(3), 525–565.

Huck, N. (2015). Pairs trading: does volatility timing matter? *Applied Economics*, 47(57), 6239–6256

Huck, N. (2019). Large data sets and machine learning: Applications to statistical arbitrage. *European Journal of Operational Research*, 278(1), 330–342

Jordà, Ã., & Taylor, A. M. (2012). The carry trade and fundamentals: Nothing to fear but FEER itself. *Journal of International Economics*, 88(1), 74–90

Kao, C. (1999). Spurious regression and residual-based tests for cointegration in panel data. *Journal of Econometrics, 90*(1), 1–44.

Krauss, C., Do, X. A., & Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *European Journal of Operational Research*, 259(2), 689–702

Kozhan, R., & Tham, W. W. (2012). Execution risk in high-frequency arbitrage. *Management Science*, 58(11), 2131–2149

McMillan, D. G., & Speight, A. E. (2006). Nonlinear dynamics and competing behavioral interpretations: Evidence from intra-day FTSE-100 index and futures data. *Journal of Futures Markets: Futures, Options, and Other Derivative Products*, 26(4), 343–368

Neely, C. J., & Weller, P. A. (2013). Lessons from the evolution of foreign exchange trading strategies. *Journal of Banking & Finance*, 37(10), 3783–3798

Papantonis, I. (2016). Cointegration-based trading: evidence on index tracking & market-neutral strategies. *Managerial Finance*, 42(5), 449–471

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Systems with Applications, 42*(1), 259–268.

Qian, B., & Rasheed, K. (2007). Stock market prediction with multiple classifiers. *Applied Intelligence*, 26(1), 25–33

Ross, S. A. (1976). The Arbitrage Theory of Capital Asset Pricing. *Journal of Economic Theory*, 13, 341–360

Schnaubelt, M., Fischer, T. G., & Krauss, C. (2020). Separating the signal from the noise–financial machine learning for Twitter. *Journal of Economic Dynamics and Control*, 114, 103895

Schultz, P., & Shive, S. (2010). Mispricing of dual-class shares: Profit opportunities, arbitrage, and trading. *Journal of Financial Economics*, 98(3), 524–549

Tsay, R. S. (1998). Testing and modeling multivariate threshold models. *Journal of the American Statistical Association, 93*(443), 1188–1202.