Expert Systems with Applications 162 (2020) 113730

Contents lists available at ScienceDirect

Expert Systems with Applications

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

A new investment method with AutoEncoder: Applications to crypto currencies ☆

Masafumi Nakano [a], Akihiko Takahashi [b,*]

[a] GCI Asset Management, 10F Chiyoda First Bldg. East, 3-8-1 Nishi-Kanda, Chiyoda-ku, Tokyo 101-0065, Japan
[b] Faculty of Economics, The University of Tokyo, 7-3-1 Hongo Bunkyo-ku, Tokyo 113-0033, Japan

ARTICLE INFO

Article history:
Received 13 January 2020
Revised 23 April 2020
Accepted 6 July 2020
Available online 19 July 2020

Keywords:
AutoEncoder
Crypto currency
Delta hedging
Artificial neural network

ABSTRACT

This paper proposes a novel approach to the portfolio management using an AutoEncoder. In particular, features learned by an AutoEncoder with ReLU are directly exploited to portfolio constructions. Since the AutoEncoder extracts characteristics of data through a non-linear activation function ReLU, its realization is generally difficult due to the non-linear transformation procedure. In the current paper, we solve this problem by taking full advantage of the similarity of ReLU and an option payoff. Especially, this paper shows that the features are successfully replicated by applying so-called dynamic delta hedging strategy. An out of sample simulation with crypto currency dataset shows the effectiveness of our proposed strategy.

1. Introduction

It has been well-recognized that the asset allocation problem is one of the central issues in financial investment decisions. However, for portfolio constructions investors inevitably need to estimate some unknown variables such as expected returns, variances and other risk measures, which often suffers serious and considerable estimation errors. Especially, estimation of expected returns are more important because it mainly decides the sign of an asset position, i.e. taking long/short position, though it is one of the most controversial issues as discussed in the previous researches (e.g., Campbell & Thompson, 2007; Fama, 1965; Fama & French, 1993; Fama & French, 2012; Welch & Goyal, 2007). Therefore, to overcome this problem, two different approaches have been taken: One develops sophisticated return prediction models, and the other constructs portfolios independent from estimation of the expected returns, which are sometimes refereed as "μ-free strategies" (e.g., Braga, 2015). This paper contributes to the second approach by proposing a new μ-free strategy.

As for the first approach, by taking non-linearity and non-Gaussian properties of financial markets into account, many researchers have proposed sophisticated models based on Bayesian time-series analysis (e.g., Johannes, Korteweg, & Polson, 2014; Nakano, Takahashi, & Takahashi, 2017a) and machine learning techniques (e.g., Ballings, Van den Poel, Hespeels, & Gryp, 2015; Chong, Han, & Park, 2017; de Oliveira, Nobre, & Zarate, 2013; Nakano, Takahashi, & Takahashi, 2018). On the contrary, there are few researches for μ-free strategies. Although three strategies, i.e. minimum variance (MinVar), equally weighted risk contribution (ERC) portfolio by Maillard, Roncalli, and Teiletche (2010) and most diversified (MD) portfolio by Choueifaty and Coignard (2008) are well-known, they sometimes suffer large drawdowns. This crucial drawback is avoided by our proposed new μ-free strategy.

Concretely, MinVar and MD strategies tend to have portfolio weight concentrations. That is, the weights concentrate on assets with the low volatilities and correlations such as bonds. Therefore, their portfolios take large exposures to bond crash risks, which is not desirable for the investors. In addition, ERC portfolio with the universe constituting of a few asset classes does not work at all due to its long only restriction. For instance, even if all individual stocks included in S&P500 are tradable, the ERC portfolio cannot avoid market crashes, which causes substantial losses. Furthermore, one of the biggest problems with the volatility based portfolios is that those portfolios equally evaluate the upside and downside variations. In addition, since a large market crash sometimes breaks normal correlations among the asset prices so drastically that all of them get close to one, volatility based portfolios cannot avoid large drawdowns caused by a market crash.

☆ The views expressed in this paper are our own and do not reflect the institutions we are affiliated with. The authors are not responsible or liable in any manner for any losses and/or damages caused by the use of any contents in this research.

* Corresponding author.
E-mail address: akihikot@e.u-tokyo.ac.jp (A. Takahashi).

https://doi.org/10.1016/j.eswa.2020.113730
0957-4174/

Consequently, investors have a great demand for a new μ-free strategy with an ability to avoid substantial losses and drawdowns. A new μ-free strategy proposed in this paper enables to avoid those by incorporating features embedded in option contracts into portfolio constructions through artificial neural network (ANN). To the best of our knowledge, this is the first paper which applies ANN to the construction of a μ-free strategy, namely replicating a zero floored payoff without estimation of expected returns.

Here, we briefly explain ANN and its applications to the finance. Initiated by a pioneering work of Hinton, Osindero, and Teh (2006), deep learning technique has been developed in a tremendous speed, which realizes fast and precise learning of multi-layered ANNs. In general, ANNs consist of input, hidden and output layers, where an each layer has multiple information processing units called neurons. As reported in Cybenko (1989), this complex structure makes it possible to approximate non-linear functions with high accuracy. Naturally, there are a number of previous literatures on the applications of ANNs to financial investment problems (e.g., Ballings et al., 2015; Chong et al., 2017; Huck, 2010; Nakano et al., 2018; de Oliveira et al., 2013). In particular, we make use of an AutoEncoder with ReLU to obtain a small number of effectively diversified factors, whose detailed motivation is explained in Section 2.1.

An AutoEncoder, introduced by Hinton and Salakhutdinov (2006), is a kind of ANN trained to learn the smaller number of latent features which can reconstruct the input data itself as much as possible. At first, an AutoEncoder gets famous for pretraining of the deep layered neural network (DNN) (Hinton & Salakhutdinov, 2006), though pretraining process is not necessarily required due to the recent development of training algorithms. Consequently, an AutoEncoder is currently applied to dimensionality reduction (e.g., Hinton & Salakhutdinov, 2006; Wang, Yao, & Zhao, 2016), clustering (e.g., Song, Liu, Huang, Wang, & Tan, 2013; Xie, Girshick, & Farhadi, 2016), and anomaly detection (e.g., An & Cho, 2015; Sakurada & Yairi, 2014; Zhou & Paffenroth, 2017). In the context of financial investment, an AutoEncoder has been exploited as an extension of the pretraining process (e.g., Chong et al., 2017; Bao, Yue, & Rao, 2017). Differently, the current paper directly exploits features extracted by an AutoEncoder for portfolio constructions. One of the main reason why previous researches do not directly exploit obtained features is that realization of the extracted features is generally very difficult due to non-linear transformation procedures. The current work solves this problem by introducing a dynamic delta hedging strategy, which is explained with more details in Section 2.

To show the effectiveness of our proposed approach, we focus on an investment universe constituting of multiple crypto currencies as a numerical experiment. Crypto currencies, alternatives to traditional centralized currencies, are now rapidly emerging as new financial instruments thanks to the efficiency of crypto currency markets. See Le Tran and Leirvik (2019a, 2019b) and Aslan and Sensoy (2019) for notable researches on the efficiency of crypto currency markets. As the crypto currencies have more and more presence, their peculiar price dynamics have been clarified by some researches. For instance, Begušić, Kostanjčar, Stanley, and Podobnik (2018) has reported the heavier tails of Bitcoin returns than stocks. Also, Bitcoin has extremely high returns, high volatilities and low correlations to traditional assets, as reported in Briere, Oosterlinck, and Szafarz (2015). Therefore, it is an important task to show that our proposed strategy works even with those risky but also important assets. Moreover, since there are few researches which focus on portfolios with multiple crypto currencies, analyzing a multiple crypto currency portfolio itself is significant, apart from our proposed AutoEncoder based strategy.

As a result, numerical out-of-sample experiments demonstrate that our proposed approach robustly attains substantial risk reduction. In addition, the AutoEncoder based portfolio records the higher risk adjusted returns than the traditional strategies.

The remainder of the paper is organized as follows. Section 2 explains our proposed strategy based on AutoEncoder. Section 3 summarizes detailed settings for the numerical experiments. Section 4 shows the resulting investment performance. Finally, Section 5 concludes.

2. Methodology

2.1. An introduction of new option factor

In general, although an investor has a large investment universe, it is very time-consuming and ineffective to manage those instruments one by one, especially under situations where there are some relationship among the universe. Hence, there is a demand for an approach to find the smaller number of factors that can represent a market best. Concretely, an investor wants to obtain new factors which can explain all the variation of a investment universe by their linear transformation.

To solve this problem, principal component analysis (PCA) is often employed by scholars as well as practitioners. PCA is an orthogonal linear transformation which converts a large set of correlated variables into a new coordinate system, where the new variables called principal components are uncorrelated and ordered so that the first few retain most of variations of original variables. Hence, PCA is widely used as a dimension reduction technique in various fields such as biology, physics, engineering, and finance (e.g., Dultzin-Hacyan & Ruano, 1996; Gewers et al., 2018; Kambhatla & Leen, 1997; Lardic, Priaulet, & Priaulet, 2003; OuYang, Xu, Huang, & Chen, 2011; Rebonato, 2018).

Especially limited to finance, there is a lot of application examples. For instance, PCA enables to control movements in an yield curve by three or four driving factors, namely level, slope, and curvature factor (e.g. Nakano et al., 2018; Rebonato, 2018). Also, the uncorrelated variables help investors to design a well-diversified portfolio with more efficiency as shown in Lohre, Neugebauer, and Zimmer (2012), Lohre, Opfer, and Orszag (2014) and Partovi and Caputo (2004). Furthermore, to confirm risk diversification of a portfolio, risk managers often monitor the risk exposure and sensitivity to principal components (e.g., Rebonato, 2018; Meucci, 2009).

One of special reasons for such popularity of PCA in finance is that the principal components, i.e. linear transformation procedure in PCA can be easily realized by a simple position adjustment. Especially for an application to a portfolio management, implementability of factors is essential because an investor's goal is to make a profit in the real market through their portfolio. While PCA has such advantages, there exist some drawbacks in an application of PCA to the portfolio management.

Firstly, the principal components do not have a direct economic meaning by themselves, though we can interpret those based on obtained results. That is, they are only uncorrelated variables without clear directions, which implies that the estimation of expected returns is still necessary for a construction of portfolios with fine risk adjusted returns or restricted downside deviations. Since the estimation of expected returns is one of the most difficult and controversial topics in finance as shown in a lot of existing researches (e.g., Campbell & Thompson, 2007; Kwon & Moon, 2007; Lettau & Ludvigson, 2001; Nakano et al., 2017a; Pai & Lin, 2005; Welch & Goyal, 2007), this additional step deteriorates the advantages of PCA, i.e. its simplicity.

Secondly, due to the procedure in PCA, its representative ability depends on a linear transformation only, which implies that PCA may fail to capture potentially nonlinear structures within the investment universe. In other words, the use of nonlinear transformation may enable to extract the features which can explain variations of a market with higher accuracy. Instead, we may implement some nonlinear procedure in a real market by dynamically adjusting the position.

Furthermore, for the use of asset return data, there arises some problems in the essential operation of PCA, i.e. the estimation of eigenvalues of a sample covariance matrix. That is, a classical asymptotic theory, starting with Anderson (1958), shows that eigenvalues of the sample covariance matrix are consistent and asymptotically normal estimators of the population eigenvalues, at least when the data follow a multivariate normal distribution. However, asset returns are known to exhibit time-varying volatilities and heavy tails as shown in Bollerslev, Engle, and Wooldridge, (1988), Engle, Ng, and Rothschild (1990) and Rom and Ferguson, (1994), which contradicts to above required assumptions.

For these reasons, we want to create new factors satisfying the following points:

- One sided variation, i.e. no downside deviation to attain a high investment performance.
- Composed by a non-linear transformation to obtain more representative ability.
- Implementable through trading in a real market.

To achieve these conditions, we introduce a new option factor whose underlying asset is a portfolio constituting of all the assets among a given investment universe. That is, although a linear transformed factor is utilized as well as PCA, we focus on its positive part only. Clearly, this option factor satisfies one sided deviation and non-linearity thanks to a nice feature of an option. To find out this newly introduced option factor, we resort to an artificial neural network, i.e. AutoEncoder (AE), which is explained in the next section.

2.2. AutoEncoder

An AutoEncoder is a kind of ANN which is trained to reproduce the input data itself in an unsupervised manner, whose effectiveness is empirically validated in various fields (e.g., An & Cho, 2015; Hinton & Salakhutdinov, 2006; Sakurada & Yairi, 2014; Song et al., 2013; Wang et al., 2016; Xie et al., 2016; Zhou & Paffenroth, 2017). In summary, our AutoEncoder is a three layered neural network with ReLU activation function where the loss function is specified as mean squared errors (MSEs) between inputs and outputs.

Let us remark that the features obtained by this AutoEncoder correspond to the new option factors introduced in the previous Section 2.1. Firstly, by its construction, the AutoEncoder with its loss function extracts the features which are able to explain all the variations by its non-linear transformation as much as possible. Secondly, the nonlinear activation function, i.e. ReLU enables to attain no downside deviations of the extracted features. These two characteristics are the necessary qualifications for the option factor.

In addition, ReLU also contributes to implementability of the option factor. One of the main problems for the use of AutoEncoder is that the realization of nonlinear activation function in a real market is generally very challenging despite of its significance. However, with regard to ReLU, i.e. a zero floor function, the extracted feature can be realized by the financial derivatives called "Call Option". Of course in the real market, although there does not exist a corresponding call option, we can replicate a call option by dynamically adjusting our position called delta, which is explained in the next Section 2.3.

2.3. Realizing ReLU by option replication

Firstly, a three layered AutoEncoder constituting of input, hidden and output layers is generally described as the following nonlinear mapping $f : \boldsymbol{x} \in \mathbb{R}^{N_d} \mapsto \hat{\boldsymbol{x}} \in \mathbb{R}^{N_d}$:

$$\begin{aligned} &\boldsymbol{z} = g(\boldsymbol{y}),\ \boldsymbol{y} = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}\left(= (y_k)_{k=1,\dots,N_h}\right), \quad \boldsymbol{W} \in \mathbb{R}^{N_h \times N_d},\ \boldsymbol{b} \\ &\in \mathbb{R}^{N_h}, \quad g : \mathbb{R}^{N_h} \to \mathbb{R}^{N_h} \quad \hat{\boldsymbol{x}} = h(\tilde{\boldsymbol{z}}), \quad \tilde{\boldsymbol{z}} = \dot{\boldsymbol{W}}\boldsymbol{z} + \tilde{\boldsymbol{b}}, \quad \dot{\boldsymbol{W}} \\ &\in \mathbb{R}^{N_d \times N_h},\ \tilde{\boldsymbol{b}} \in \mathbb{R}^{N_d}, \quad h : \mathbb{R}^{N_d} \to \mathbb{R}^{N_d} \end{aligned} \tag{1}$$

where N_d and N_h stands for the number of units in input and hidden layer, respectively. In this section, we show how to replicate features $\boldsymbol{z}_t = (z_{t,i})_i = g(\boldsymbol{W}\boldsymbol{x}_t) = (\max(\boldsymbol{W}_i \cdot \boldsymbol{x}_t, 0))_i$, where $\boldsymbol{W}_i = (w_{i,j})_j \in \mathbb{R}^{\mathbb{N}}_{\mathrm{d}}$ stands for the i-th row vector of $\boldsymbol{W} = (w_{i,j})_{i,j}$ obtained by AutoEncoder. Although there are N_h features, we focus on replication of the i-th feature $z_{t,i}$ without loss of generality.

Here, let us briefly introduce an European call option, which is a financial agreement contracted at initial time t_0 between two parties, i.e. a buyer and a seller. The buyer is given the right, but not the obligation, to buy a particular financial instrument called "underlying asset" for a predetermined price ("strike") K at a certain time t_M called "maturity". Hereafter, S_t denotes the underlying asset price at time t. Since the payoff of a call option at the maturity can be written as $(S_{t_M} - K)^+$, an investor with initial wealth S_{t_0} can approximately reproduce the zero floored return $\max(r_{t_M}, 0)$ by a call option with strike $K = S_{t_0}$.[1]

In our numerical experiments, initial time t_0, maturity t_M and the underlying asset corresponds to the current investment start date t_j, the next day t_{j+1} and a portfolio constituting of 38 assets weighted by $\boldsymbol{W}_i$, respectively. However, since such an ideal option is not traded in a real market as stated in the previous section, it is necessary to replicate the corresponding option for realization of the feature obtained by AE. Although there are a number of researches for the option replication strategy such as Boyle and Vorst (1992), Leland (1985) and Takahashi and Yamazaki (2009), we adopt the most simple approach called dynamic delta hedging.

Under an assumption of no transaction costs, continuous trading and Black–Scholes (BS) model (Black & Scholes, 1973) for an underlying asset price process,[2] the option payoff can be replicated by dynamically adjusting the position. Hereafter, Δ_t and M_t stands for the position investing in the underlying asset and a risk free asset at time t, respectively. Then, for all time $s \in [t_0, t_M)$, holding the following position (Δ_s, M_s) is necessary and sufficient for the replication of an European call option:

$$\begin{aligned} &\Delta_s = N(d_+(s)), \\ &M_s = -Ke^{-r(t_M - s)}N(d_-(s)), \\ &d_\pm(s) = \frac{\log(S_s/S_{t_0}) + (r \pm \sigma_*^2/2)(t_M - s)}{\sigma_* \sqrt{t_M - s}}, \end{aligned} \tag{2}$$

where $N(\cdot), r$ and σ_* stands for a standard normal cumulative distribution function, risk-free rate in the market, and volatility of the underlying asset. As stated above, although this dynamic delta hedging assumes continuous time rebalancing, we cannot execute trading orders continuously in the real market. Therefore, the discretized dynamic delta hedging is implemented in our numerical experiments, whose details are explained later in Section 3.2.2.

[1] Precisely, the replicated return is slightly lower than the objective due to an option premium paid to the seller at initial time t_0.

[2] Under the Black–Scholes model, the volatility of the underlying asset and a risk free rate are assumed to be constants. See Shreve (2004) for details and the proofs.

2.4. Delta hedge based equally risk budgeting portfolio

In addition to the discovery and empirical implementation of option factors, a portfolio based on the newly introduced factors is also constructed to evaluate their effectiveness, because one of the main reasons for their introduction is to obtain meaningful factors for financial investment. Although the previous section shows the way to calculate weights for the i-th option factor, there still remains an important problem how to allocate investor's wealth to the option factor. In the following, assume the investment period is N days, that is, $0 = t_0 < t_1 < \cdots < t_{N-1} < t_N = T$.

Initiated by modern portfolio theory (Markowitz, 1952), there arises a number of researches for portfolio constructions which achieve investor's own targets for risk-adjusted returns or/and additional risk measures (e.g., Black & Litterman, 1992; Nguyen & Gordon-Brown, 2012; Wang, Wang, & Watada, 2011). However, most of the approaches inevitably need to estimate some unknown variables including expected returns, variances and other risk measures, which often causes serious and considerable estimation errors.

Especially, estimations of expected returns are very difficult but also important for a portfolio construction as suggested by Chopra and Ziemba (1993), which reports that estimation errors in mean are at least 10 times as important for a mean–variance portfolio as those in variance. To put it simply, since the sign of expected returns roughly corresponds to the sign of the positions, i.e. positive (negative) expected returns leads to long (short) positions, estimation errors of expected returns strongly affects the portfolio weights. Hence, many researchers have developed the risk-based μ-free strategies such as minimum-variance or risk parity portfolio (Maillard et al., 2010) which are independent from estimations of expected returns.

In the current paper, we implement a relatively simple μ-free strategy to confirm raw effects of obtained option factors. Particularly, we construct a constant ratio portfolio constituting of a long position of option factors because an option factor basically creates only positive return.[3] Specifically, we construct an equal weight portfolio constituting of the option factors whose underlying factors are adjusted to have the same volatility level α. If the levels of volatilities are different among the option factors, the contribution of a factor with low volatility is hidden by the other factors with high volatilities, which implies that we cannot fully utilize a diversification effect of the option factors.

Here, we note that our AutoEncoder is trained by daily return data, which implies that an extracted feature by AE with ReLU can be regarded as an ATM call option[4] with one day maturity. Since the investment period is much longer than the option maturity, it is necessary to roll options on each day, i.e. close the current contract and open a new option contract for the same underlying asset with the same conditions.

Namely, at each date t_j, we construct a portfolio constituting of call options with maturity $t_M = t_{j+1}$ and strike $K = S^{\ell}_{t_j}$ for $\ell = 1, \ldots, N_h$, where S^{ℓ}_t denotes the price of ℓ-th underlying asset at time t. Actually, such options are not tradable in the real market, we replicate the options by dynamic delta hedging introduced in Section 2.3. As a result, the portfolio weight $\boldsymbol{\omega}^{(j)}_t = \left(\omega^{(j)}_{t,1}, \ldots, \omega^{(j)}_{t,N_d}, \omega^{(j)}_{t,N_d+1}\right)' \in \mathbb{R}^{N_d+1}$ at time $t \in [t_j, t_{j+1})$ can be written as follows:

$$\begin{aligned}
\left(\omega^{(j)}_{t,1}, \ldots, \omega^{(j)}_{t,N_d}\right)' &= \frac{1}{N_h}\sum_{i=1}^{N_h}\Delta^{(j)}_{t,i}\lambda^{(j)}_i\tilde{\boldsymbol{W}}_{\boldsymbol{i}},\\
\lambda^{(j)}_i &:= \frac{\alpha}{\sqrt{\tilde{\boldsymbol{W}}_{\boldsymbol{i}}'\Sigma_j\tilde{\boldsymbol{W}}_{\boldsymbol{i}}}},\\
\omega^{(j)}_{t,N_d+1} &= 1 - \sum_{i=1}^{N_d}\omega^{(j)}_{t,i}
\end{aligned} \tag{3}$$

where N_h and $\tilde{\boldsymbol{W}}_{\boldsymbol{i}}$ represents the number of option factors and the i-th row vector of the weight matrix $\tilde{\boldsymbol{W}}$ learned by AE, respectively. Also, Σ_j denotes the covariance matrix of raw assets' rates of returns which is estimated based on information by time t_j.

In addition, $\Delta^{(j)}_{t,i}$ stands for the ratio of the i-th option factor necessary for the replication of a non-linear option payoff at time $t \in [t_j, t_{j+1})$, which corresponds to Δ_s appearing in Eq. (2). Moreover, $\left(\omega^{(j)}_{t,1}, \cdots, \omega^{(j)}_{t,N_d}\right)'$ and $\omega^{(j)}_{t,N_d+1}$ represents the weight for crypto currencies and the risk free asset, respectively. Furthermore, $\lambda^{(j)}_i$ means an adjustment coefficient for the i-th option factor, which scales a volatility of the underlying asset to α. Since the standard deviation of factors with the weight $\tilde{\boldsymbol{W}}_{\boldsymbol{i}}$ is $\sqrt{\tilde{\boldsymbol{W}}_{\boldsymbol{i}}'\Sigma_j\tilde{\boldsymbol{W}}_{\boldsymbol{i}}}$, a volatility of the i-th factor with the scaled weight vector $\lambda^{(j)}_i\tilde{\boldsymbol{W}}_{\boldsymbol{i}}$ becomes α.

As shown from Eq. (3), this portfolio does not require knowledges for expected asset returns, which means that this strategy is clearly a μ-free strategy. In addition, this portfolio does not rely on the future information after time t_j if the weight matrix $\tilde{\boldsymbol{W}}$ is calculated by the data before time t_j.

Finally, we note that this simplest equal weight strategy is appropriate for our analysis as stated above, because a complex portfolio construction procedure makes it difficult to investigate raw features of the option factors. Of course, we are able to apply option factors to the other various investment strategies, which will be one of our future research topics.

3. Application to crypto currencies

3.1. Dataset

For numerical experiments, we have downloaded from hourly closing price data $\{P_{t,i}\}_{t,i}$ over the period from 2017/12/22 0:00 (GMT) to 2019/05/18 0:00 (GMT) from a crypto currency exchange, Binance which is one of the most largest crypto currency exchanges all over the world. Here, $P_{t,i}$ stands for a closing price of i-th crypto currency at time t. In the following, time index $t = 0, 1, \ldots, 12192$ corresponds to 2017/12/22 0:00, 2017/12/22 1:00 and 2019/05/18 0:00, respectively.

Then, we calculate daily returns by $r_{t_j,i} = P_{t_j,i}/P_{t_{j-1},i} - 1$, where t_j denotes the time for every 24 hour. For example, t_0, t_1 and t_{508} corresponds to 2017/12/22 0:00, 2017/12/23 0:00 and 2019/05/18 0:00, respectively. This setup implies that we try to add a zero floor to the daily returns, which are realized by hourly discretized delta hedging.

We use the following 38 crypto currencies, which implies that the number of input units N_d is 38. Precise index names are listed in Table 1. These 38 crypto currencies are chosen in terms of the available data period and trading volume at Binance.

In addition, we also calculate hourly return data by $r_{t,i} = P_{t,i}/P_{t-1,i} - 1$, which are used in the dynamic delta hedging as explained in Section 2.3. As stated above, the position is rebalanced once an hour based on this hourly return data $\{(r_{t,i})_i\}_{t=t_{j-1},t_{j-1}+1,\ldots,t_j-1}$ in the dynamic hedge for the replication of option factors.

[3] Although the portfolio is constructed by a long position of the option factors, actually a short position may be held if taking a short position is necessary for the construction of option factors themselves.

[4] An ATM call option is a call option whose strike K equals to the price of the underlying asset at inception.

Table 1
Name list of crypto currencies.

0x	Aelf	Aeron	Ambrosus	Basic Attention Token	Binance Coin
Bitcoin	BitShares	Cardano	Chainlink	Dash	Enjin Coin
EOS	Ethereum	Ethereum Classic	Etherparty	Everex	ICON
Iota	Komodo	Lisk	Litecoin	Monaco token	Monero
Neo	Nuls	OmiseGO	Po.et	Qtum	Ripple
Stellar	Time New Bank	Tron	Verge	Viberate	Waltonchain
Waves	Zcash				

3.2. Implementation setup

3.2.1. AutoEncoder

In the current work, four kinds of AutoEncoder are implemented depending on the number of features, i.e. $N_h = 2, 5, 10$ and 20 to see the sensitivity of hidden units. Moreover in our AutoEncoder, the bias term $\boldsymbol{b}$ in the encoder is set to be zero because an existence of the bias term may lead to a portfolio which highly concentrates on a risk free asset. (The bias term, i.e. an addition of a return corresponds to a return from the risk-free asset.).

As a training dataset D for our AutoEncoder, we use a scaled return data $\boldsymbol{x}_{t_j} = \left\{x_{t_j,i}\right\}_i$ in the first 300 days, i.e. $D = \left\{\boldsymbol{x}_{t_j}\right\}_{j=1,\dots,300}$. Hereafter, we set $T_{all} := 508$ and $T_{train} := 300$. Concretely, the above return data $\boldsymbol{r}_{t_j} = \left(r_{t_j,i}\right)_i$ are scaled to training data $\boldsymbol{x}_{t_j}$ by its sample standard deviation $\boldsymbol{\sigma} = (\sigma_i)_i$ as follows:

$$x_{t_j,i} := \frac{r_{t_j,i}}{\sigma_i}, \quad \sigma_i^2 := \frac{1}{T_{train}} \sum_{j=1}^{T_{train}} \left(r_{t_j,i} - \frac{1}{T_{train}} \sum_{j=1}^{T_{train}} r_{t_j,i} \right)^2 \tag{4}$$

Since the total number of daily return data is 508, the rest of 208 data are used as test data for an out-of-sample investment simulation. Consequently, the investment starts at 2018/11/19.

For learning of these parameters, we employ a stochastic gradient descent (SGD) method, the so-called "Adam" (Kingma & Ba, 2014) where minibatches with size 32 are introduced. Also, we set the number of epochs (i.e. the number of training times) to be 500. Finally, let us remind that as introduced in Section 2.4, $\tilde{\boldsymbol{W}}$ and $\tilde{\boldsymbol{W}}_{\boldsymbol{i}}$ represents the weight matrix learned by AE and its i-th row vector, respectively.

Finally, we mention to instability of learning results of the AutoEncoder. That is, learning results generally depend on seeds of random numbers for parameter initialization, which sometimes affects on the resulting features especially in the case of $N_h = 10$ and 20. Considering that the cases of $N_h = 10$ and 20 include learning of features corresponding to subtle market variations, it seems reasonable that their learning results are not robust compared to those of the first several features which explains the majority of market variations.

In order to obtain the robust results, we employ a model averaging technique, where we first train the 50 patterns of AutoEncoders with the different random seeds. Then, we use an average of AutoEncoders whose Sharpe ratio in the training period is in the top 10 percentile as an investment performance in Section 4. Although this model averaging procedure is not necessary in the cases of $N_h = 2, 5$ because of the robustness of their leaning results, we apply this procedure to $N_h = 2, 5$ to implement a fair comparison analysis.

3.2.2. Dynamic delta hedging and portfolio construction

In this section, we explain details of our option factor replication and the resulting portfolio construction. Firstly as stated in Section 2.4, option factors $z_{t_j,i}$ are adjusted to have the same volatility α for the application to our portfolio construction. Specifically, we adjust the underlying factors to have 20% volatility in annual basis, that is 21.4 basis points $= 0.2/\sqrt{365 \times 24}$ in hourly basis, which implies that α appearing in Eq. (3) is set to be 21.4 basis points. Consequently, the i-th option factor $z_{t_{j+1},i}$, i.e. the replication target over the period $[t_j, t_{j+1})$ can be written as follows:

$$\begin{aligned}
z_{t_{j+1},i} &= \max\left(\lambda_i^{(j)} \tilde{\boldsymbol{W}}_{\boldsymbol{i}} \cdot \boldsymbol{x}_{\boldsymbol{t_{j+1}}}, 0\right) = \left(\lambda_i^{(j)}\left(\tilde{\boldsymbol{W}}_{\boldsymbol{i}} \circ \boldsymbol{\sigma}^{-1}\right) \cdot \boldsymbol{r}_{\boldsymbol{t_{j+1}}}\right)^+ \\
\lambda_i^{(j)} &:= \frac{\alpha}{\sqrt{\bar{\boldsymbol{W}}_{\boldsymbol{i}}{}' \Sigma_j \bar{\boldsymbol{W}}_{\boldsymbol{i}}}}, \\
\bar{\boldsymbol{W}}_{\boldsymbol{i}} &:= \tilde{\boldsymbol{W}}_{\boldsymbol{i}} \circ \boldsymbol{\sigma}^{-1}, \\
\alpha &= 0.00214,
\end{aligned} \tag{5}$$

where $\circ$, $\boldsymbol{\sigma}^{-1}$ and Σ_j stand for element-wise product, a 38-dim vector $\left(\sigma_i^{-1}\right)_i$ introduced in Eq. (4) sample covariance matrix based on hourly return data by time t_j, respectively. With these notations, the underlying asset price of the i-th option factor which corresponds to S_t in Section 2.3 can be described as a portfolio value constituting of 38 assets weighted by $\lambda_i^{(j)} \bar{\boldsymbol{W}}_{\boldsymbol{i}}$. Hereafter, $\hat{\boldsymbol{W}}_{\boldsymbol{i}}^{\boldsymbol{j}}$ denotes $\lambda_i^{(j)} \bar{\boldsymbol{W}}_{\boldsymbol{i}}$, i.e. $\hat{\boldsymbol{W}}_{\boldsymbol{i}}^{\boldsymbol{j}} := \lambda_i^{(j)} \bar{\boldsymbol{W}}_{\boldsymbol{i}}$.

As for the implementation of our "delta hedge based equally risk budgeting portfolio", we resort to an hourly discretization. That is, at each discretized time $t \in \{t_j, t_j + 1, \dots, t_{j+1} - 1\}$, the portfolio weight is rebalanced according to its weight $\boldsymbol{\omega}_t^{(j)}$ calculated by the following equation:

$$\begin{aligned}
\left(\omega_{t,1}^{(j)}, \dots, \omega_{t,N_d}^{(j)}\right)' &= \frac{1}{N_h} \sum_{i=1}^{N_h} \Delta_{t,i}^{(j)} \hat{\boldsymbol{W}}_{\boldsymbol{i}}^{\boldsymbol{j}}, \\
\hat{\boldsymbol{W}}_{\boldsymbol{i}}^{\boldsymbol{j}} &:= \lambda_i^{(j)} \bar{\boldsymbol{W}}_{\boldsymbol{i}}, \\
\lambda_i^{(j)} &:= \frac{\alpha}{\sqrt{\bar{\boldsymbol{W}}_{\boldsymbol{i}}{}' \Sigma_j \bar{\boldsymbol{W}}_{\boldsymbol{i}}}} \\
\omega_{t,N_d+1}^{(j)} &= 1 - \sum_{i=1}^{N_d} \omega_{t,i}^{(j)}, \\
\Delta_{t,i}^{(j)} &= N\left(d_{+,i}^{(j)}(t)\right), \\
d_{+,i}^{(j)}(t) &= \frac{\log\left(S_t^{(i)}/S_{t_j}^{(i)}\right) + \left(r + \sigma_{*,i}^2/2\right)(t_{j+1} - t)}{\sigma_{*,i}\sqrt{t_{j+1} - t}},
\end{aligned} \tag{6}$$

where $S_t^{(i)}$ represents a price of the i-th underlying asset recursively calculated by $S_{t+1}^{(i)} = S_t^{(i)} \times \left(\hat{\boldsymbol{W}}_{\boldsymbol{i}}^{\boldsymbol{j}} \cdot \boldsymbol{r}_{\boldsymbol{t}}\right)$ for time $t \in \{t_j, t_j + 1, \dots, t_{j+1} - 1\}$. Also as shown in Eq. (6), a volatility of the i-th underlying asset $\sigma_{*,i}$ and a risk-free rate r is necessary for the portfolio construction. With regard to a risk free rate r, we assume r to be 1.59% on an annual basis for all the period, which is an actual rate at 2018/11/01. Concerning the volatility $\sigma_{*,i}$, since the underlying asset is a linear combination of 38 assets, we can obtain its volatility if the covariance matrix of those 38 assets $\tilde{\Sigma}$ are given. Especially, the covariance matrix $\tilde{\Sigma}_j$ for the dynamic delta hedging over the period $[t_j, t_{j+1})$ is estimated by the following exponential moving average (EMA) methods with data before t_j:

$$\sigma_{m,n} = \sum_{\ell=0}^{t_j-1}(1-\delta)\delta^{\ell}\left(\tilde{r}_{t_j-\ell,m} - \mu_m\right)\left(\tilde{r}_{t_j-\ell,n} - \mu_n\right)$$
$$\mu_k = \sum_{\ell=0}^{t_j-1}(1-\delta)\delta^{\ell}\tilde{r}_{t_j-\ell,k} \tag{7}$$

where the parameter δ is chosen to be 12/13 so that the center of mass of weights is 12 hours. In addition, $\tilde{r}_{t,i}$ denotes a log return of $r_{t,i}$, i.e. $\tilde{r}_{t,i} = \log(1 + r_{t,i})$ because the Black–Scholes model assumes a log-normally distributed return. Although we choose this EMA method as often used in existing researches (e.g., Hurst, Ooi, & Pedersen, 2013; Nakano, Takahashi, & Takahashi, 2017b; Nakano, Takahashi, & Takahashi, 2017c; Nakano, Takahashi, & Takahashi, 2019; Rather, Agarwal, & Sastry, 2015; Takahashi & Yamamoto, 2013), results do not strongly depend on the estimation methods due to robustness of the volatility estimation.

4. Investment result

In this section, we present the performance of out of sample simulations as stated above. That is, an AutoEncoder is trained by training data, i.e. the first 300 days, and the investment performance is evaluated by the rest of data.

4.1. Effect of the number of hidden units

We firstly confirm the comparison analysis for the different number of option factors, which are summarized by Table 2. As a benchmark, the performance of buy and hold strategy of Bitcoin (BTC) is also shown in Table 2.

From these results, we can see that the performance of our proposed methodology depends on the number of hidden units. Namely, the cases of a small number of units, i.e. $N_h = 2$ and 5 record high risk adjusted returns, while performances based on 10 and 20 option factors do not perform well compared to the former cases. In particular, the performance of $N_h = 20$ are inferior to the benchmark BTC performance.

Although it seems natural that the risk-adjusted return gets better due to diversification effects as the number of factors increases, the actual performance is vice versa. As one of the possible reasons for this performance degradation, we can think of over-fitting problems or dynamic environmental changes over the training and test period. However, they are not the cause of this phenomenon because the performance in cases of $N_h = 20$ is still worse even when the AutoEncoder is trained by all the data including our test dataset.

Consequently, it is necessary to analyze the features of obtained option factors more closely, which is discussed in a supplemental file of this paper. See Appendix.

In the following sections, we choose the results of $N_h = 5$ as a result of AutoEncoder because this case has the most distinctive feature as explained in the supplemental file.

4.2. Comparison with traditional strategies

Next, we compare performances of our methodology with traditional strategies. Concretely for comparison, we calculate three μ-free strategies, i.e. equal weight strategy (EW), minimum variance portfolio (MinVar), and equally-weighted risk contributions portfolio (ERC), and three return prediction based strategies, i.e. multi-bang-bang (MBB) strategy, equally weighted long short (ELS) strategy, and mean variance (MV) portfolio.

MBB strategy is a strategy which invests all the wealth to an asset with the highest expected return. ELS strategy is an equally weighted portfolio whose position side (namely long or short) is determined depending on signs of the expected returns. Finally, a MV portfolio is a strategy which maximizes the expected return of a portfolio penalized by the portfolio volatility. Since option factors are realized on daily basis, we rebalance those strategies with daily intervals.

With regard to return predictions, we resort to two different methodology. One is an exponential moving average (EMA) method, which is often used in academic as well as in practice due to its simplicity and effectiveness. The other is an autoregressive integrated moving average (ARIMA) model, which is often used in time series analysis to better understand data and obtain the more accurate prediction. Especially, ARIMA model is often applied to the data with non-stationarity such as financial time series data. The parameters in ARIMA model is determined by AIC, which is reestimated at every rebalancing timing using all the data available at the current time.

In addition to these strategies, we also show the performance of buy and hold strategy of Bitcoin (BTC) as a market benchmark. Here, we note that since there are a few existing researches that investigate a portfolio with so large crypto currency universe, it is also important and significant to construct and analyze the investment performance of those traditional portfolios.

In the construction of MinVar and ERC, 24 hour EMA[5] is used for the estimation of the covariance matrix. Differently from those variance based strategies, since the performance of mean variance portfolio largely changes depending on estimates of expected returns and covariance matrices, we test four types of EMA, i.e. 96, 48, 24 and 12 hour EMA and ARIMA model, which correspond to 96, 48, 24, 12 and AR[6] in Table 3, respectively. The risk aversion parameter γ of the MV portfolio is determined to have the same annual volatility, i.e. 20% as the AutoEncoder in all cases.

From Table 3, we can see that our approach successfully attains substantial risk reduction compared to the other traditional strategies, which leads to the highest risk adjusted returns, i.e. Sharpe and Sortino ratio among all the strategies. Here, we note that Sharpe ratio is the most important and standard performance measure because it shows the portfolio excess return per unit of risk. For instance, the portfolio with the higher Sharpe ratio provides better return than the one with lower Sharpe ratio and the same risk.

Although crypto currencies have favorable features for investors, i.e. high returns and low correlations to traditional assets, they also have extremely high volatilities. Therefore, especially for multi crypto currency universe, there is a great demand to create a strategy which enables to reduce their high risks. However, as shown in Table 3, all of traditional strategies fail to reduce their risks.

Particularly as for μ-free strategies which put high values on their risk control, two of them (i.e. EW and ERC) show destructive performances. In addition, although the risk adjusted returns of MinVar are relatively high, its risk measures are inferior to the simple BTC, which implies the failure of MinVar portfolio, because it aims to minimize its portfolio variance. All the crypto currencies constituting of the current investment universe are positively correlated, differently from the traditional investment universe where there are negatively correlated assets such as stock and bond. Therefore, the robustness of risk-based strategies is totally broken by hard market crashes. In other words, the traditional μ-free strategies do not work with the crypto currency universe.

Next, we focus on the performance of return prediction based strategies. Firstly, we can see that all of the MBB strategies record the destructive performances, which implies importance of a

[5] α hour EMA means that the weight decaying factor δ in Eq. (7) is $(\alpha - 1)/(\alpha + 1)$, which implies that the latest α hours data have about 86% of weight for EMA.

[6] 24 hour EMA is used for the covariance matrix in ARMV.

Table 2
Effect of the number of option factors on the investment performance.

	$N_h = 2$	$N_h = 5$	$N_h = 10$	$N_h = 20$	BTC
CR	11.1%	9.6%	6.0%	−0.2%	20.4%
SD	4.2%	3.6%	3.3%	2.7%	63.9%
DD	2.2%	2.0%	1.8%	1.6%	46.1%
MDD	2.4%	1.9%	1.9%	2.6%	51.8%
ShR	262.7%	268.2%	183.6%	NA	31.9%
SoR	506.1%	489.4%	326.0%	NA	44.2%
StR	463.5%	504.1%	322.7%	NA	39.4%

CR = annualized compound return, SD = annualized standard deviation, DD = annualized downside deviation, MDD = maximum drawdown, ShR = Sharpe ratio, SoR = Sortino ratio, StR = Sterling ratio. NA is presented in the fourth column because the risk adjuted returns, i.e. ShR, SoR and StR do not make any sense for the strategy with negative return. N_h represents the number of features obtained by AE.

Table 3
Investment performance over the test period.

	AE	MinVar	ERC	EW	BTC	MV96	MV48	MV24	MV12	MVAR
CR	9.6%	161.3%	−5.7%	−8.3%	20.4%	100.5%	136.6%	289.0%	−82.3%	−13.8%
SD	3.6%	78.0%	86.9%	88.9%	63.9%	43.9%	59.3%	116.6%	531.3%	186.7%
DD	2.0%	51.2%	65.5%	67.2%	46.1%	30.7%	39.0%	71.9%	290.7%	81.7%
MDD	1.9%	47.2%	60.4%	61.1%	51.8%	22.2%	27.3%	47.9%	99.3%	67.3%
ShR	268.2%	206.8%	NA	NA	31.9%	229.2%	230.3%	247.9%	NA	NA
SoR	489.4%	315.2%	NA	NA	44.2%	327.1%	349.8%	401.7%	NA	NA
StR	504.1%	341.6%	NA	NA	39.4%	451.9%	500.6%	603.0%	NA	NA

	MBB96	MBB48	MBB24	MBB12	MBBAR	ELS96	ELS48	ELS24	ELS12	ELSAR
CR	−98.7%	−99.4%	−98.7%	−94.4%	−98.7%	97.3%	6.3%	−15.6%	18.0%	50.4%
SD	231.4%	228.1%	238.5%	237.5%	238.5%	73.8%	71.2%	68.6%	65.3%	64.1%
DD	124.2%	124.2%	125.4%	121.5%	125.4%	45.9%	46.5%	46.0%	41.8%	17.1%
MDD	91.8%	94.8%	93.0%	86.7%	93.0%	36.2%	47.4%	47.3%	45.0%	12.3%
ShR	NA	NA	NA	NA	NA	131.8%	8.9%	NA	27.5%	78.5%
SoR	NA	NA	NA	NA	NA	211.7%	13.6%	NA	42.9%	294.6%
StR	NA	NA	NA	NA	NA	268.5%	13.4%	NA	39.9%	407.9%

CR = annualized compound return. SD = annualized standard deviation. DD = annualized downside deviation. MDD = maximum drawdown. ShR = Sharpe ratio. SoR = Sortino ratio. StR = Sterling ratio. AE = our proposed strategy. MinVar = minimum variance portfolio. ERC = equally-weighted risk contributions portfolio. EW = equally weighted portfolio. BTC = Bitcoin. MV96, MV48, MV24, MV12, MVAR = mean variance portfolio based on 96/48/24/12 hour EMA and ARIMA model, respectively. MBB96, MBB48, MBB24, MBB12, MBBAR = multi-bang-bang strategy based on 96/48/24/12 hour EMA and ARIMA model, respectively. ELS96, ELS48, ELS24, ELS12, ELSAR = equally weighted long short (ELS) strategy based on 96/48/24/12 hour EMA and ARIMA model, respectively.

diversification effect in the crypto currency market. Let us remind that MBB strategy strongly depends on estimates of the expected returns because all the wealth is invested in one asset with the highest expected return, while the other two strategies are diversified among the given crypto currency universe.

As for the other two return prediction based strategies, the performance strongly depends on estimates of the expected returns. Concretely, although some of the strategies record extremely high return which enables to attain relatively high risk adjusted returns, all of them fail their risk reduction. Here, we remind that MV portfolios are adjusted to have the same 20% volatility in annual base as the AutoEncoder. However, Table 3 shows that the resulting volatility are more than 40% in all MV cases while that of our proposed method is less than 5%, which shows the high risk reduction ability of our proposed approach.

In addition, Table 3 shows that MDD of the AE based portfolio is extremely low. Let us remark that it is risky for investors to invest a fund whose performance strongly depends upon its start timing. Therefore, MDD is an important performance measure because it shows how much investors lose if they begin to invest their fund at the worst timing. Conversely, investors can expect robust performance from the portfolio with low MDD independent of investment timing. For instance, those who invest MV24 possibly suffer 47.9% loss in the worst timing, while investors with our AE based portfolio suffer only 1.9% loss even in the worst case, which also supports the robustness of our approach.

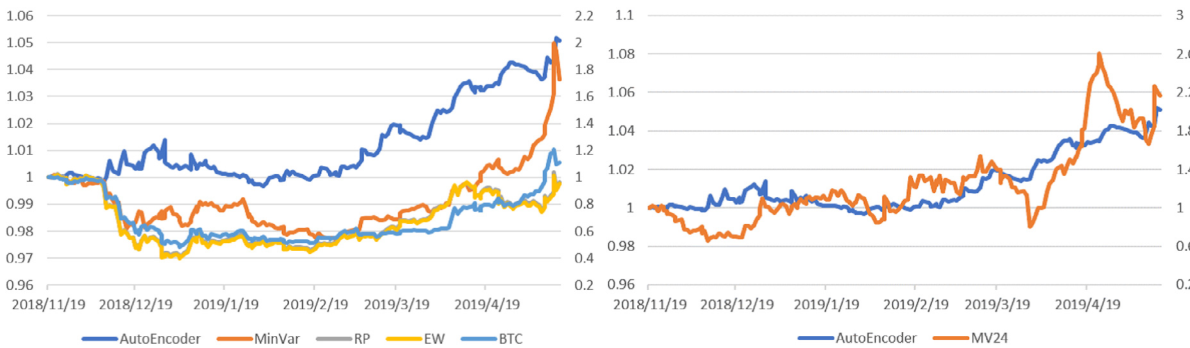
Fig. 1 presents comparison of the investment performance for each strategy, which also shows the superiority of our scheme. That is, the proposed portfolio does not suffer serious damages during the first half period when there is a large market crash. In addition, our portfolio again escapes from a large drawdown in the second half period when MV24 record the serious 47.9% drawdown. Since the proposed approach remains to be flat in such difficult periods, it records the lowest standard and downside deviation and maximum drawdown, which leads to the fine risk adjusted returns.

4.3. Further investigation for practical implementation

4.3.1. Statistical test for Sharpe ratio

In this section, we implement a statistical test to confirm robustness of our proposed approach. Since investment performances largely depend on their investment timing, the high performance measure resulting from its past track record does not always guarantee such high performance. In other words, investors are often deceived by temporally sampled high performance measures recorded just by luck, which implies that risk measures are also unobservable random variables.

Once we regard performance measures as random variables, we can implement a statistical test for rejecting the hypothesis that observed performance measures are below a certain threshold with a given significance level. Especially, we focus on Shrape ratio, i.e. the most well-known and important performance measure by applying an asymptotic distribution of a sample based estimate of Sharpe ratio which Opdyke (2007) derives based on results by Christie (2005).



Performance comparison to μ-free strategies   Performance comparison to MV24

Fig. 1. Investment performance: comparison with traditional strategies. In the two figures, the left and right axis corresponds to portfolio value of AutoEncoder and the other traditional strategies, respectively. As for the return prediction based strategies, since the scale is totally different depending on the strategies, we focus on MV24 which shows the highest Sharpe ratio in the return prediction based portfolios.

Concretely, Opdyke (2007) shows that $\widehat{\text{ShR}}$, i.e. the sample-based estimate of ShR asymptotically follows a normal distribution under stationary and ergodicity assumptions (without i.i.d. assumptions) for time series of portfolio returns:

$$\sqrt{T}\left(\widehat{\text{ShR}} - \text{ShR}\right) \underset{\sim}{a} N\left(0, 1 + \frac{\text{ShR}^2}{4}\left[\frac{\mu_4}{\sigma^4} - 1\right] - \text{ShR}\frac{\mu_3}{\sigma^3}\right), \tag{8}$$

where $X \underset{\sim}{a} N(x, y)$ means that X asymptotically follows a normal distribution with mean x and variance y. Also, μ_3/σ^3 and μ_4/σ^4 represents skewness and kurtosis of the returns, respectively. (See Eq. (6) in p.4 of Opdyke (2007).)

Table 4 shows a probability that the true Sharpe ratio of a strategy is less than zero. Here, we omit the strategies with negative Sharpe ratios in Table 3. This table clearly presents the superiority of our proposed approach. That is, our proposed strategy can reject the null hypothesis $\text{H}_0 : \text{ShR} < 0$, i.e. the true Sharpe ratio of the strategy is less than zero at 5% significance level, while all of the other strategies cannot. In other words, only our proposed strategy can accept the alternative hypothesis $\text{H}_1 : \text{ShR} \geqslant 0$.

4.3.2. Rebalancing scheme for replication and transaction cost effect

Firstly, this section analyzes effects of different rebalancing methods for replications on investment performances. Since the main subject of this paper is to propose a new μ free strategy which is robust against large drawdowns, the hourly discretization scheme, i.e. the most straightforward way is adopted to replicate a given option payoff. However, there is no guarantee that this hourly discretization scheme is optimal in an actual implementation because the BS model based dynamic delta hedging is theoretically justified only under the continuous trading assumption. Therefore, we need to implement other rebalancing schemes for replications to examine their effects on actual investment performances.

Concretely, two types of simple adjustment are implemented.[7] The first one increases the fixed rebalancing interval, 1 hour. Namely, we change it to two and four hours, whose investment performances are shown in the sixth (2 Hour) and seventh column (4 Hour) in Table 5. The other type is to rebalance a portfolio only when the delta hedging position at t, Δ_t deviates by a prefixed amount (α) from the previous hedging position Δ_{t-1}. Concretely, the replication portfolio is rebalanced only when $|\Delta_t - \Delta_{t-1}| > \alpha$, where four types of α is tested, i.e., $\alpha = 40\%, 30\%, 20\%$ and 10%. The resulting investment performances are presented from the second to fifth column ($\Delta 40, \Delta 30, \Delta 20, \Delta 10$) in Table 5.

From Table 5, we can see that the hourly discretization scheme is not optimal at all. That is, the risk adjusted returns (ShR, SoR, StR) and P-value(= P(ShR<0)) of the newly introduced rebalancing schemes except 4Hour are superior to those of the hourly discretization scheme.

Generally in replication of an option contract, when the underlying asset price does not show a clear trend towards ITM (in the money) nor OTM(out of the money) directions, rebalancing with less frequencies enjoys smaller costs. In contrast, when the price shows a strong trend towards ITM(OTM) directions, infrequent rebalancing does not catch it up to get(suffer) less(more) profits(-costs). In total, infrequent rebalancing schemes $\Delta 30$ and $\Delta 40$ provide better results than the others in this empirical analysis.

In the following, we introduce transaction costs into the analysis to consider the more realistic setting. Particularly, we assume that 1.2 basis points transaction costs are charged per trading according to the fee schedule set by Binance, which is the data source of our simulations. Also for comparison, we show performances of traditional strategies which record positive returns in the case of no transaction costs.

Table 6 shows performances of our proposed and traditional strategies with transaction costs, which clearly indicates that returns of our strategies get worse by more frequent rebalancing. However, the risks, i.e. SD, DD and MDD still remain small even for the 1Hour scheme whose rerun substantially gets worse from 9.6% to 1.1%. Thus, our proposed scheme can achieve the least requirement as a new μ free strategy, namely creating portfolios which attains robust risk reduction without estimation of expected returns.

As for the traditional strategies that are rebalanced on daily basis, although their performances also get worse, effects of transaction costs are smaller than the 1Hour rebalancing scheme.

Next, we focus on results of our strategy based on newly introduced rebalancing schemes. From Table 6, we can see that our strategies with $\Delta 40$ and $\Delta 30$ still work well to outperform all the existing traditional strategies. In terms of SoR and P-value (= P (ShR<0)), $\Delta 20$ also gives the better result than traditional strategies. In contrast, the 2 Hour and 4 Hour schemes do not. This result suggests rebalancing only when the delta hedging position changes largely rather than automatically reducing trading intervals, since it costs too much to rebalance a portfolio even when the position fluctuations are small. Because simple adjustment schemes of the

[7] Taking the randomness of sample Sharpe ratios as shown in Section 4.3.1 into consideration, the analysis in this section adopts a simple model averaging method to get rid of the effects caused by seeds of random numbers instead of selecting well performing cases in the training period as explained in Section 3.2.1.

Table 4
P-value, P(ShR< 0).

AutoEncoder	MinVar	BTC	MV96	MV48	MV24	ELS96	ELS48	ELS12	ELSAR
2.3%	10.9%	32.3%	10.5%	10.0%	9.2%	16.2%	37.0%	33.1%	18.2%

AutoEncoder = our proposed strategy. MinVar = minimum variance portfolio. BTC = Bitcoin. MV96, MV48, MV24 = mean variance portfolio based on 96/48/24 hour EMA, respectively. ELS96, ELS48, ELS12, ELSAR = equally weighted long short (ELS) strategy based on 96/48/12 hour EMA and ARIMA model, respectively.

Table 5
Performance sensitivity to the rebalancing strategy.

	1 Hour (Base case)	Δ40	Δ30	Δ20	Δ10	2 Hour	4 Hour
CR	9.6%	9.8%	9.7%	9.2%	8.9%	9.7%	8.2%
SD	3.6%	2.6%	2.5%	2.9%	3.2%	3.5%	3.7%
DD	2.0%	1.5%	1.3%	1.6%	1.8%	1.9%	2.3%
MDD	1.9%	1.0%	0.5%	1.1%	1.6%	1.3%	1.8%
ShR	268.2%	382.4%	383.1%	312.7%	274.1%	280.2%	222.2%
SoR	489.4%	675.4%	764.7%	570.9%	490.7%	501.4%	353.6%
StR	504.1%	950.8%	1791.0%	813.1%	567.5%	767.3%	462.4%
P-value	2.3%	0.31%	0.12%	0.97%	2.13%	1.84%	5.69%

CR = annualized compound return. SD = annualized standard deviation. DD = annualized downside deviation. MDD = maximum drawdown. ShR = Sharpe ratio. SoR = Sortino ratio. StR = Sterling ratio. P-value = P(ShR< 0). Base case = rebalancing the portfolio at every hour. Δ40, Δ30, Δ20, Δ10 = rebalancing the portfolio only when the delta hedging position changes more than 40/30/20/10 %, respectively. 1Hour, 2Hour, 4Hour = rebalancing the portfolio for every 1/2/4 hour, respectively.

Table 6
Effect of the transaction cost.

	1 Hour	BTC	Δ40	Δ30	Δ20	Δ10	2 Hour	4 Hour
CR	1.1%	20.4%	6.6%	6.2%	4.7%	3.0%	3.4%	3.3%
SD	3.4%	63.9%	2.6%	2.5%	3.0%	3.3%	3.5%	3.7%
DD	2.1%	46.1%	1.5%	1.4%	1.7%	2.0%	2.1%	2.4%
MDD	2.8%	51.8%	1.4%	0.6%	1.5%	2.2%	2.2%	2.6%
ShR	32.8%	31.9%	256.5%	243.9%	157.7%	91.5%	96.9%	89.4%
SoR	54.1%	44.2%	430.7%	451.6%	267.7%	150.8%	160.0%	136.3%
StR	40.8%	39.4%	477.7%	962.1%	302.5%	135.7%	157.4%	126.9%
P-value	39.8%	32.3%	3.0%	2.8%	11.6%	24.3%	23.0%	25.1%

	MinVar	MV96	MV48	MV24	ELS96	ELS48	ELS12	ELSAR
CR	108.0%	80.7%	95.0%	138.9%	92.1%	2.6%	12.6%	22.1%
SD	77.9%	43.9%	59.3%	116.5%	73.8%	71.2%	65.3%	33.1%
DD	51.8%	31.0%	39.5%	73.2%	46.0%	46.6%	41.9%	23.3%
MDD	51.3%	24.1%	29.2%	49.4%	36.7%	48.2%	45.9%	23.7%
ShR	138.6%	184.0%	160.1%	119.2%	124.8%	3.6%	19.3%	66.9%
SoR	208.7%	260.5%	240.1%	189.7%	200.3%	5.5%	30.1%	95.0%
StR	210.5%	335.3%	325.2%	281.1%	251.1%	5.3%	27.5%	93.4%
P-value	15.7%	13.5%	14.7%	15.7%	16.9%	38.5%	35.1%	28.2%

CR = annualized compound return. SD = annualized standard deviation. DD = annualized downside deviation. MDD = maximum drawdown. ShR = Sharpe ratio. SoR = Sortino ratio. StR = Sterling ratio. P-value = P(ShR< 0). Δ40, Δ30, Δ20, Δ10 = rebalancing the portfolio only when the delta hedging position changes more than 40/30/20/10 %, respectively. 1Hour, 2Hour, 4Hour = rebalancing the portfolio for every 1/2/4 hour, respectively. MinVar = minimum variance portfolio. BTC = Bitcoin.

second type achieve fine performances even in the presence of transaction costs, we expect that more sophisticated rebalancing strategies such as in Zakamouline (2006) enhance those furthermore, which is one of our future research topics.

5. Conclusion

This paper has developed a new approach to the asset allocation problem using an AutoEncoder. Particularly, we have directly applied factors obtained by an AutoEncoder to constructing a downside hedged portfolio. Although realization of the factors is generally difficult due to their non-linear features arising from the activation function, the similarity in ReLU and an option payoff enables to replicate those by a dynamic delta hedging. Furthermore, we have proposed a new portfolio named as "delta hedge based equally risk budgeting portfolio" constituting of extracted option factors adjusted by their volatilities.

As a numerical example, we focus on a novel investment universe, a set of crypto currencies, which are new financial alternative assets with both natures of currency and commodity. With a dataset of multi crypto currencies, an out of sample numerical experiment has shown that our proposed approach attains the better risk return profile than the traditional strategies. Especially, our strategy succeeds in the substantial reduction of risks and drawdowns.

CRediT authorship contribution statement

Masafumi Nakano: Conceptualization, Methodology, Software, Formal analysis, Investigation, Writing - original draft, Writing - review & editing, Visualization. **Akihiko Takahashi:** Conceptualization, Methodology, Formal analysis, Investigation, Writing - review & editing, Writing - review & editing, Supervision, Project administration.

Declaration of Competing Interest

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

Acknowledgement

We are very grateful to the editor Professor Binshan Lin and two anonymous referees for their precious comments and suggestions, which substantially improve the previous version of the paper. This research is supported by CARF (Center for Advanced Research in Finance). Also, this work is supported by JSPS KAKENHI Grant No. JP17J09127.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, athttps://doi.org/10.1016/j.eswa.2020.113730.

References

An, J., & Cho, S. (2015). Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE, 2*(1).

Anderson, T. W. (1958). *An introduction to multivariate statistical analysis: Wiley series in probability and mathematical statistics: Probability and mathematical statistics*. .

Aslan, A., & Sensoy, A. (2019). Intraday efficiency-frequency nexus in the cryptocurrency markets. *Finance Research Letters*.

Ballings, M., Van den Poel, D., Hespeels, N., & Gryp, R. (2015). Evaluating multiple classifiers for stock price direction prediction. *Expert Systems with Applications, 42*(20), 7046–7056.

Bao, W., Yue, J., & Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PloS One, 12*(7). e0180944.

Begušić, S., Kostanjčar, Z., Stanley, H. E., & Podobnik, B. (2018). Scaling properties of extreme price fluctuations in Bitcoin markets. *Physica A: Statistical Mechanics and its Applications, 510*, 400–406.

Black, F., & Litterman, R. (1992). Global portfolio optimization. *Financial Analysts Journal, 48*(5), 28–43.

Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of Political Economy, 81*(3), 637–654.

Bollerslev, T., Engle, R. F., & Wooldridge, J. M. (1988). A capital asset pricing model with time-varying covariances. *Journal of Political Economy, 96*(1), 116–131.

Boyle, P. P., & Vorst, T. (1992). Option replication in discrete time with transaction costs. *The Journal of Finance, 47*(1), 271–293.

Braga, M. D. (2015). *Risk-based approaches to asset allocation: Concepts and practical applications*. Springer.

Briere, M., Oosterlinck, K., & Szafarz, A. (2015). Virtual currency, tangible return: Portfolio diversification with bitcoin. *Journal of Asset Management, 16*(6), 365–373.

Campbell, J. Y., & Thompson, S. B. (2007). Predicting excess stock returns out of sample: Can anything beat the historical average? *The Review of Financial Studies, 21*(4), 1509–1531.

Chong, E., Han, C., & Park, F. C. (2017). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Systems with Applications, 83*, 187–205.

Chopra, V. K., & Ziemba, W. T. (1993). The effect of errors in means, variances, and covariances on optimal portfolio choice. *Journal of Portfolio Management, Winter, 1993*(19), 2.

Choueifaty, Y., & Coignard, Y. (2008). Toward maximum diversification. *The Journal of Portfolio Management, 35*(1), 40–51.

Christie, S. (2005). *Is the Sharpe ratio useful in asset allocation?* Macquarie Applied Finance Centre Research Paper. .

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems, 2*(4), 303–314.

de Oliveira, F. A., Nobre, C. N., & Zarate, L. E. (2013). Applying Artificial Neural Networks to prediction of stock price and improvement of the directional prediction index? Case study of PETR4, Petrobras, Brazil. *Expert Systems with Applications, 40*(18), 7596–7606.

Dultzin-Hacyan, D., & Ruano, C. (1996). General statistics and principal component analysis of multiwavelength properties of Seyfert galaxies. *Astronomy and Astrophysics, 305*, 719.

Engle, R. F., Ng, V. K., & Rothschild, M. (1990). Asset pricing with a factor-ARCH covariance structure: Empirical estimates for treasury bills. *Journal of Econometrics, 45*(1–2), 213–237.

Fama, E. F. (1965). The behavior of stock-market prices. *The Journal of Business, 38*(1), 34–105.

Fama, E. F., & French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics, 33*(1), 3–56.

Fama, E. F., & French, K. R. (2012). Size, value, and momentum in international stock returns. *Journal of Financial Economics, 105*(3), 457–472.

Gewers, F. L., Ferreira, G. R., de Arruda, H. F., Silva, F. N., Comin, C. H., Amancio, D. R., & Costa, L. D. F. (2018). *Principal component analysis: A natural approach to data exploration*. arXiv preprint arXiv:1804.02502. .

Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation, 18*(7), 1527–1554.

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science, 313*(5786), 504–507.

Huck, N. (2010). Pairs trading and outranking: The multi-step-ahead forecasting case. *European Journal of Operational Research, 207*(3), 1702–1716.

Hurst, B., Ooi, Y. H., & Pedersen, L. H. (2013). Demystifying managed futures. *Journal of Investment Management, 11*(3), 42–58.

Johannes, M., Korteweg, A., & Polson, N. (2014). Sequential learning, predictability, and optimal portfolio returns. *The Journal of Finance, 69*(2), 611–644.

Kambhatla, N., & Leen, T. K. (1997). Dimension reduction by local principal component analysis. *Neural Computation, 9*(7), 1493–1516.

Kingma, D. P., & Ba, J. (2014). *Adam: A method for stochastic optimization*. arXiv preprint arXiv:1412.6980. .

Kwon, Y. K., & Moon, B. R. (2007). A hybrid neurogenetic approach for stock forecasting. *IEEE Transactions on Neural Networks, 18*(3), 851–864.

Lardic, S., Priaulet, P., & Priaulet, S. (2003). PCA of the yield curve dynamics: Questions of methodologies. *Journal of Bond Trading and Management, 1*(4), 327–349.

Leland, H. E. (1985). Option pricing and replication with transactions costs. *The Journal of Finance, 40*(5), 1283–1301.

Le Tran, V., & Leirvik, T. (2019a). A simple but powerful measure of market efficiency. *Finance Research Letters, 29*, 141–151.

Le Tran, V., & Leirvik, T. (2019b). Efficiency in the markets of crypto-currencies. *Finance Research Letters, 101382*.

Lettau, M., & Ludvigson, S. (2001). Consumption, aggregate wealth, and expected stock returns. *The Journal of Finance, 56*(3), 815–849.

Lohre, H., Opfer, H., & Orszag, G. (2014). Diversifying risk parity. *Journal of Risk, 16*(5), 53–79. .

Lohre, H., Neugebauer, U., & Zimmer, C. (2012). Diversified risk parity strategies for equity portfolio selection. *The Journal of Investing, 21*(3), 111–128.

Maillard, S., Roncalli, T., & Teiletche, J. (2010). The properties of equally weighted risk contribution portfolios. *The Journal of Portfolio Management, 36*(4), 60–70.

Markowitz, H. (1952). Portfolio selection. *The Journal of Finance, 7*(1), 77–91.

Meucci, A. (2009). Managing diversification. *Risk, 22*(5), 74.

Nakano, M., Takahashi, A., & Takahashi, S. (2017a). Generalized exponential moving average (EMA) model with particle filtering and anomaly detection. *Expert Systems with Applications, 73*, 187–200.

Nakano, M., Takahashi, A., & Takahashi, S. (2017b). Creating investment scheme with state space modeling. *Expert Systems with Applications, 81*, 53–66.

Nakano, M., Takahashi, A., & Takahashi, S. (2017c). Fuzzy logic-based portfolio selection with particle filtering and anomaly detection. *Knowledge-Based Systems*.

Nakano, M., Takahashi, A., & Takahashi, S. (2018). Bitcoin technical trading with artificial neural network. *Physica A: Statistical Mechanics and its Applications, 510*, 587–609.

Nakano, M., Takahashi, A., & Takahashi, S. (2019). State space approach to adaptive fuzzy modeling for financial investment. *Applied Soft Computing, 105590*.

Nakano, M., Takahashi, A., Takahashi, S., & Tokioka, T. (2018). On the effect of Bank of Japan's outright purchase on the JGB yield curve. *Asia-Pacific Financial Markets, 25*(1), 47–70.

Nguyen, T. T., & Gordon-Brown, L. (2012). Constrained fuzzy hierarchical analysis for portfolio selection under higher moments. *IEEE Transactions on Fuzzy Systems, 20*(4), 666–682.

Opdyke, J. D. J. (2007). Comparing Sharpe ratios: So where are the p-values? *Journal of Asset Management, 8*(5), 308–336.

OuYang, D., Xu, J., Huang, H., & Chen, Z. (2011). Metabolomic profiling of serum from human pancreatic cancer patients using 1 H NMR spectroscopy and principal component analysis. *Applied Biochemistry and Biotechnology, 165*(1), 148–154.

Pai, P. F., & Lin, C. S. (2005). A hybrid ARIMA and support vector machines model in stock price forecasting. *Omega, 33*(6), 497–505.

Partovi, M. H., & Caputo, M. (2004). Principal portfolios: Recasting the efficient frontier. *Economics Bulletin, 7*(3), 1–10.

Rather, A. M., Agarwal, A., & Sastry, V. N. (2015). Recurrent neural network and a hybrid model for prediction of stock returns. *Expert Systems with Applications, 42*(6), 3234–3241.

Rebonato, R. (2018). *Bond pricing and yield curve modeling: A structural approach*. Cambridge: Cambridge University Press. https://doi.org/10.1017/9781316694169.

Rom, B. M., & Ferguson, K. W. (1994). Post-modern portfolio theory comes of age. *Journal of Investing, 3*(3), 11–17.

Sakurada, M., & Yairi, T. (2014). Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis* (p. 4). ACM. .

Shreve, S. E. (2004). *Stochastic calculus for finance II: Continuous-time models* (Vol. 11). Springer Science & Business Media.

Song, C., Liu, F., Huang, Y., Wang, L., & Tan, T. (2013). Auto-encoder based data clustering. In *Iberoamerican Congress on Pattern Recognition* (pp. 117–124). Berlin, Heidelberg: Springer.

Takahashi, A., & Yamamoto, K. (2013). Generating a target payoff distribution with the cheapest dynamic portfolio: An application to hedge fund replication. *Quantitative Finance, 13*(10), 1559–1573.

Takahashi, A., & Yamazaki, A. (2009). Efficient static replication of European options under exponential Levy models. *Journal of Futures Markets, 29*, 1–15. https://doi.org/10.1002/fut.20339.

Wang, B., Wang, S., & Watada, J. (2011). Fuzzy-portfolio-selection models with value-at-risk. *IEEE Transactions on Fuzzy Systems, 19*(4), 758–769.

Wang, Y., Yao, H., & Zhao, S. (2016). Auto-encoder based dimensionality reduction. *Neurocomputing, 184*, 232–242.

Welch, I., & Goyal, A. (2007). A comprehensive look at the empirical performance of equity premium prediction. *The Review of Financial Studies, 21*(4), 1455–1508.

Xie, J., Girshick, R., & Farhadi, A. (2016). Unsupervised deep embedding for clustering analysis. In *International conference on machine learning* (pp. 478–487).

Zakamouline, V. I. (2006). Optimal hedging of options with transaction costs. *Wilmott Magazine, 118*, 70–82.

Zhou, C., & Paffenroth, R. C. (2017). Anomaly detection with robust deep autoencoders. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 665–674). ACM. .