Quantitative Finance, 2021
Vol. 21, No. 10, 1753–1772, https://doi.org/10.1080/14697688.2021.1905171

Option hedging using LSTM-RNN: an empirical analysis

JUNHUAN ZHANG *†‡ and WENJUN HUANG†‡

†School of Economics and Management, Beihang University, Beijing, People's Republic of China
‡Key Laboratory of Complex System Analysis, Management and Decision (Beihang University), Ministry of Education, Beijing, People's Republic of China

(*Received 20 May 2020; accepted 12 March 2021; published online 24 June 2021*)

This paper proposes an optimal hedging strategy in the presence of market frictions using the Long Short Term Memory Recurrent Neural Network (LSTM-RNN) method, which is a modification of the method proposed in Buehler *et al.* (Deep hedging. *Quant. Finance*, 2019, **19**(8), 1271–1291). The market frictions are transaction costs, liquidity constraints, trading limits and cost of funds. The loss function is a spectral risk measure. We first make an empirical analysis of the LSTM-RNN model of real option markets, which are the Asian market (domestic market 50 ETF option, Hong Kong Hang Seng Index Option, Nikkei Index Option), the North American market (S&P 500 Index Option) and the European market (FTSE 100 Index Option). The benchmark models are from Leland (Option pricing and replication with transaction costs. *J. Finance.*, 1985, **40**(5), 1283–1301), Boyle and Vorst (Option replication in discrete time with transaction costs. *J. Finance*, 1992, **47**(1), 271–293) and Whalley and Wilmott (A hedging strategy and option valuation model with transaction costs. OCIAM Working Paper, Mathematical Institute, Oxford, 1993). Finally, we compare the results from the LSTM-RNN model with benchmark models involving transaction costs for both simulated market data generated by Geometric Brownian Motion (GBM) and the Heston model and real market data. The results show that the LSTM-RNN model outperforms benchmark models for low or medium volatility (<0.8), OTM moneyness and under a certain risk level ($<80\%$) in the GBM market setting. For the 50ETF option market, the LSTM-RNN model outperforms benchmark models for ATM and under a certain risk level ($<15\%$). For the HSI option market, the LSTM-RNN model outperforms benchmark models when transaction costs are smaller than 1.5%. For the Nikkei and S&P 500 option markets, the LSTM-RNN model always outperforms benchmark models. For the FTSE option market, the LSTM-RNN model outperforms benchmark models when moneyness is not too deeply ITM.

Keywords: Option hedging; Deep learning; Risk management; Portfolio optimization

JEL Classification: G11, D81

1. Introduction

Options hold an indispensable position in financial markets due to their diversity and flexibility. They are an important tool for investors who wish to arbitrage, hedge or speculate. According to Boyd (2015), making markets for options has always been one of the main businesses of market makers. However, how to properly control option risk exposure is the primary challenge faced by each option market maker due to the complexity of options. The option hedging problem has always been an essential issue in risk management.

The traditional option hedging strategy is based on the Black–Scholes–Merton (BSM) option pricing model. More precisely, traders charge a fee calculated with the BSM pricing formula for sellings an option at the beginning of a trade and during the trade they manage the market risk using the Greeks. However, Jankova (2018) shows that the BSM model has many limitations because it assumes that the market is frictionless, which does not conform to actual financial markets. Market makers have to adjust their positions by relying on experience, which results in poor operability and low accuracy. Moreover, the hedging strategies based on the traditional model, either analytically or numerically, depend on the estimation of a market parameter from the underlying asset's price S_t. Evidently, a misspecification of S_t's dynamics could possibly lead to disastrous hedging errors. Therefore, a non-parametric hedging strategy is needed to solve this problem under transaction costs.

Traditional option pricing models are developed by taking the transaction costs using the option replication strategy

*Corresponding author. Email: junhuan_zhang@buaa.edu.cn, junhuan.zhang@gmail.com

into consideration. Leland (1985) first accounts for option pricing using a hedging strategy with proportional transaction fees. In this paper, transactions are conducted in discrete time and there is a given transaction fee. Leland's Option Pricing and Replication with Transactions Costs is revised to obtain an option pricing and replication model with transaction fees. He himself proposed a modified volatility to solve the hedging error problem caused by transaction costs. The basic idea is to adjust the replication at a given time interval in the framework of the continuous-time Black–Scholes–Merton model. By adding an item containing the transaction cost factor to the volatility, the increase in the option price caused by the adjusted volatility can just offset the transaction cost. Toft (1996), after Leland's groundbreaking correction of the BSM model, calculates the expected value of the hedging error and gives an explicit solution. Kabanov and Safarian (1997) give the range of hedging errors with proportional transaction costs and argue that Leland's option replication strategy is only speculative. Soner *et al.* (1995) also point out that this method cannot be accurately replicated, because as long as the price of the option is positive, the profit of holding the underlying stock is higher than the option value on the expiration date. Leland's discontinuous replication method is actually the Local-in-Time method, which is based on the premise of determining the replication interval in advance to find the optimal option replication solution. This method was further extended by Boyle and Vorst (1992) and Whalley and Wilmott (1993). It is now widely used in practical applications thanks to its low computational complexity.

However, option replication has its inherent flaws. It has also been proved in the previously mentioned literature that this method cannot be accurately replicated. In fact, Leland (1985) conducted a more detailed research and obtained a strategy whose hedging error is infinitely close to zero. But, this strategy gives an upper bound for the option price. Considering the disadvantages of the option replication pricing strategy, Hodges (1989) recognizes that pricing methods must incorporate certain optimization factors to find efficient option hedging strategies. The most representative research is the local risk minimization strategy proposed by Lamberton *et al.* (1998), and the idea of maximizing utility proposed by Davis (1997). These provide new insights into option hedging and basically present a new framework to price and hedge financial derivatives, such as in the classic study by Owen (2002).

Recently, machine learning methods have been applied to the hedging options. Hutchinson *et al.* (1994) study the use of neural networks to price and hedge European options and give a parameter-free (thus greek-free) pricing and hedging strategy which is different from BSM, but their paper does not consider trading costs. Andreou *et al.* (2008) combine the artificial neural network and parametric models to price and hedge European options. They use the main idea of the previous paper, but add some explicit features to the model which increases the accuracy and performance of the hedging strategy. Ru and Shin (2011) present a methodology to hedge an option's delta dynamically using ANN, trying to improve the option delta hedging weakness caused by the BSM model's impractical assumptions. Halperin (2017) develops a discrete-time option pricing model based on the Q-Learning method of reinforcement learning. Spiegeleer *et al.* (2018) show how machine learning techniques can help solve traditional quant problems such as option pricing and hedging. In the fitting and estimation context, deploying deep learning algorithms can speed up the solution process as well as giving a fair accuracy. Krauss *et al.* (2017) apply deep learning techniques in statistical arbitrage instead of derivative pricing and hedging and prove that the tree-based model could give a substantial return over time. In Deep Hedging by Buehler *et al.* (2019), the authors consider the hedging problem from the perspective of the market maker's charge for over-the-counter trades. The market maker needs to have the same convex risk (refer to İlhan *et al.* (2009)) when he holds a short position on derivatives as when he does not hold a position at all. The minimum cash flow then leads to the corresponding hedging scheme. Kolm and Ritter (2019) propose a reinforcement learning model to optimally trade off the trading cost and the hedging variance. Buehler, Mohan *et al.* (2019) propose an entropy measure to maximize a hedged portfolio's profit/loss distribution with liquid hedging instruments and test their algorithm using simulated data with transaction costs.

Buehler *et al.* (2019), Buehler, Mohan *et al.* (2019) indeed claim that they take into account market frictions other than transaction costs, such as market impact and liquidity constraints, in their modeling; however, neither theory nor numerical experiments are given. In fact, few empirical studies were made in the above mentioned studies. Therefore, we modify the model proposed in Buehler *et al.* (2019) and conduct multiple empirical tests. First, we consider more market friction factors, such as transaction costs, liquidity constraints, trading limits and costs of funds and give explicit expressions for the modeling. Second, we apply the modified model to real markets, which are the Asian market (domestic market 50 ETF option, Hong Kong market Hang Seng Index Option, Nikkei Index Option), the North American market (S&P 500 Index Option) and the European market (FTSE 100 Index Option). Third, we compare the experimental results from the modified model with benchmark models with transaction costs using both simulated and real market data. The benchmark models are the Leland model, the Boyle–Vorst model and the Whalley–Wilmott model. The results show that a higher volatility induces higher CVaR loss in the GBM market setting. When moneyness increases, the different markets show different properties. A higher risk level induces higher CVaR loss in most scenarios, except for the LSTM-RNN model for the Nikkei and the S&P 500. A higher transaction cost proportion induces higher CVaR loss in all markets. In the GBM market setting, the LSTM-RNN model outperforms benchmark models for low or medium volatility (<0.8), OTM moneyness and under a certain risk level ($<80\%$). For the 50ETF option market, the LSTM-RNN model outperforms benchmark models for ATM and under a certain risk level ($<15\%$). For the HSI option market, the LSTM-RNN model outperforms benchmark models when transaction cost is smaller than 1.5%. For the Nikkei and S&P 500 option market, the LSTM-RNN model always outperforms benchmark models. For the FTSE option market, the LSTM-RNN model outperforms benchmark models when moneyness is not too deeply ITM. Lastly, we apply the LSTM-RNN model to exotic options and show some preliminary results. These show that hedging strategy derived from the LSTM-RNN model performs well for the spectral risk measure.

The rest of this paper is structured as follows. In Section 2, we establish an optimization problem by the modeling option hedging problem taking into account transaction costs, liquidity constraints, trading limits and cost of funds. In Section 3, we use both simulated markets generated by the Geometric Brownian Model (GBM) and the Heston model, and use real option market data to compare the hedging results of the modified model and the classic models. Then, we conduct an empirical analysis for market frictions other than transaction costs and show some preliminary results on hedging exotic options. Section 4 presents the conclusion and suggests further work.

2. Option hedging modeling

2.1. Portfolio value under market frictions

The incomplete market consists of two assets: traded underlying asset, c.f. stock, S_t and non-traded option C_t. We consider an option market maker selling m units of European options at time 0 whose initial price is p_0 thus potentially the agent will pay m units of option's payoff $h(S_T)$ at expiry time T.

To hedge this liability, one can only trade a certain amount of traded underlying asset. Under transaction cost, the agent may hold Δ_t shares of stock S_t at time $t \in [0, T]$. The trading strategies thus can be defined as $\{\Delta_{t_k}\}$ where $t_k, k \in [0, n]$ is a discrete variable because continuous hedging is not possible in the real market. With this setting: $t_0 = 0, t_n = T, t_i - t_{i-1} = T/n, i \in [1, n]$. Specifically, at time 0 and time T the agent does not hold any position on underlying asset, which means $\Delta_{t_0} = \Delta_{t_n} = 0$. This method is mainly inspired from Monoyios (2004).

At expiry time T, the agent's wealth consists of four parts:

- Option premium: $m \cdot p_0$. Option seller gains the premium.
- Liability at maturity: $m \cdot h(S_T)$.
- Stock position holding PnL: Over the period t_k to t_{k+1}, the agent holds $\{\Delta_{t_k}\}$ stock thus generates PnL $\Delta_{t_i} \cdot (S_{t_{i+1}} - S_{t_i})$. Summing up all PnL (transaction cost not included), we will get: $\sum_{i=0}^{n-1} \Delta_{t_i} \cdot (S_{t_{i+1}} - S_{t_i})$.
- Transaction cost: We assume that buying or selling x units of stocks will incur $c(x)$ cost. The agent adjusts its position by longing or shorting $|\Delta_{t_i} - \Delta_{t_{i-1}}|$ at time t_i. Therefore, transaction cost over the period $[0, T]$ can be expressed as follows: $\sum_{i=0}^{n} c(|\Delta_{t_i} - \Delta_{t_{i-1}}|)$.

The agent's PnL at expiry time T is as follows:

$$W_T = m \cdot p_0 - m \cdot h(S_T) + \sum_{i=0}^{n-1} \Delta_{t_i} \cdot (S_{t_{i+1}} - S_{t_i}) - \sum_{i=0}^{n} c(|\Delta_{t_i} - \Delta_{t_{i-1}}|). \quad (1)$$

Here we consider market maker shorts 1 European option, hence the case of $m = 1$. In real market trading, transaction cost will be proportional to underlying asset value and relatively small, so we define $c(x) = \epsilon \cdot S \cdot x$, where ϵ is the proportional transaction costs. The agent's portfolio PnL at time T can be rewritten as follows:

$$W_T(\Delta_t) = p_0 - h(S_T) + \sum_{i=0}^{n-1} \Delta_{t_i} \cdot (S_{t_{i+1}} - S_{t_i}) - \sum_{i=0}^{n} \epsilon \cdot S_{t_i} \cdot |\Delta_{t_i} - \Delta_{t_{i-1}}|. \quad (2)$$

Apart from transaction costs, market frictions include liquidity constraint, trading limits and cost of funds.

- Liquidity constraint: Suppose the cash available for market maker from issuing date to the expiry date is C, then the opposite value of the sum of cash flows at any moment should be inferior to C. That is to say, $-W_t < C$ where

$$W_t = \begin{cases} p_0, & t = 0 \\ p_0 + \sum_{i=0}^{j-1} \Delta_{t_i} \cdot (S_{t_{i+1}} - S_{t_i}) - \sum_{i=1}^{j} \epsilon \cdot S_{t_i} \cdot |\Delta_{t_i} - \Delta_{t_{i-1}}|, & 0 < t < T, t_j = t \\ p_0 - h(S_T) + \sum_{i=0}^{j-1} \Delta_{t_i} \cdot (S_{t_{i+1}} - S_{t_i}) - \sum_{i=1}^{j} \epsilon \cdot S_{t_i} \cdot |\Delta_{t_i} - \Delta_{t_{i-1}}|, & t_j = t = T \end{cases} \quad (3)$$

- Trading limit: Different entities may have different risk levels. The risk exposure during issuing date of the option and its expiry date should be inferior to a certain level based on the trading limit. As a result, the hedge ratio should be larger than certain value β, which gives us the following.

$$\Delta_{t_i} > \beta \Delta_{t_i}^{BS}, \quad (4)$$

where $\Delta_{t_i}^{BS}$ is the Greek Delta derived from BSM formula.
- Cost of funds: Minimum margin is the initial amount required to be deposited into a margin account before selling short. Because short selling is essentially selling of stocks that are not owned, there are strict margin requirements. The occupation cost is therefore

$$M = \sum_{i=1}^{n} \epsilon' \Delta_{t_i} S_{t_i} H(\Delta_{t_i}), \quad (5)$$

where ϵ' is the margin level, $H(x)$ is one Heaviside-like function and defined as follows:

$$H(x) = \begin{cases} -1, & x < 0 \\ 0, & x \geq 0 \end{cases}. \quad (6)$$

2.2. Loss function

The market maker provides market's liquidity, gain bid/ask spread while trying to control the risk. Logically, the agent wishes to minimize their expected loss at maturity. In this paper, we only focus on hedging process and do not take bid/ask spread into consideration. Suppose the option's initial price p_0 is given, if a loss function can be fixed $l : \mathbb{R} \rightarrow \mathbb{R}_+$, then we can define the optimal hedging strategy as follows:

$$\inf_{\Delta_t} \mathbb{E}(l(W_T(\Delta_t))). \tag{7}$$

In mathematical finance, a risk measure is defined as a mapping from a set of random variables to the real numbers. This set of random variables represents portfolio returns and the real number presents market risk. A spectral risk measure, which is commonly used in business, satisfies several properties:

- Positive homogeneity: for every portfolio returns X and positive value λ¿0, $l(\lambda X) = \lambda l(X)$;
- Monotonicity: for portfolio returns X and Y such that $X \geq Y, l(X) \leq l(Y)$;
- Sub-additivity: for portfolio returns X and Y, $l(X + Y) \leq l(X) + l(Y)$;
- Translation-invariance: for every portfolio returns X and $\alpha \in \mathbb{R}, l(X + \alpha) = l(X) - \alpha$;

A common risk metric used in risk management is Conditional Value at Risk (CVaR), it determines the potential loss in the portfolio being assessed and the probability of occurrence for the defined loss. We do not use entropy value or expectation value because we focus on potential losses rather than returns. Given risk-reversion coefficient of a trader,VaR can be defined as follows: $\mathrm{VaR}_\gamma(W) = \inf(m \in \mathbb{R} : \mathbb{P}(W < -m) \leq \gamma)$. Please note that VaR is not a spectral risk measure because it is not subaddictive. Then given a confidence level $\alpha \in [0, 1)$, based on the study of Follmer *et al.* (2004), the loss function (CVaR) can be easily defined as the integral of VaR over the risk level:

$$l(W) = \frac{1}{1-\alpha} \int_0^{1-\alpha} \mathrm{VaR}_\gamma(W)\, \mathrm{d}\gamma. \tag{8}$$

2.3. *Optimization problem solution*

Define market setting MS_k as the ensemble of all market states, including asset price (predicted or historical), transaction cost proportion, option premium, liability at maturity, liquidity constraint, trading limits and cost of funding at time t_k. Recall that $t_0 = 0, t_n = T, t_i - t_{i-1} = T/n, i \in [1, n]$. In the following tests we set hedging times $n = 30$.

As the optimal problem defined in 2.2, we aim to find the optimal position holding series in order to minimize the loss function under certain constraints. Assume that for any given market setting $MS = (MS_k), k = 1, 2, \ldots, n$, there always exists an optimal hedging strategy $\Delta_t = (\Delta_{t_k}), k = 1, 2, \ldots, n$ to minimize the objective function, e.g.

$$\forall\, MS, \exists f, s.t.\ \Delta_t = f(MS) \text{ to minimize } \mathbb{E}(\mathrm{l}(\mathrm{W_T}(\Delta_t))). \tag{9}$$

An approach to resolve this optimization problem is the reinforcement learning (RL) method. According to Sutton and Barto (1998), Kolm and Ritter (2019), Buehler, Mohan *et al.* (2019), we develop the RL system as follows.

(1) Agent (AG): option market maker, it interacts with environment and decides which strategy to execute.

(2) Environment (E): the environment at each point is defined as the market information at that time, which includes underlying prices, derivative prices, historical prices, underlying position prior to this moment, transaction cost proportion, trading limit, cost of funds, etc.

(3) State space (D): to derive the hedging strategy for European option, the state d_t at time t should contain underlying price S_t, underlying position prior to this moment Δ_{t-1}(the initial underlying position is set to 0), market friction parameters (transaction cost proportion ϵ etc). The state space D consists of all states, which can be defined as follows:

$$D := \mathbb{R}_+ \times \mathbb{R} \times \mathbb{R}_+ = \{(S_t, \Delta_{t-1}, \epsilon),\ 0 < t \leq T\}. \tag{10}$$

(4) Action space ($A = \{a_t\}$): a_t means the amount of underlying that the market maker longs or shorts at time t. For example, if the market maker longs n shares of underlying assets, $a_t = +n$; and if the market maker shorts n shares of underlying assets, $a_t = -n$. Thus the underlying position at time t: $\Delta_t = \Delta_{t-1} + a_t$. This is determined by liquidity constraints and trading limit: a market maker cannot long or short an asset without a limit. It is a bounded interval in $\mathbb{R}$.

(5) Policy (π): it determines the action a_t based on the state $d_t \in D$. It is a mapping from any state to the distributions over the actions. That is:

$$\pi : D \rightarrow \Delta(A). \tag{11}$$

(6) Reward (R): the reward at each time comes from positive cash flow and negative cash flow. The positive one includes the price change of underlying in the portfolio and income generated from derivative's sales (only at time 0). The negative one includes contingent liability due to short position in underlying (only at time T) and transaction cost. To be consistent with RL objective (usually, maximize objective function), we, therefore, reverse the symbol of objective function proposed in 2.1. The reward R_{t_i} at time t_i can be rewritten as:

$$R^\pi_{t_i} = \begin{cases} -p_0, & i = 0 \\ -\Delta^\pi_{t_{i-1}} \cdot (S_{t_i} - S_{t_{i-1}}) & \\ +\epsilon \cdot S_{t_i} \cdot |\Delta^\pi_{t_i} - \Delta^\pi_{t_{i-1}}|, & 0 < i < n \\ \max(S_T - K, 0), & i = n \end{cases}. \tag{12}$$

Considering the cost of funds, negative cash flow should be appended to the reward:

$$R^{\pi'}_{t_i} = R^\pi_{t_i} + M. \tag{13}$$

(7) Accumulated reward (G): the sum of rewards to receive in the future ($> t_i$):

$$G^\pi_{t_i} = R^\pi_{t_{i+1}} + R^\pi_{t_{i+2}} + \cdots + R^\pi_{t_n}. \tag{14}$$

(8) Value function (v): expectation of accumulated reward, in this article, the expectation is defined as that under

spectral risk measure:

$$v_{t_i}^{\pi} = \mathbb{E}(l(G_{t_i}^{\pi} \mid d_{t_i} = d)). \tag{15}$$

The objective of RL is to find optimal strategy π^*:

$$v_{t_i}^{\pi^*}(d) = \sup_{\pi \in \Delta(A)} v_{t_i}^{\pi}(d). \tag{16}$$

Our model aims to find the deterministic actions, therefore, deterministic policy needs to be applied. We propose a direct policy search to solve the optimization problem and find the best hedging strategy. The action space of this model is continuous, therefore, it is not possible to derive values for all actions, only the gradient descent method can be used to converge to the optimal strategy.

However, the action a^* derived from the optimal policy π^* of gradient descent method may not be in the feasible action space A, which means:

$$\exists d, s.t. \pi^*(d) \to a^* \notin A. \tag{17}$$

In order to address this issue, we could add the constraints to reward as penalty terms and search the optimal strategy for reward with penalty. A feasible action space must satisfy:

$$\begin{cases} C + W_t > 0 \\ \Delta_{t_i} - \beta \Delta_{t_i}^{BS} > 0 \end{cases}. \tag{18}$$

The reward with penalty can be written as follows:

$$R_{t_i}^{\pi''} = R_{t_i}^{\pi} + P(C + W_t) + P(\Delta_{t_i} - \beta \Delta_{t_i}^{BS}), \tag{19}$$

where $R_{t_i}^{\pi''}$ is the reward with penalty, $R_{t_i}^{\pi}$ is the reward defined in term 5. $P(x)$ is the penalty function, when $x > 0$, there is no penalty which leads to $P(x) = 0$; when $x < 0$, there is positive penalty which results in $P(x) > 0$. A possible function can be defined as:

$$P(x) = 1000(|x| - x). \tag{20}$$

We try to directly establish the relation between policy and state: because the neural network could approximate any non-linear function arbitrarily well, we could assume $\pi(d) = f_\theta(d)$ with deterministic parameters θ, input state k and output states a. Next, we will give an overview of how this function is derived:

(1) Network Construction: Due to computation power, we do not hope the policy at one time depends on the whole state space D. It is feasible to assume that the strategy at one time is mainly determined by the state at that time (the underlying price, etc.), thus the neural network can be defined as a simple fully connected ANN (Artificial Neural Network): $f_\theta = f_\theta(d_t)$. In the presence of transaction cost, frequent position adjustments will increase the market maker's cost, while reducing adjustment frequency will result in tracking error. Therefore, the optimal policy should be influenced by the current underlying position, which means the simple ANN is not enough for us. The neural network should depend on the input of current node and output of previous node: $f_\theta = f_\theta(d_t, \Delta_{t-1})$, LSTM-RNN (Recurrent Neural Network). Under some path-dependent cases, such as barrier option hedging or existence of liquidity constraint, the output states of past nodes have a significant impact on optimal hedging strategy. For example, if liquidity risk exists, the neural network should be defined as $f_\theta = f_\theta(d_t, \Delta_{t-1}, \Delta_{t-2}, \Delta_{t-3}, \ldots, \Delta_0)$. It means that an internal state based on past information is essential for strategy generation. Therefore, we need a special LSTM-RNN network, which can perform well even in long sequences. LSTM (Long Short Term Memory, introduced by Hochreiter and Schmidhuber (1997)) network is the best choice to transmit the previous messages. A 3-layer basic LSTM neural network is used in the following numerical tests.

(2) Parameter Training: The parameters of neural network are the ensembles of nodes weights in each layer. Evidently, training θ is to train the weights of nodes. The simplest way to train these parameters is a gradient descent method and its basic idea is:
   (a) Random initialization of parameter of θ;
   (b) Generate the interaction process by the current strategy f_θ and calculate $\nabla_\theta v_t(\theta)$;
   (c) Set learning rate α, update parameter $\theta = \theta + \alpha \nabla_\theta v_t(\theta)$;
   (d) If the parameter converges, finish training process, otherwise, turn to step 2.

(3) Training Data: Our algorithm aims to find the optimal underlying position series given underlying price paths, thus the data used to train network parameters are underlying price paths. For simulated market experiments, we rely on the constant volatility model and the stochastic volatility model to conduct Monte Carlo simulation to generate price paths. For real market experiments, due to the lack of historical data, we generate price paths by constant volatility model where volatility is estimated as one-year historical volatility and drift parameter is calculated by one-year historical data.

Here we use Adam algorithm (introduced by Kingma and Ba (2014)) to train LSTM network. The algorithm can thus be described as the following.

(1) Generate underlying price paths by Monte Carlo or fetch historical underlying price.
(2) Randomly choose 80% of the underlying paths as in-the-sample (ITS) data, and 20% as out-of-the-sample (OTS) data.
(3) Create an empty LSTM neural network, set the initial state to 0 then reshape to dimension (time step, batch size). Define objective function (loss function in 2.2) based on input underlying price paths.
(4) Feed the network with ITS data and train the network by Adam to minimize the objective function.
(5) Run the model on OTS data to get the CVaR value.
(6) Calibrate benchmark model on ITS data, then run these models on OTS data to calculate CVaR.
(7) Compare CVaR derived from the LSTM-RNN model and benchmark models.

3. Results

Having introduced the option hedging optimization problem in Section 2.2 and optimization problem solving method in Section 2.3, now we will conduct numerical experiments in this section and show how our method is more feasible and more accurate compared to previous option hedging strategies with market friction. In this section, we will considered vanilla option hedging problem, e.g. the agent shorts a vanilla call and hedge it with underlying asset, thus the payoff function in Section 2 can be defined as: $h(S_T) = \max\{S_T - K, 0\}$, where K is the strike price.

To begin with, we introduce three option hedging strategies under transaction cost based on BSM (Black–Scholes–Merton) model as the benchmark; Then we will turn to numerical experiments on simulated data generated by the GBM model and Heston model; Lastly, we will test our algorithm on real market data and compare the result with that of the benchmark model.

3.1. Benchmark models

As extensions of the Black–Scholes pricing formula, there are many models aiming at finding optimal strategies of discretely replicating an option with transaction costs. The most representative of these models are Leland (1985) model, Boyle and Vorst (1992) model and (Whalley and Wilmott 1993) model.

Leland (LL) proposed a corrected volatility to cover hedge error caused by transaction cost and he proved that the corrected volatility is equal to:

$$\widehat{\sigma}^2 = \sigma^2\left(1 + \sqrt{\frac{2}{\pi}}\frac{\epsilon}{\sigma}\sqrt{\delta t}\right), \tag{21}$$

where $\widehat{\sigma}$ is the corrected volatility taking account the transaction cost, σ is the volatility of the underlying asset, ϵ is the proportional transaction costs, δt is the time interval.

Boyle–Vorst (BV) introduced an option pricing method based on binary tree model and derived a corrected volatility under transaction cost:

$$\widehat{\sigma}^2 = \sigma^2\left(1 + \frac{2\epsilon\sqrt{n}}{\sigma\sqrt{T}}\right), \tag{22}$$

where n is the total hedging times and T is the time to maturity.

Whalley–Wilmott (WW) model was an extension of Davis' model, solving a three-dimensional free boundary problem by finding its asymptotic solution. They created a hedging bandwidth in the model, if the previous position is beyond the edge of the hedging bandwidth, the agent needs to long or short underlying asset. The hedging bandwidth is defined as $[\Delta_t - B_t, \Delta_t + B_t]$ where Δ_t is greek *Delta* calculated by Black–Scholes model at time t and B_t is defined as:

$$B_t = \left(\frac{3\epsilon S_t e^{-r(T-t)}\Gamma_t^2}{2\gamma}\right)^{\frac{1}{3}}, \tag{23}$$

and the corresponding hedging strategy is as follows.

$$y_t = \begin{cases} \Delta_t - B_t, & y_{t-1} < \Delta_t - B_t \\ \Delta_t + B_t, & y_{t-1} > \Delta_t + B_t \\ y_{t-1}, & \Delta_t - B_t \le y_{t-1} \le \Delta_t + B_t \end{cases}, \tag{24}$$

where y_t is the position that the agent needs to hold at time t, γ is the risk aversion coefficient, Γ is the greek Gamma in BSM model.

3.2. Simulated market experiments

Monte Carlo method is particularly useful in evaluating risks and uncertainty because it can simulate stochastic process in given parameters. In this section, we will apply geometric Brownian motion model (constant volatility model) and Heston model (stochastic volatility model) to simulate stock price separately, then derive the hedging errors for benchmark models and the model presented in this paper. The following is a brief reminder of the models:

- In the GBM model, stock price follows:

$$\mathrm{d}S_t = \mu S_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}W_t, \tag{25}$$

where S_t is the stock price at time t, μ is the market drift and σ is stock's volatility, W_t is the Wiener process. For time interval $t \in [0, T]$, let $t_0 = 0, t_n = T, t_i - t_{i-1} = T/n, i \in [1, n]$. With this setting, the formula can be discretized as:

$$S_{t_i} = S_{t_{i-1}} \exp\left[\left(\mu - \frac{\sigma^2}{2}\right)(t_i - t_{i-1}) + \sigma Z_i\sqrt{t_i - t_{i-1}}\right] \quad \text{for } i \in [1, n], Z_i \sim N(0, 1), \tag{26}$$

We could then apply this formula to generate underlying price paths.

- In the Heston model, stock price follows:

$$\mathrm{d}S_t = \mu S_t\,\mathrm{d}t + \sqrt{v_t}S_t\,\mathrm{d}W_t^S, \tag{27}$$

$$\mathrm{d}v_t = \kappa(\theta - v_t)\,\mathrm{d}t + \xi\sqrt{v_t}\,\mathrm{d}W_t^v, \tag{28}$$

where μ is the market drift, θ is the long run average price variance, κ is the rate of mean reversion of v_t to the long run average, ξ is the vol of vol. The correlation between the two separate Wiener processes W_t^S and W_t^v is ρ:

$$\mathrm{d}W_t^S\,\mathrm{d}W_t^v = \rho\,\mathrm{d}t, \tag{29}$$

Unlike the GBM model, there is no explicit solution for Heston SDE, thus one must use a numerical approximation to obtain both paths (v_t and S_t). A common method utilized is known as Euler Discretization. With the same setting of GBM model,

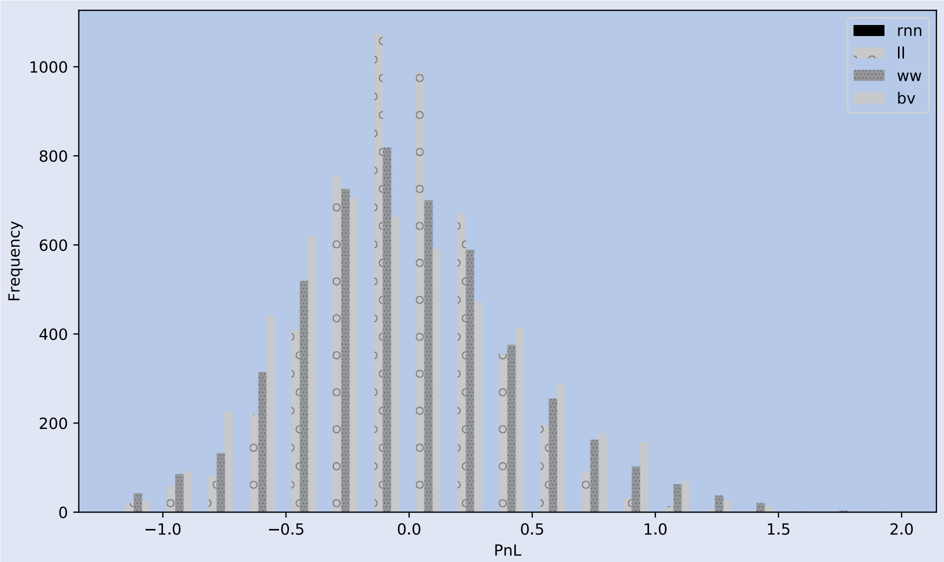

Figure 1. PnL distribution for different models.

the full truncation scheme could effectively give the underlying price path and volatility path:

$$S_{t_i} = S_{t_{i-1}}\exp\left[\left(\mu - \frac{1}{2}v^+_{t_{i-1}}\right)(t_i - t_{i-1}) + \sqrt{v^+_{t_{i-1}}}\sqrt{t_i - t_{i-1}}z_1\right], \tag{30}$$

$$v_{t_i} = v_{t_{i-1}} + \kappa(\theta - v^+_{t_{i-1}})(t_i - t_{i-1}) + \xi\sqrt{v^+_{t_{i-1}}}\sqrt{t_i - t_{i-1}}z_2, \tag{31}$$

where $x^+ = \max(x, 0)$, z_1 and z_2 are two standard normal random variables with correlation ρ.

We implement a 3-layer basic LSTM neural network in TensorFlow and train the model by two volatility models separately. The training parameters are:

- Architecture: input layer with 60 nodes, two LSTM cells of size 30, one linear layer with 30 nodes, output layer with 1 node.
- Optimizer: Adam with learning rate = 0.005 and batch size = 1000.

After each training, we will modify the stock price paths, option strike price and risk level to compare the feasibility and accuracy of the LSTM-RNN model with the benchmark.

3.2.1. GBM model setting. In this section we will train the LSTM-RNN model using market prices generated by constant volatility model. Note that in previous studies no experiments on vanilla options hedging with transaction costs in GBM model were performed, in particular, not using LSTM-RNN. To run Monte Carlo simulation, we need some basic market settings. Suppose we daily hedge an ATM option with one month to mature, the training parameters are (market drift and stock volatility are presented in annualized terms): initial stock price S_0: 100, market drift μ: 0.1, stock volatility σ: 0.2, time horizon T: 1/12 (one month), time steps dt: 1/360 (one day), simulation times: 5000 where 4000 paths are ITS data and 1000 are OTS data.

Table 1. In-the-sample data CVaR.

Model	CVaR	Mean	Standard deviation	MC error
LSTM-RNN	0.83	− 0.03	0.41	0.03
Leland	0.81	− 0.04	0.36	0.02
Whalley–Wilmott	1.51	− 0.03	0.48	0.03
Boyle–Vorst	0.80	− 0.04	0.48	0.03

Thus the stock price can be generated by the parameters above. Then we need to calculate hedge error of benchmark strategies. To obtain hedging strategies of Leland model and BV model, we need transaction cost proportion, hedging time interval, total hedging times, strike price of option (to calculate delta and gamma). For the WW model, we need to know trader's risk aversion rate. Following are the parameters: strike price K: 100 (ATM), transaction cost proportion ϵ: 0.02%, hedging time interval Δt: 1/360 (daily rebalancing), total hedging times n: 30, confidence level α: 0.99. To set the LSTM-RNN model, the initial option price p_0 is defined as the option price derived from BSM model and the risk aversion rate $\gamma = \frac{1}{1-\alpha}$.

3.2.2. GBM model results. Figure 1 shows the final PnL of different models for OTS data (1000 paths). It looks like for these four models, the PnL follows a normal distribution. PnL for Leland model is the most concentrated and its variance is the smallest. WW model and BV model have a dispersive distribution. The LSTM-RNN model is a bit worse than the Leland model. Table 1 shows basic statistical information of the PnL distribution including mean and standard deviation, Monte Carlo error (MC error) and CVaR loss (defined in Section 2.2) for each model. The MC errors reported below correspond to 95% confidence levels and a fixed constant as 1.96.

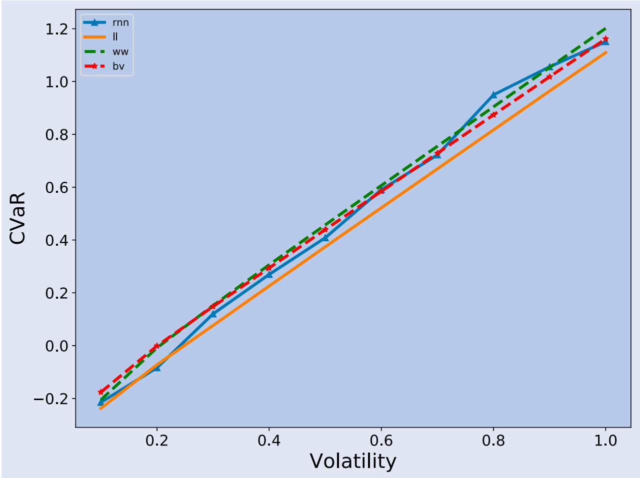

Figure 2. CVaR loss increases when stock volatility goes up.

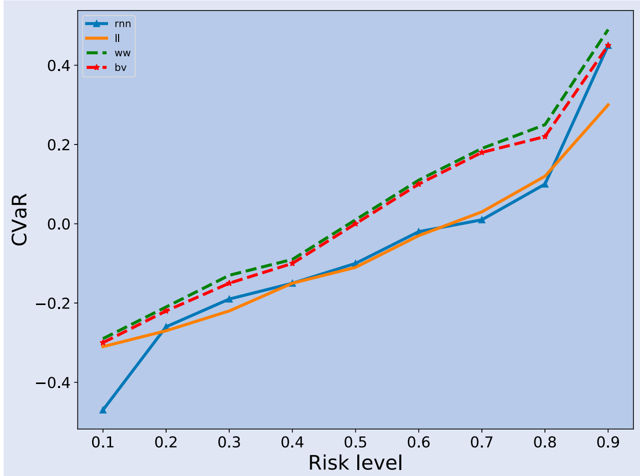

Figure 4. CVaR increases when the risk level rises.

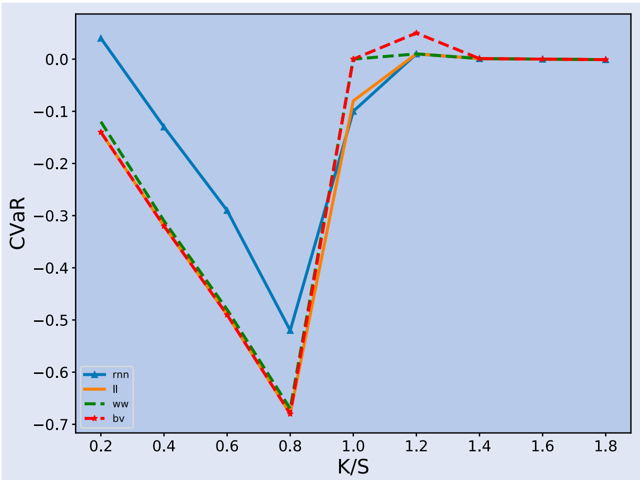

Figure 3. CVaR loss has the lowest value when moneyness = 0.8.

Table 1 shows that the average loss (mean) and simulation error (MC error) are quite similar for these four models, however, CVaR loss shows different properties. At 99% risk level, BV model's loss is the smallest among these four models and WW model's loss is the biggest. The LSTM-RNN model's performance is close to the Leland model. From table 1 and figure 1, we can safely draw the conclusion that at 99% risk level, our algorithm performance is close to the Leland model and BV model's CVaR. To analyze if more samples would improve the performance of LSTM-RNN, we generate 1000, 5000, 10 000, 50 000 and 100 000 paths and find that the corresponding CVaRs are 2.36, 0.83, 0.79, 0.67 and 0.65 for OTS sample and the MC errors are 0.05, 0.02, 0.01, 0.007 and 0.004. This demonstrates that increasing training samples could significantly improve LSTM-RNN's performance. However, due to the computational power, we will choose 10 000 for the following tests.

In order to test the feasibility of the LSTM-RNN model, we will change stock price, strike price, risk level, proportional transaction costs and then compare the CVaR loss of different strategies. The results are shown in figures 2 and 3.

Figure 2 shows that CVaR loss increases when stock's volatility rises up for all four models. The sensitivities of benchmark models are nearly the same, but the Leland model has a relatively low intercept, which makes it the most accurate among the four models. The LSTM-RNN model's accuracy is very close to that of Leland model at low volatility ; however, when stock volatility reaches a high level (>0.8), the LSTM-RNN model's performance is closer to those of WW model and BV model. It can be inferred from the figure that the LSTM-RNN model is less robust, when stock's volatility is low, the LSTM-RNN model performs well; but when the volatility becomes too large, we are probably exposed to larger risk.

As we can see in figure 3, CVaR loss decreases from deep ITM to close ATM; then there is a sharp rise for moneyness varying from 0.8 to 1; from ATM to OTM the CVaR loss is nearly 0. In this scenario, benchmark models' losses are nearly the same; the LSTM-RNN model's loss is slightly bigger than them, which means the LSTM-RNN model performs slightly worse than benchmark models for ITM options. But in reality, when CVaR loss is negative it means that we have a gain on average so for ITM option, the LSTM-RNN model gains less than benchmarks; and for OTM options, performances are nearly the same.

It can be inferred from figure 4 that when the risk level rises, the CVaR loss will be larger than before. Four models show similar properties. Leland is the most stable one, even at high-risk level, Leland's CVaR loss does not increase that much. CVaR for WW model and BV model are close: it rises up rapidly when the risk level increases. However, when at a relatively low (<0.2) or high level (>0.8), the LSTM-RNN model's CVaR loss rises up quickly but when the risk level is in the middle, the LSTM-RNN model does not change so much. From this figure, we can suggest that when trader's risk level is high (>0.8), we can take the Leland model as our optimal strategy; when it is smaller than 0.8, we will use the LSTM-RNN model as the best strategy to hedge the option.

Figure 5 demonstrates that the hedging error is proportional to transaction cost proportion. The linear relationship can be predicted given the formulas of the models. LSTM-RNN model is the most solid when transaction cost proportion

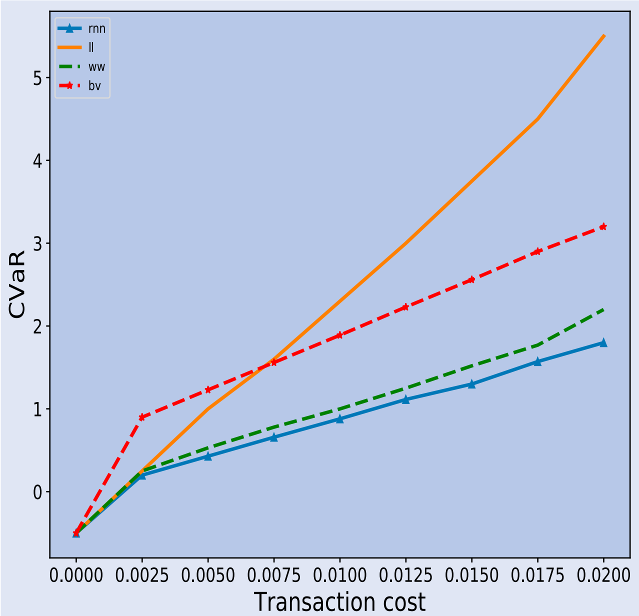

Figure 5. CVaR loss is proportional to transaction cost proportion.

increases and the Leland model is the worst one facing augmentation of the transaction cost. Therefore, if we want a model which is not sensitive to transaction cost, LSTM-RNN is our best choice.

To sum up, our hedging strategy (LSTM-RNN) performs better than the benchmark in low or medium volatility (<0.8), OTM moneyness and certain risk level ($<80\%$), besides the LSTM-RNN model is not sensitive to transaction cost. The most stable model, however, is the Leland model, this model is the least sensitive to volatility and risk level.

3.2.3. Heston model setting and data. Heston model is a stochastic volatility model with more parameters to calibrate. Unlike the GBM model, the constant volatility assumption is not valid, thus models like Leland, BV and WW can no longer be used as benchmark. Hence, in this section, we will compare hedging error of the Heston model and LSTM-RNN model.

Heston gave us a closed-form solution for European option, which can be used as our initial option price. However, Δ cannot be derived easily as in Black–Scholes model because closed-form solution does not exist. Here we approximate delta of Heston model by its definition:

$$\Delta_{\text{Heston}} \approx \frac{C(S+\Delta S) - C(S)}{\Delta S}. \tag{32}$$

The following are the parameters for Heston model: initial stock price S_0: 100, initial variance v_0: 0.01, market drift μ: 0.1, long run variance θ: 0.04, mean reversion rate κ: 1, volatility of volatility ξ: 0.1, the correlation coefficient between stock price and volatility ρ: -0.7, time horizon T: 1/12(one month), time steps dt: 1/360(one day), simulation times: 10 000 where 8000 paths are ITS data and 2000 are OTS data.

Thus the stock price and variation can be generated by the parameters above. Like in the settings above, we need transaction cost proportion, hedging time interval, total hedging times, strike price of option to calculate hedge error. The parameters are: strike price K: 100 (ATM), transaction cost proportion ϵ: 0.02%, hedging time interval Δt: 1/360 (daily rebalancing), total hedging times n: 30

To set the LSTM-RNN model, initial option price p_0 is defined as the option price derived from Heston model and risk level α is 0.5.

3.2.4. Heston model results. For OTS data (2000 paths), the PnL for Heston model and LSTM-RNN model is shown in figure 6.

As we can see in figure 6, for the LSTM-RNN model, the error is normally distributed and the PnL is from -12 to 5. For Heston model, more than 1750 out of 2000 samples are around 0, but the extreme scenario leads us to a loss of 58. From CVaR perspective, Heston model's CVaR is larger than that of LSTM-RNN (8.26 compared to 5.63) and their MC errors are very similar (0.31 compared to 0.26). We could then conclude that Heston model has a better accuracy and risk of LSTM-RNN model is easier to control because it is bell-shaped and looks like it follows normal distribution; the risk of Heston model, on the other hand, is far more complicated because the worst case drives a significant loss.

Comparing the result of this model to benchmark models presented in the previous section, we may draw the conclusion that the LSTM-RNN model is much easier to generalize. It is not necessary to build complicated mathematical models for each market assumptions, instead, you may just put stock price paths and train certain parameters which will give you an optimal hedging strategy.

3.3. *Real market results*

In this section, we conduct five experiments in the real market setting to verify if our algorithm is feasible and robust compared to benchmark models. We analyze markets all over the world by comparing the CVaR loss of different strategies, including Asian market (domestic market 50 ETF option, Hong Kong market Hang Seng Index Option, Nikkei Index Option), North American market (S&P 500 Index Option) and European market (FTSE 100 Index Option).

3.3.1. 50 ETF option. 50 ETF Option is the first equity-linked option traded publicly on Chinese exchange, starting in February 2015. As its name implies, the underlying security is the SSE 50 ETF. As it is one of the two equity options traded on exchange, it has a huge volume and it makes sense to analyze the hedging strategy of market maker, option issuer and option seller.

Here we take the 50 ETF option in 2018 as our object of study and we take 30 trading days before the expiry date as the initial issuing date. For example, in November the option expiry date is 11/28/2018 and the initial issuing date is 10/28/2018. The initial price of 2018 November's 50 ETF is 2.419. To test the CVaR loss for different models including the model presented in this paper and the benchmark models, we need to set the required parameters.

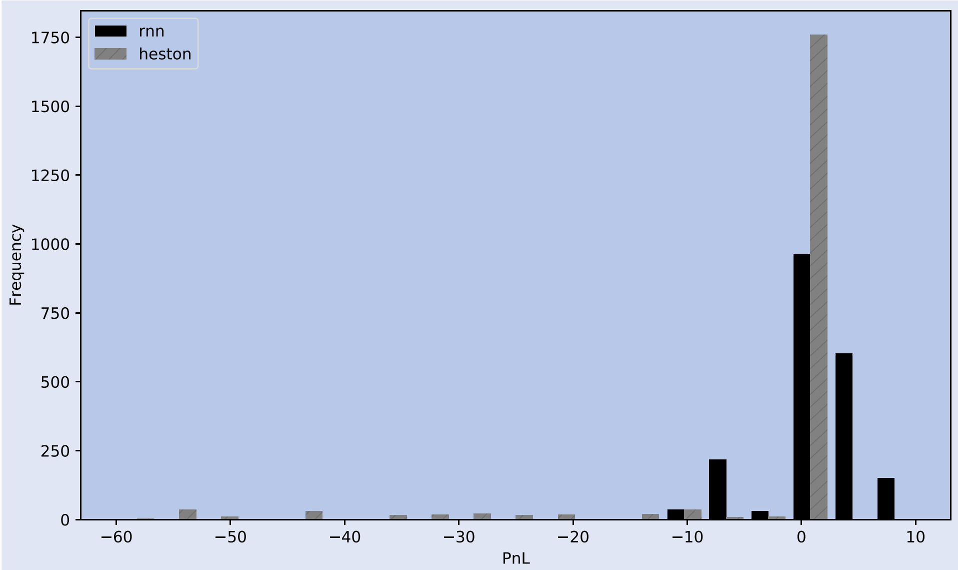

Figure 6. PnL distribution for LSTM-RNN model and Heston model.

For benchmark models, the volatility and market drift need to be determined. We use historical volatility in the last 12 months as the constant volatility in the benchmark models and historical market drift of the last 12 months data as the market drift. Here we consider an ATM option, therefore the strike price for November's option is set to 2.4. Transaction cost proportion is relatively small, we set the value to 0.02%. As in the previous setting, we adjust our stock position at the end of each day thus daily rebalancing.

As mentioned in Section 2.3, we train LSTM-RNN parameters by underlying price paths generated from GBM model. The volatility and market drift used in GBM are the same as those in benchmark models, which are historical volatility and market drift of the last 12 months. The testing price path is the real market price.

Following shows the parameters used in the real market setting for option in November: initial 50 ETF price S_0: 2.419, market drift μ: 3.80%, stock volatility σ: 0.28, time horizon T: 1/12(one month), time steps dt: 1/360(one day), strike price K: 2.4 (ATM), transaction cost proportion ϵ: 0.02%, hedging time interval Δt: 1/360 (daily rebalancing), total hedging times n: 30, confidence level α: 0.5.

After calculation for the LSTM-RNN model and benchmark models, the CVaR loss is shown in the corresponding figure 7.

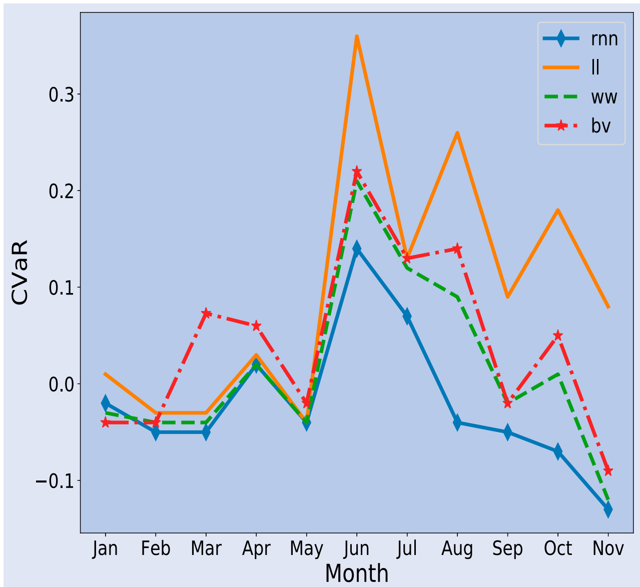

Figure 7. 50ETF-Hedging error of different strategies in different months.

The figure shows that the LSTM-RNN model performs well in real market setting. The hedging errors for all models follow the same pattern: CVaR loss reaches to a peak in June and decreases from June to November. This pattern can be explained by huge volatility in June in the Chinese stock market. WW model, BV model, Leland model and LSTM-RNN model have a similar performance during the whole year. We can reach to the conclusion that the four models are almost equally sensitive to stock volatility in domestic market.

If we keep the settings above and change the strike and transaction cost proportion (November), the results can be shown in figures 8 and 9. In figure 8, benchmark models are not sensitive to moneyness change, their CVaR loss is relatively stable; however, the LSTM-RNN model performs the best for ATM option; while for deep OTM options, the LSTM-RNN model's loss is bigger than those of Leland model and BV model. It means that the LSTM-RNN model can well hedge the Gamma exposure for ATM options but not for OTM options. In figure 9, the CVaR loss rises when transaction cost increases. The LSTM-RNN model is the most stable method and the Leland model is the most unstable. When transaction cost is small, WW and BV models rise up quickly; then they increase slower than Leland. The LSTM-RNN model is the most stable among the four. Figure 10 shows the relation of CVaR to risk level. Leland model and BV model are not sensitive to the change of risk level. The WW model increases slowly when the risk level rises up. The LSTM-RNN model shows unstable properties compared to other models. When the risk level remains in a low level,

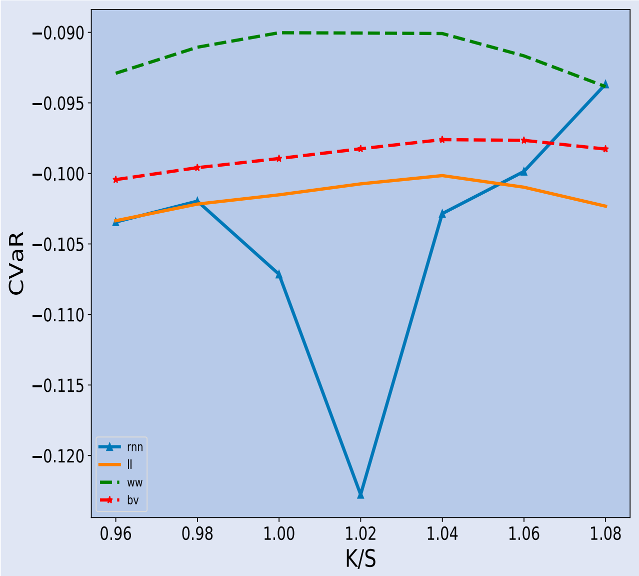

Figure 8. 50ETF-CVaR loss is stable for benchmark models but has the largest gain when ATM for LSTM-RNN model.

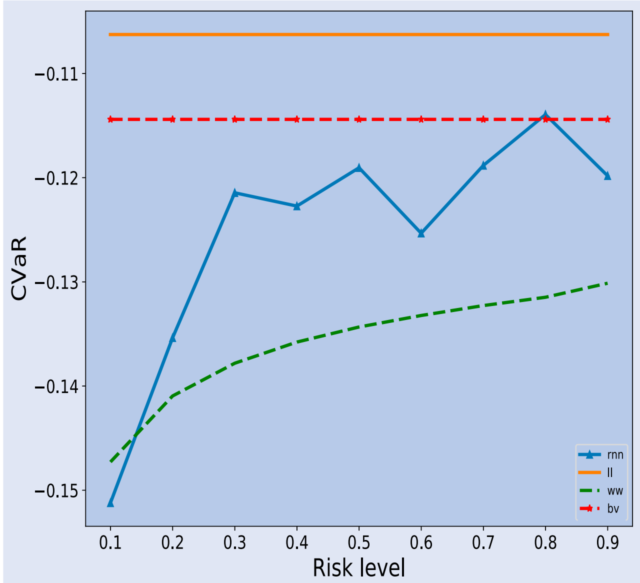

Figure 10. 50ETF-CVaR loss increases when the risk level goes up, but some models are more sensitive to risk level change.

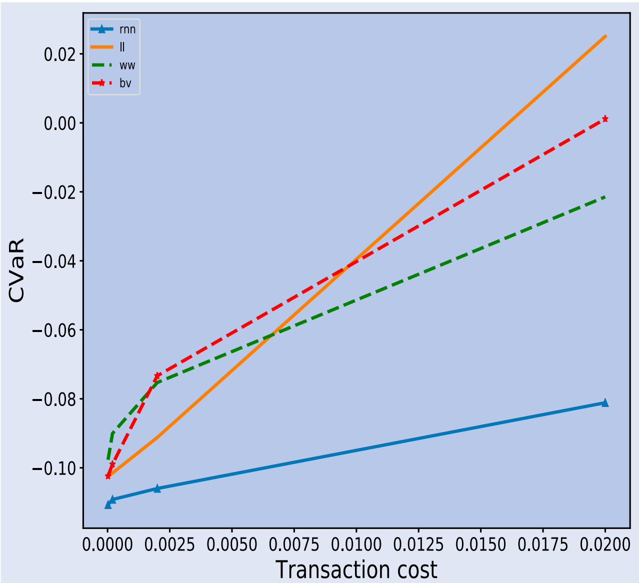

Figure 9. 50ETF-CVaR loss is proportional to transaction cost proportion.

CVaR for LSTM-RNN model increases fast; while when the risk level passes 0.3, CVaR begins to fluctuate around -0.125. For this market, the most robust model is WW model especially when facing a high-risk level.

3.3.2. Hang Seng index option. Hang Seng Index (HSI), the benchmark of the Hong Kong stock market, is one of the best known indices in Asia and widely used by fund managers as their performance benchmark. An HSI Option is an option contract based on the Hang Seng Index. Like in the previous section, we take the HSI option in 2018 as our study object and we take 30 trading days before expiry date as the initial issuing date. The volatilities are calculated for different months by the same method mentioned in Section 3.3.1.

For market drift, we also use past 12 months market drift. The following shows the parameters: initial Hang Seng index S_0: 25454.55, market drift μ: -2.43%, stock volatility σ: 0.174, time horizon T: 1/12(one month), time steps dt: 1/360(one day), strike price K: 25400(ATM), transaction cost proportion ϵ: 0.02%, hedging time interval Δt: 1/360 (daily rebalancing), total hedging times n: 30, confidence level α: 0.5.

CVaR losses of LSTM-RNN model and benchmark models for different months are shown in figure 11. From this figure, we can conclude that apart from January and June, the LSTM-RNN model has the lowest CVaR loss. BV model is slightly better than other benchmark models in terms of accuracy (BV model's CVaR is smaller than the other two). Similar to the analysis conducted in 3.3.1, two figures presenting relations between CVaR hedge error and strike price/transaction cost proportion could be obtained (figures 12 and 13).

In figure 12, the CVaR is still stable for traditional models. But the LSTM-RNN model's CVaR loss shows some different properties. The pattern is nearly the same as figure 3: decreases from deep ITM to ATM, and increases from close ATM to OTM. In reality, when CVaR loss is negative it means that we have a gain on average so the LSTM-RNN model actually makes profits; while other models make losses.

In figure 13, CVaR loss increases when transaction cost increases for all models apart from for BV model when transaction cost proportion is low (< 0.0025). The Leland model has the largest slope, which means that with the augmentation of transaction cost proportion, its CVaR loss increases the fastest. WW model has the second-largest slope. The BV model's hedging error increases the slowest when transaction cost increases. The LSTM-RNN model performs best among all models when transaction cost proportion is lower than 0.015, while when it is higher than 0.015, the BV model is the best.

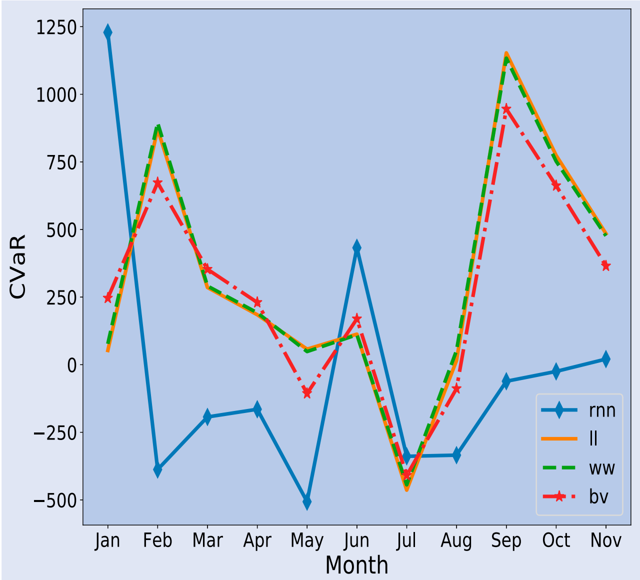

Figure 11. HSI-Hedging error of different strategies in different months.

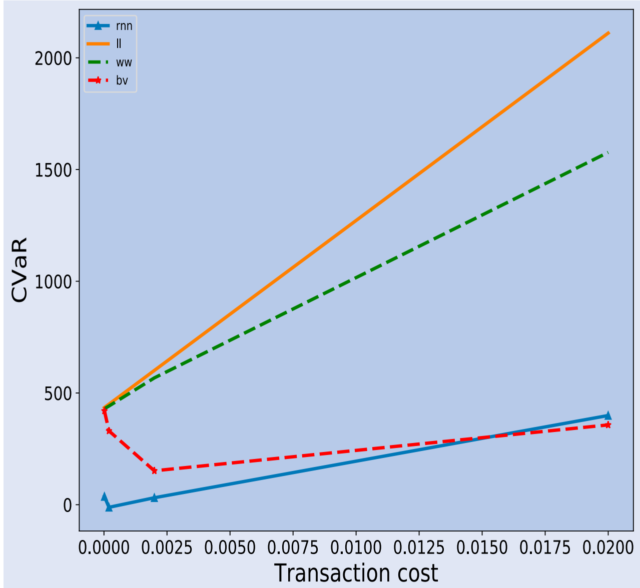

Figure 13. HSI-CVaR loss is proportional to transaction cost proportion.

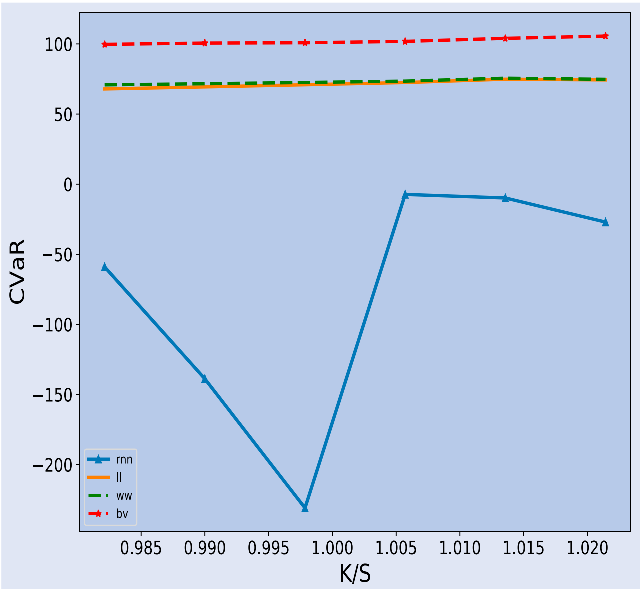

Figure 12. HSI-Only CVaR loss for the LSTM-RNN model varies when moneyness changes.

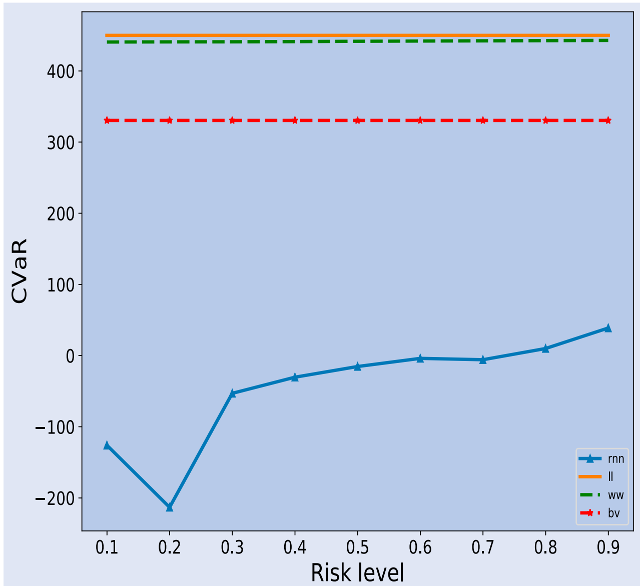

Figure 14. HSI-Only CVaR loss for the LSTM-RNN model varies when the risk level changes.

Comparing figures 12 and 14, we know that traditional models are not sensitive to risk level or moneyness for Hang Seng index because they barely change when moneyness or risk level changes. For these models, Leland and WW model have a similar loss, the BV model is more accurate. The LSTM-RNN model has the smallest CVaR and it increases when the risk level rises up.

3.3.3. Nikkei index option. Nikkei 225 Option is based on the Nikkei Stock Average which is a major Japanese stock index used as a benchmark in various financial instruments. The Nikkei Stock Average (Nikkei 225) is used around the globe as the premier index of Japanese stocks and is very popular with Japanese retail investors particularly. It is calculated and published by a Japanese newspaper publisher Nikkei Inc.

Similarly to the analysis conducted in 3.3.1 and 3.3.2, the Nikkei option in 2018 is our study object and 30 trading days before the expiry date is regarded as the initial issuing date. The volatility and market drift are calculated in the way presented in Section 3.3.1. The parameters are as follows: initial Nikkei 225 index S_0: 22658.16, market drift μ: 1.03%, stock volatility σ: 0.14, time horizon T: 1/12(one month), time steps dt: 1/360(one day), strike price K: 22625(ATM), transaction cost proportion ϵ: 0.02%, hedging time interval Δt: 1/360 (daily rebalancing), total hedging times n: 30, confidence level α: 0.5.

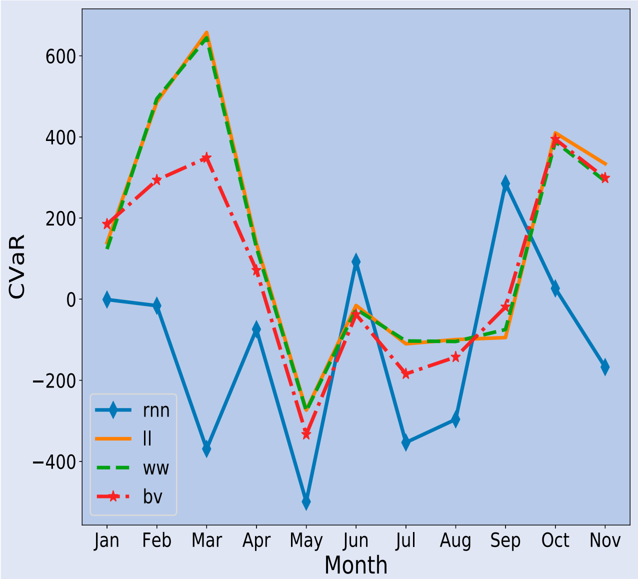

Figure 15. Nikkei–Hedging error of different strategies in different months.

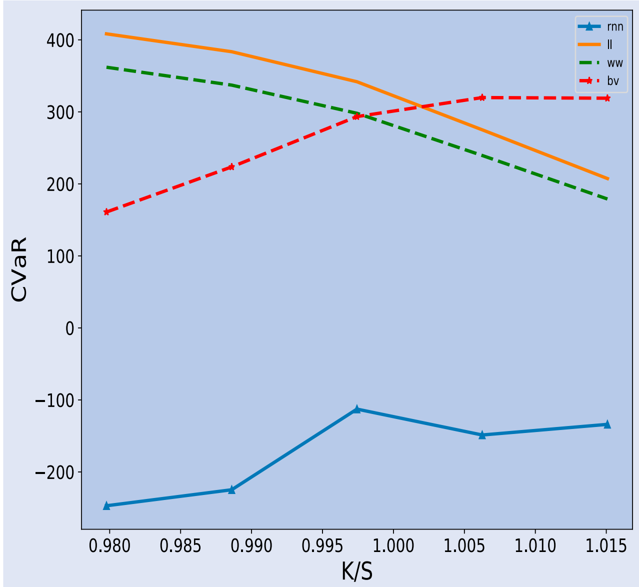

Figure 16. Nikkei-CVaR loss for BV and LSTM-RNN model increases when moneyness decreases, while others show opposite properties.

This is the setting for November, and if we apply the similar settings to the whole year by changing the initial index price (the corresponding price), stock volatility (calculated by historical volatility) and strike price (ATM price), a figure presenting the CVaR loss for different months can be obtained (figure 15).

Figure 15 is a fairly interesting one. The LSTM-RNN model has the highest loss in June and September. For other months it performs very well regarding the CVaR loss. WW model's performance is quite close to that of the Leland model. Usually, the BV model is stable as shown in the previous sections, yet in this figure, it shows weak robustness: In March and October, it reaches its peak. This phenomenon is probably caused by the Gamma risk of Nikkei index. Neural network may overfit the Gamma risk, therefore the model overreacts to small Gamma change. To verify the model's feasibility to moneyness and transaction cost proportion, we modify the strike price and transaction cost. Figures 16–18 show the corresponding results.

Comparing to figure 12, figure 16 shows that when moneyness decreases from ITM to OTM, the CVaR loss decreases for model Leland and WW; instead, for the BV model, this error increases. This is an interesting property; it shows some inversion effect compared to Hong Kong market. The LSTM-RNN model shows the same property as in figure 3, that's to say at ATM moneyness it is exposed to the largest possible gain.

Comparing to figure 7, the LSTM-RNN model's CVaR shows different properties in figures 11 and 15. In figure 7, the hedging error is proportional to transaction cost proportion and hedging error for these four models in descending order is: Leland, BV, WW and RNN. In figures 11 and 15; however, CVaR hedging error for the BV model decreases at first, then increases with augmentation of transaction cost proportion. Hedging error for these four models in descending order is: Leland, WW, BV and RNN. In this market, LSTM-RNN model seems to be less sensitive facing modification of

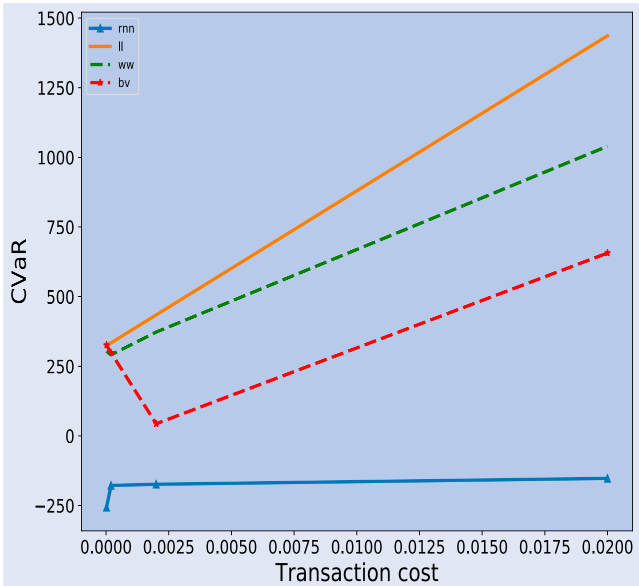

Figure 17. Nikkei-CVaR loss is proportional to transaction cost proportion.

transaction cost, which proves the feasibility of the LSTM-RNN model.

Figure 18 shows the same property as figure 14: Leland model, WW model and BV model are stable in terms of risk level and LSTM-RNN is sensitive to the change of risk level. In this graph, one unexpected observation is that when the risk level rises up, the CVaR decreases. This probably caused by the negative interest rate. The CVaR of the LSTM-RNN model for Nikkei Index option remains negative, which means we can always have a profit at the expiry date.

3.3.4. S&P 500 index option. The famous S&P 500 Index is widely regarded as the leading benchmark of the overall U.S.

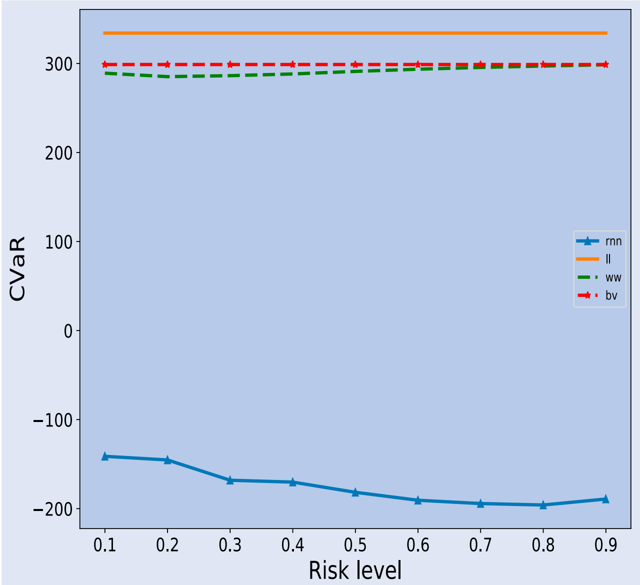

Figure 18. Nikkei-Only CVaR loss for the LSTM-RNN model is sensitive to risk level change.

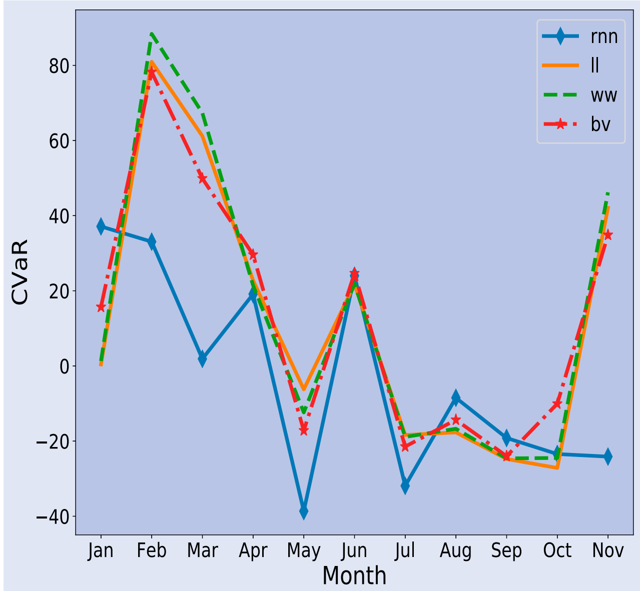

Figure 19. S&P 500-Hedging error of different strategies in different months.

stock market. Comprised of 500 leading companies, the index is considered the best single gauge of large cap U.S. equities. S&P 500 Index Option is evidently based on S&P 500 Index.

The necessary parameters that are calculated in the same way as in previous sections are as follows: initial S&P index S_0: 2768.78, market drift μ: 0.54%, stock volatility σ: 0.13, time horizon T: 1/12(one month), time steps dt: 1/360(one day), strike price K: 2770(ATM), transaction cost proportion ϵ: 0.02%, hedging time interval Δt: 1/360 (daily rebalancing), total hedging times n: 30, confidence level α: 0.5.

This is the setting for November, and if we apply similar settings to the whole year by changing the initial index price (the corresponding price), stock volatility (calculated by historical volatility) and strike price (ATM price), a figure presenting the CVaR loss for different months can be obtained (figure 19).

According to figure 19, we can see that the CVaR for benchmark models follows the same trend while the LSTM-RNN has a unique trend. Apart from January, August and September, LSTM-RNN model has the best hedging strategy regarding to CVaR because apart from these three months, the values of LSTM-RNN's CVaRs are smallest among all models. CVaR loss reaches the same peak in June for all four models. Models' patterns are quite similar, this is probably because the American stock market is a mature market, and the hedging error does not depend on models but on the estimation of volatility.

Like in the settings above, the strike price and the transaction cost proportion can be modified to verify the model's feasibility for American stock market, following figures show the results.

In figure 20, the relation between CVaR hedge error and strike price is different from figures 3, 8, 12 and 16. For benchmark models, the hedging error is the largest for ATM options, instead, for the LSTM-RNN model, the error is the smallest for ATM options. As a result, From ITM to ATM, the hedging error increases for benchmark models; vice versa for the

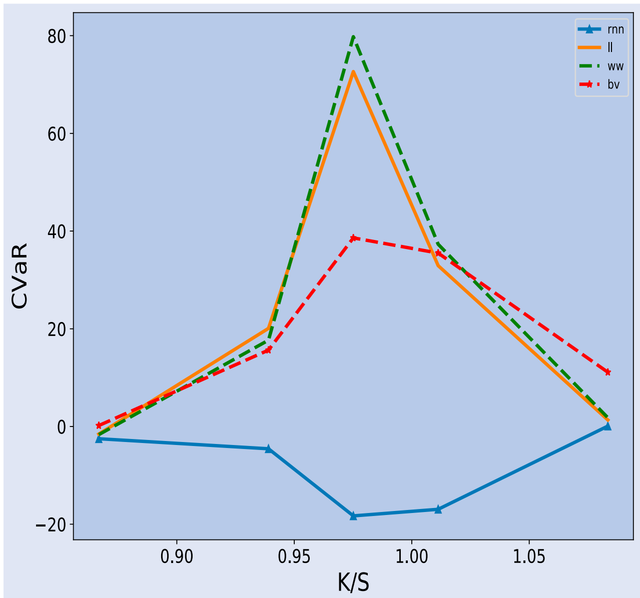

Figure 20. S&P 500-Relation between CVaR loss and moneyness of the LSTM-RNN model is different from that of benchmark models.

LSTM-RNN model. The trend is opposite from ATM to OTM. An explanation could be Gamma exposure as discussed in Section 3.3.1. Leland and WW model do not take Gamma into consideration thus have a huge Gamma exposure and reach loss peak for ATM option, while LSTM-RNN model is not sensitive to Gamma variation. Thus it stays at its lowest point for ATM option.

Figure 21 shows similar properties presented in previous figure 17. In this figure, hedging error is proportional to transaction cost proportion. Leland model has the largest slope, which means that with the augmentation of transaction cost proportion, the CVaR hedging error increases the fastest. The WW model has the second largest slope, the BV model has the third largest and the LSTM-RNN model has the smallest one.

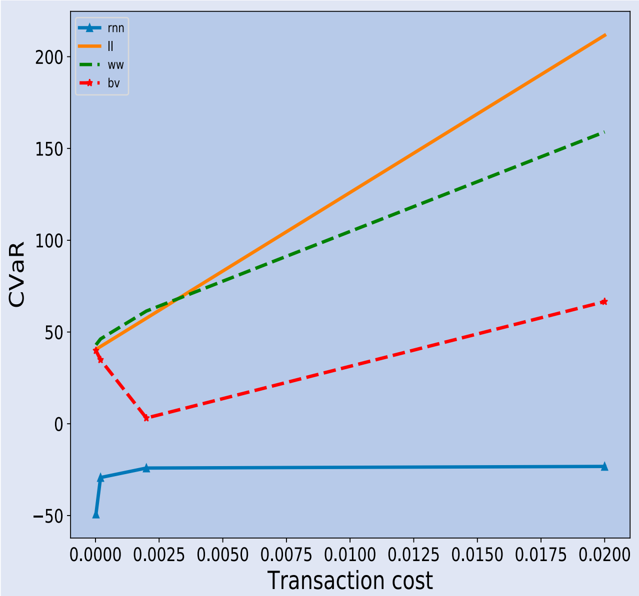

Figure 21. S&P 500-CVaR loss is proportional to transaction cost proportion.

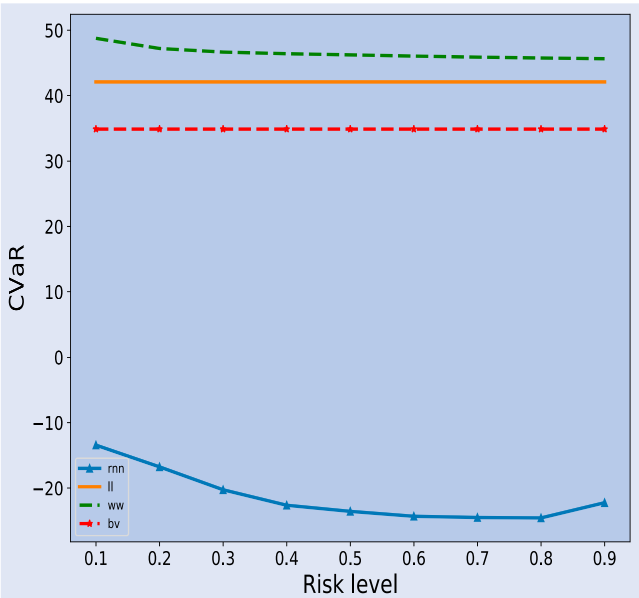

Figure 22. S&P 500-Only CVaR loss for the LSTM-RNN model is sensitive to risk level change.

As is stated in Section 3.2.2, the linear relationship can be predicted given the formulas of the models and the LSTM-RNN model is the most solid facing augmentation of transaction cost.

Figure 22 has the same pattern as figures 14 and 18. Traditional models have a constant CVaR no matter how the risk level changes. WW model has the largest CVaR, which results in the potential largest loss; Leland model has the second largest loss and BV model's CVaR is smaller than these two models. The LSTM-RNN model's CVaR loss, however, decreases when the risk level increases and smaller than 0.8, then it increases when the risk level ranges from 0.8 to 0.9.

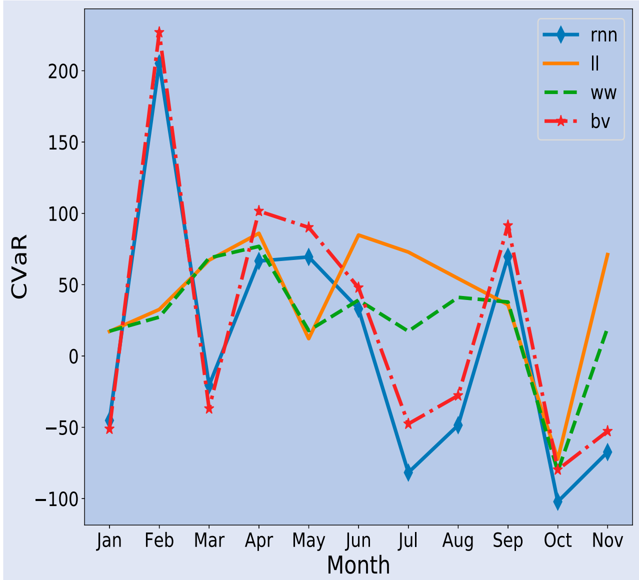

Figure 23. FTSE-Hedging error of different strategies in different months.

3.3.5. FTSE 100 index option. FTSE 100 index options are option contracts in which the underlying value is based on the level of the FTSE 100, the UK equivalent of the Dow Jones Industrial Average and tracks the performance of the top 100 largest companies by market capitalization listed on the London Stock Exchange. The necessary parameters are calculated in the same way as in previous sections and they are as follows: initial S&P index S_0: 7026.99, market drift μ: -0.38%, stock volatility σ: 0.119, time horizon T: 1/12(one month), time steps dt: 1/360(one day), strike price K: 7000(ATM), transaction cost proportion ϵ: 0.02%, hedging time interval Δt: 1/360 (daily rebalancing), total hedging times n: 30, confidence level α: 0.5.

This is the setting for November, and if we apply the similar settings to the whole year by changing the initial index price (the corresponding price), stock volatility (calculated by historical volatility) and strike price (ATM price), a figure presenting the CVaR loss for different months can be obtained (figure 23).

In figure 23, four models can be clearly divided into two groups: Leland model and WW model have the same trend, LSTM-RNN model and BV model have the same pattern. The first group's CVaR violates more than the second group. Like in the settings above, the strike price and the transaction cost proportion can be modified to verify the model's feasibility for American stock market, following figures 24 and 25 show the results.

In figure 24, for Leland model and WW model, the hedging error is the largest for ATM options, instead, for BV model and RNN model, the error is the smallest for ATM options. As a result, from ITM to ATM, the hedging error increases for Leland model and WW model; vice versa for BV model and RNN model. The trend is opposite from ATM to OTM. In figure 20, except for BV model, the other models have same properties as in figure 24. In both figures 21 and 25, hedging error is proportional to transaction cost proportion and LSTM-RNN has the smallest slope among all. As is stated

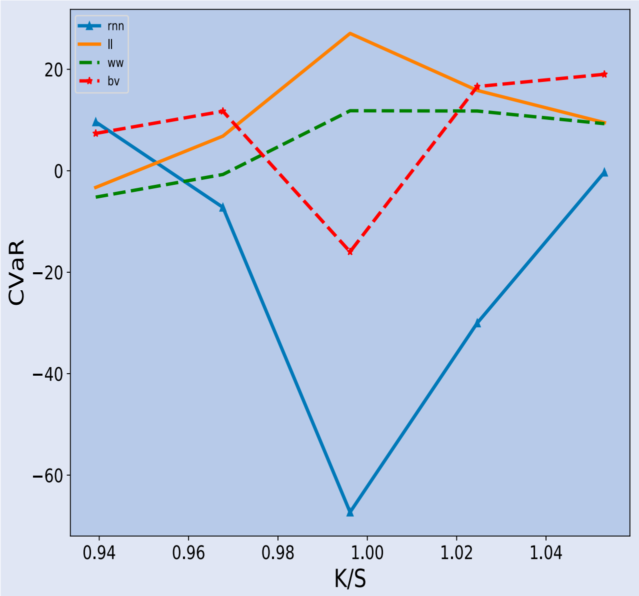

Figure 24. FTSE-Relations between CVaR loss and moneyness of LSTM-RNN model and BV model are the same.

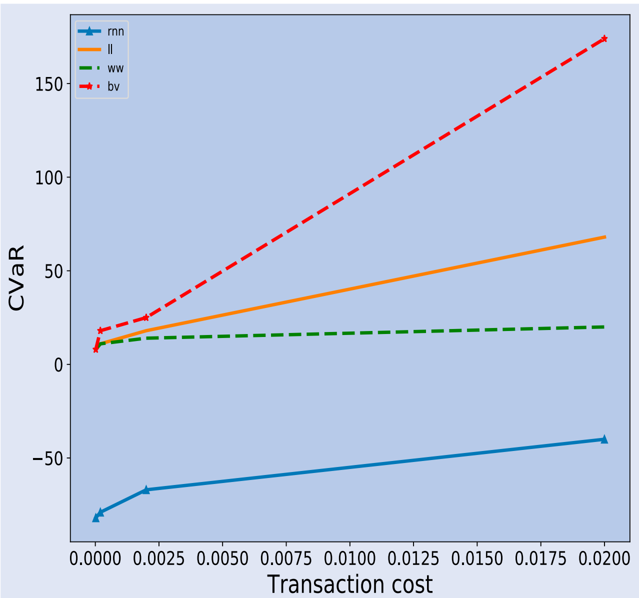

Figure 25. FTSE-CVaR loss is proportional to transaction cost proportion.

in Section 3.2.2, the linear relationship can be predicted given the formulas of the models and the LSTM-RNN model is the most solid facing augmentation of transaction cost. Figure 26 shows a similar result with figure 14: Leland, WW and BV models are not sensitive to the change of risk level; the ranking for CVaR loss is $\mathrm{CVaR_{LL}} > \mathrm{CVaR_{WW}} > \mathrm{CVaR_{BV}}$. For RNN model, its loss increases as the risk level goes up.

3.4. *Market frictions other than transaction costs*

In previous sections, we only consider the transaction cost as market friction. However, as we state in Section 2.1, apart from transaction cost, market friction includes also liquidity

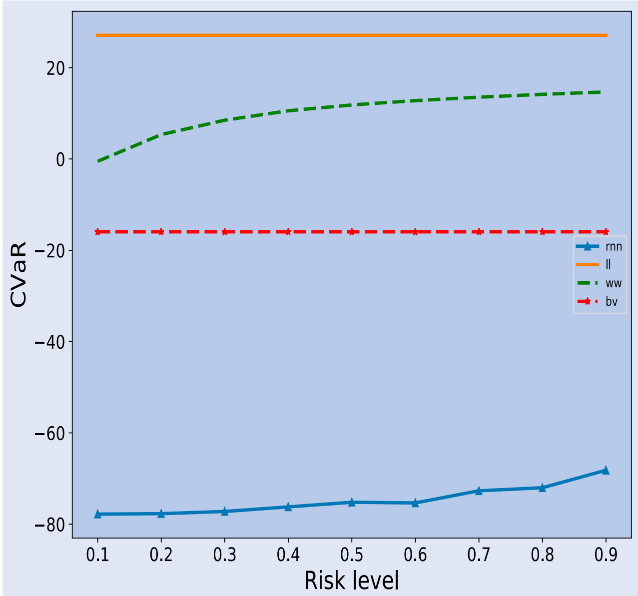

Figure 26. FTSE-Only CVaR loss for the LSTM-RNN model is sensitive to risk level change.

Table 2. Market settings for figures 27–31.

Figure	ϵ	C	β	ϵ'
27	0.02%	No limit	0	$0 \sim 1$
28	0.02%	No limit	$0 \sim 0.8$	0
29	0.02%	No limit	$0 \sim 0.8$	0
30	0.02%	$0\% \sim 120\%$	0	0
31	0.02%	$20\% \sim 120\%$	0	0

constraint, trading limits and cost of funds. In this part, we will show some primary results of hedging securities under these factors by the LSTM-RNN model. We do not introduce any benchmark models because there were any solid results, whether analytical or numerical, in previous studies.

We use the same market setting, as presented in Section 3.2.1, a traditional GBM stock price model with following parameters: initial stock price S_0: 100, market drift μ: 0.1, stock volatility σ: 0.2, time horizon T: 1/12(one month), time steps dt: 1/360(one day), simulation times: 5000, strike price K: 100 (ATM), hedging time interval Δt: 1/360 (daily rebalancing), total hedging times n: 30, confidence level α: 0.9.

And there are additional parameters presenting market friction: transaction cost proportion ϵ: 0.02%, cash available C: 80%, minimum hedge ratio β: 0.6, margin level ϵ': 0.2.

The definition of β and ϵ can be found in Section 2.1, and the cash available C is defined as the percentage of minimum cashflow, where the minimum cash flow is defined as $\min_{t\in[0,T]} W_t$. Figure 5 shows the relation between CVaR loss and transaction cost. In order to test the feasibility of the LSTM-RNN model under other market frictions, we will change cash available, minimum hedge ratio, margin level and then compare the CVaR loss under different market settings in table 2.

Figure 27 demonstrates that when margin level rises, the CVaR loss increases. This observation corresponds to reality because when the margin level increases, trader needs to put

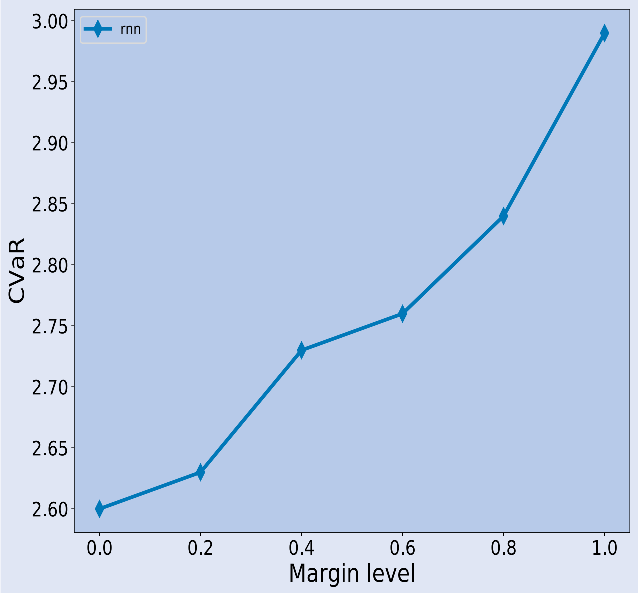

Figure 27. CVaR loss increases when the margin level goes up.

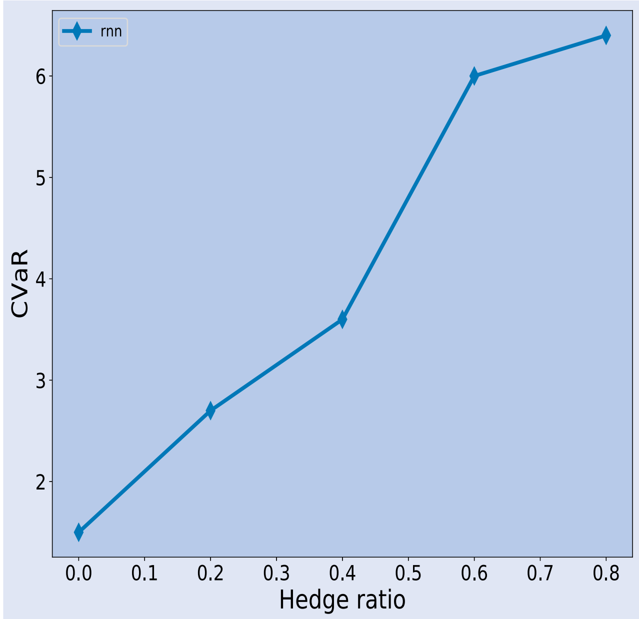

Figure 28. CVaR loss rises when hedge ratio approaches 1.

more asset as margin. As we can see in the figure, the CVaR loss does not change too much: when the margin level varies from 0 to 1, it increases only by 0.4. This can be explained by the position that the market maker holds. The market maker shorts one call, therefore it is logical for him to long the stock. As a result, the margin does not contribute much to the final PnL.

Figure 28 shows that the CVaR loss increases with the augmentation of hedge ratio. This is because the larger the hedge ratio is, the more limits on trader's stock position there will be. Thus potentially the trader loses opportunities to generate a more favorable PnL because he needs to satisfy the minimum stock holding position. Figure 29 gives the delta at 15 days for different hedge ratios. When there is no limit on stock position, the LSTM-RNN model's result is similar to that of Black–Scholes. With the increase of hedge ratio (0.2 to 0.8), the stock position of the LSTM-RNN model becomes larger.

Figure 30 shows that when the percentage of cash available increases, the CVaR loss decreases. This is coherent with our observation because when percentage of cash available decreases, the trader does not have enough cash to conduct the best hedging strategy thus there are much more potential losses. If we do not have any cash, logically the position that we hold is very limited and the CVaR loss will become very high. Figure 31 shows more details of CVaR loss when cash available varies from 20% to 120%. When cash available percentage increases, the CVaR decreases and when it is larger than 100%, meaning there are enough cash available, CVaR loss will be a constant value.

3.5. *Exotic option hedging example*

In previous sections, we only consider vanilla product hedging strategies. This is actually a very specific setting but it

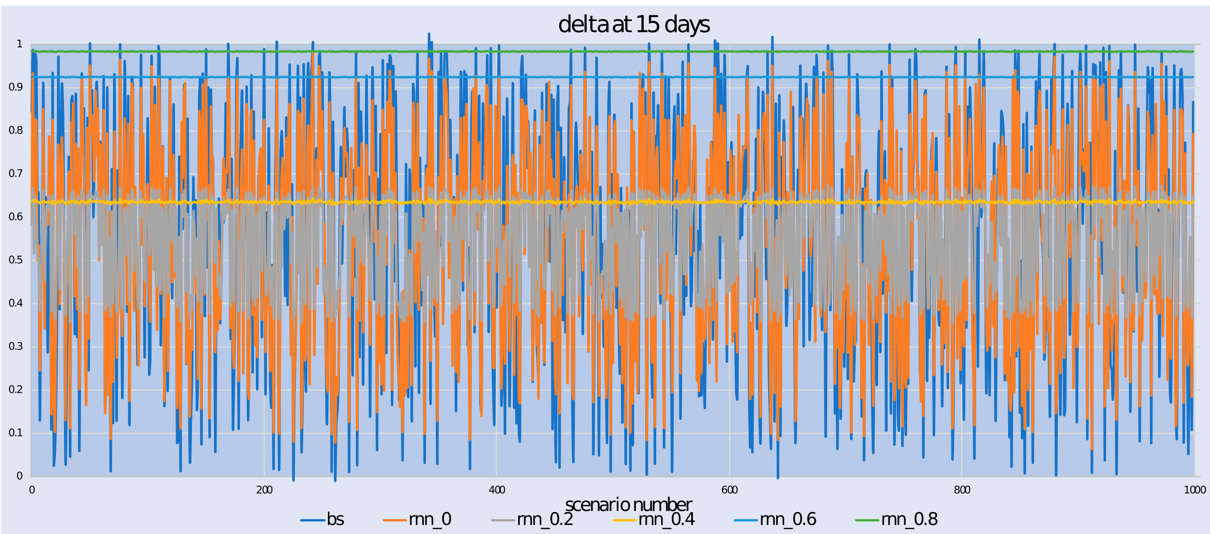

Figure 29. With hedge ratio increasing, delta at 15 days increases for different scenarios.

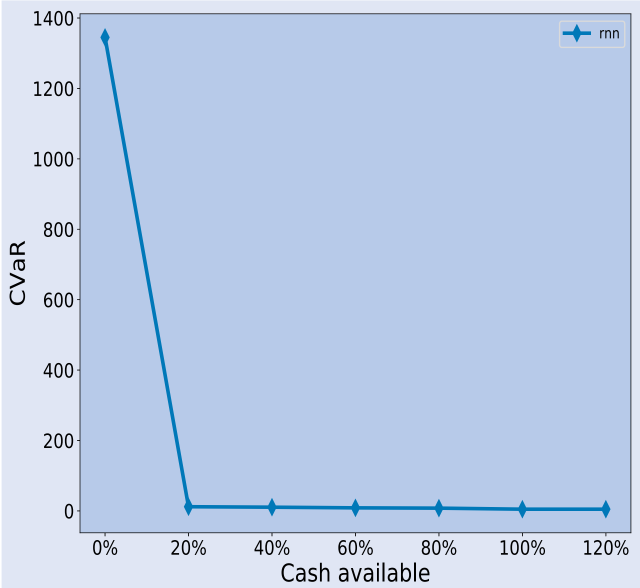

Figure 30. CVaR loss increases rapidly when cash available approaches 0.

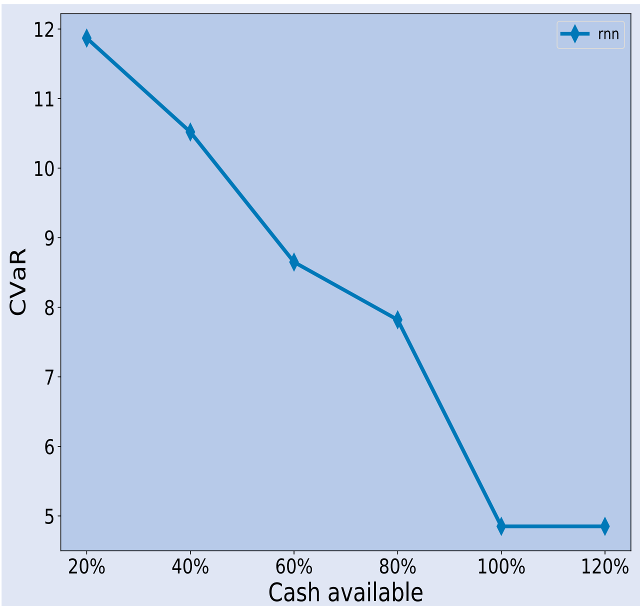

Figure 31. CVaR loss decreases when there are more cash available.

does not mean that our algorithm can not be applied to exotic options. In Section 2.1, the agent's PnL is calculated based on selling vanilla call options. By adjusting option payoff $h(S_T)$ to exotic options, we could derive the corresponding hedging strategy by LSTM-RNN method. Benchmark models in the previous chapters could not be used in this section because they are only used to hedge vanilla options.

As we stated in Section 2.1, the agent's PnL at expiry time T is equation (1). Here we consider market maker shorts 1 knock-out call option with barrier level $= B$ and strike $= K$, hence the case of $m = 1$ and

$$h(S_T) = \begin{cases} \max(S_T - K, 0), & \max_{0 \le t \le T} S_t \le B \\ 0, & \max_{0 \le t \le T} S_t > B \end{cases}. \tag{33}$$

The agent's portfolio PnL at time T (equation (3)) can be rewritten as:

$$W_T(\Delta_t) = \begin{cases} p_0 - \max(S_T - K, 0) + \sum_{i=0}^{n-1} \Delta_{t_i} \cdot (S_{t_{i+1}} - S_{t_i}) & \\ - \sum_{i=0}^{n} \epsilon \cdot S_{t_i} \cdot |\Delta_{t_i} - \Delta_{t_{i-1}}|, & \max_{0 \le t \le T} S_t \le B \\ p_0 + \sum_{i=0}^{n-1} \Delta_{t_i} \cdot (S_{t_{i+1}} - S_{t_i}) & \\ - \sum_{i=0}^{n} \epsilon \cdot S_{t_i} \cdot |\Delta_{t_i} - \Delta_{t_{i-1}}|, , & \max_{0 \le t \le T} S_t > B \end{cases}. \tag{34}$$

Similarly, we could define the optimal hedging strategy by the loss function (equation (8)) and solve the optimal problem by the method in Section 2.3. We train the LSTM-RNN model using underlying price paths generated by the GBM model. The market setting is quite similar to that in Section 3.2.1 with amendments: barrier level $B = 110$, simulation times: 10 000 where 8000 paths are ITS data and 2000 paths are OTS data, initial option price p_0 is set to 0 temporally.

For OTS data (2000 paths), the PnL for the LSTM-RNN model is shown in figure 32. It seems that the loss is normally distributed and the PnL is from -3.2 to -0.2, the mean hedging error is -1.68. If we set the initial option price to 1.68, then the PnL figure will be like figure 33 with the loss normally distributed around 0. The CVaR in this scenario is 0.72.

This example proves that RNN-LSTM is a powerful model to price options as well as generating the optimal hedging strategy. In order to use this model to price and hedge exotic European options, one may rewrite PnL expression, set initial option price to 0 and train it by underlying price paths. Once the PnL is generated, the initial price can be set to the mean hedging error and the optimal hedging strategy is the corresponding stock position holding series. Besides, the hedging strategy derived from our model performs well for spectral risk measure.

4. Conclusion

In this paper, we present a new algorithm to hedge European options under market frictions. First, we establish an optimization problem and add transaction costs, liquidity constraints, trading limits and cost of funds as market frictions. Although previous studies claim that they take into account these market frictions in their modeling, few theories or empirical studies are added. Our paper gives explicit expressions for modeling and conducts multiple empirical studies. Secondly, we find the optimal hedging strategy by minimizing the loss function using an LSTM neural network and the Adam algorithm to optimize its parameters. Thirdly, we introduce three benchmark models which extend the Black–Scholes pricing formula to incomplete markets with transaction costs. Fourthly, we conduct numerical experiments incorporating transaction costs, not only for simulated market data but also for real market data, to verify the feasibility of our algorithm and compare CVaR losses for the LSTM-RNN

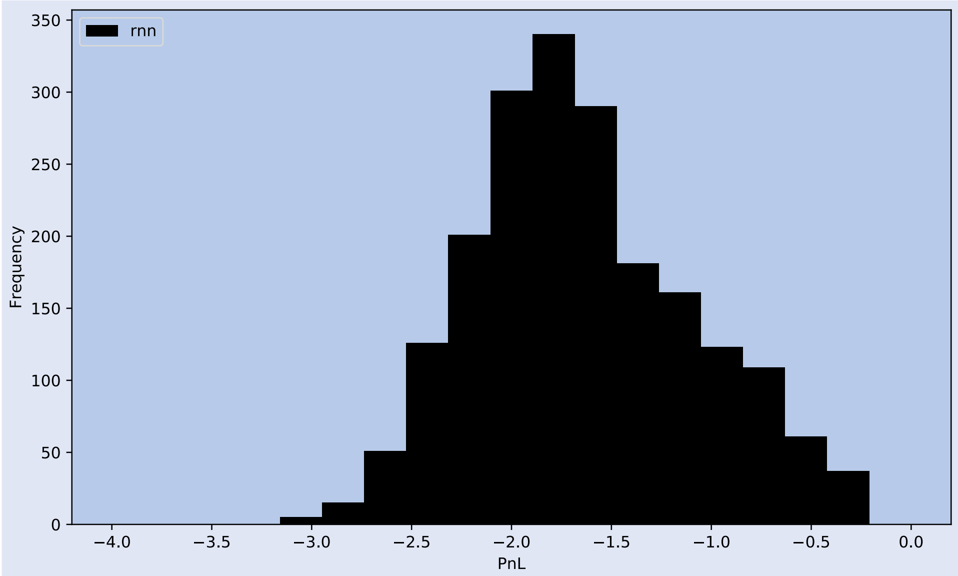

Figure 32. Barrier option hedging PnL distribution for the LSTM-RNN model (before adjusting).

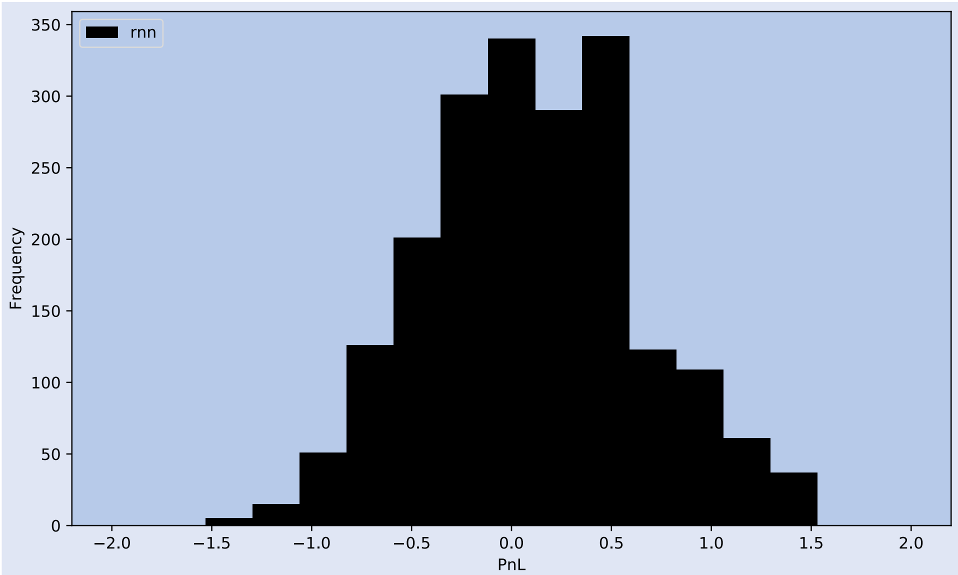

Figure 33. Barrier option hedging PnL distribution for the LSTM-RNN model (after adjusting).

model and benchmark models. Finally, we conduct an empirical analysis for market frictions other than transaction costs and show some preliminary results on hedging exotic options.

The results show that our LSTM-RNN neural network algorithm performs better than other benchmark models when calculating CVaR loss for both simulated data and real option data in nearly all option markets over the world with transaction costs. We show that the LSTM-RNN model can give optimal results under market friction factors other than transaction cost. In more detail, a higher volatility induces higher CVaR loss in a GBM market setting. When moneyness increases, different markets show different properties. A higher risk level induces higher CVaR loss in most scenarios, except for the LSTM-RNN model applied to Nikkei and S&P 500 options. A higher transaction cost proportion induces higher CVaR loss in all markets. In the GBM market setting, the LSTM-RNN model outperforms benchmark models at low or medium volatility (< 0.8), OTM moneyness and a certain risk level ($< 80\%$). For the 50ETF option market, the LSTM-RNN model outperforms benchmark models in ATM and for certain risk levels ($< 15\%$). For the HSI option market, the LSTM-RNN model outperforms benchmark models when transaction costs are smaller than 1.5%. For the Nikkei and S&P 500 option markets, the LSTM-RNN model always outperforms benchmark models. For the FTSE option market, the LSTM-RNN model outperforms benchmark models when moneyness is not deeply ITM.

Our algorithm can be applied in option market making and it provides indeed a solid method for searching for the optimal hedging strategy under market frictions. On the one hand, it is more accurate and more feasible than traditional models with transaction costs. On the other hand, the LSTM-RNN model is the first one to investigate numerical results with complex market frictions. However, compared to traditional hedging strategies, the LSTM-RNN model has two main disadvantages: the Whalley–Wilmott model is more robust than the

LSTM-RNN model, and the LSTM-RNN model is not totally explained due to the so-called blackbox of the neural network. Further studies can be done to extend our results to American option hedging strategies.

Acknowledgments

The authors are grateful for support from National Natural Science Foundation of China (grant numbers 71801008, 71850007), and Beihang University (grant number KG16003401).

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

The authors are grateful for support from National Natural Science Foundation of China [grant number 71801008], [grant number 71850007], and Beihang University [grant number KG16003401].

ORCID

Junhuan Zhang http://orcid.org/0000-0003-3969-0240

References

Andreou, P.C., Charalambous, C. and Spiros, H.M., Pricing and trading European options by combining artificial neural networks and parametric models with implied parameters. *Eur. J. Oper. Res.*, 2008, **185**(3), 1415–1433.

Boyd, N.E., Market making and risk management in options markets. *Rev. Derivatives Res.*, 2015, **18**(1), 1–27.

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

Buehler, H., Gonon, L., Teichmann, J. and Wood, B., Deep hedging. *Quant. Finance*, 2019, **19**(8), 1271–1291.

Buehler, H., Gonon, L., Wood, B., Teichmann, J., Mohan, B. and Kochems, J., Deep hedging: Hedging derivatives under generic market frictions using reinforcement learning. Available at SSRN, 2019.

Davis, M., Option pricing in incomplete markets. In *Mathematics of Derivative Securities*, edited by M. A. H. Dempster and S. R. Pliska, pp. 216–226, 1997 (Cambridge University Press: New York).

Follmer, H., Schied, A. and Lyons, T.J., *Stochastic Finance: An Introduction in Discrete Time*, Vol. 26, 2004 (Springer).

Halperin, I., QLBS: Q-learner in the Black-Scholes (-Merton) worlds, 2017.

Hochreiter, S. and Schmidhuber, J., Long short-term memory. *Neural. Comput.*, 1997, **9**(8), 1735–1780.

Hodges, S., Optimal replication of contingent claims under transactions costs. *Rev. Futures Markets*, 1989, **8**, 222–239.

Hutchinson, J.M., Lo, A.W. and Poggio, T., A nonparametric approach to pricing and hedging derivative securities via learning networks. *J. Finance*, 1994, **49**(3), 851–889.

İlhan, A., Jonsson, M. and Sircar, R., Optimal static-dynamic hedges for exotic options under convex risk measures. *Stoch. Process. Their. Appl.*, 2009, **119**(10), 3608–3632.

Jankova, Z., Drawbacks and limitations of Black-Scholes model for options pricing. *J. Financ. Stud. Res.*, 2018, **2018**, 1–7.

Kabanov, Y.M. and Safarian, M.M., On leland's strategy of option pricing with transaction costs. *Finan. Stochast.*, 1997, **1**(3), 239–250.

Kingma, D. and Ba, J., Adam: A method for stochastic optimization. *Comput. Sci.*, 2014, **5**, 00-00.

Kolm, P.N. and Ritter, G., Dynamic replication and hedging: A reinforcement learning approach. *J. Financ. Data Sci.*, 2019, **1**(1), 159–171.

Krauss, C., Do, X.A. and Huck, N., Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *Eur. J. Oper. Res.*, 2017, **259**(2), 689–702.

Lamberton, D., Pham, H. and Schweizer, M., Local risk-minimization under transaction costs. *Math. Oper. Res.*, 1998, **23**(3), 585–612.

Leland, H.E., Option pricing and replication with transaction costs. *J. Finance.*, 1985, **40**(5), 1283–1301.

Monoyios, M., Performance of utility-based strategies for hedging basis risk. *Quant. Finance*, 2004, **4**(3), 245–255.

Owen, M.P., Utility based optimal hedging in incomplete markets. *Ann. Appl. Probab.*, 2002, **12**(2), 691–709.

Ru, J.P. and Shin, H.J., An option hedge strategy using machine learning and dynamic delta hedging. *Korea Academia-Industrial Cooperation Soc.*, 2011, **12**(2), 712–717.

Soner, H.M., Shreve, S.E. and CvitaniC, J., There is no nontrivial hedging portfolio for option pricing with transaction costs. *Ann. Appl. Probab.*, 1995, **5**(2), 327–355.

Spiegeleer, J.D., Madan, D.B., Reyners, S. and Schoutens, W., Machine learning for quantitative finance: Fast derivative pricing, hedging and fitting. *Quant. Finance*, 2018, **18**(10), 1635–1643.

Sutton, R.S. and Barto, A.G., *Reinforcement Learning: An Introduction*, 1998 (The MIT Press: Cambridge, MA).

Toft, K., On the mean-variance tradeoff in option replication with transactions costs. *J. Financ. Quant. Anal.*, 1996, **31**(2), 233–263.

Whalley, A.E. and Wilmott, P., A hedging strategy and option valuation model with transaction costs. OCIAM Working Paper, Mathematical Institute, Oxford, 1993.