Trading Strategies: An Optimal Trading Systembased on LSTM and Dynamic Programming

Shengyuan Wang

swang3@macalester.edu

MSCS Department, Macalester College, Saint Paul, United States

Abstract—This study proposes a series of trading strategies for maximizing total return of Gold and Bitcoin assets over the past five years, while considering transaction commission. The authors preprocess the data by treating the floating prices of Gold and Bitcoin as two stocks, removing missing values, and using time series model LSTM to predict future prices. The LSTM model shows daily price changes in more detail and is selected as the optimal model. Monte Carlo Simulation and Markowitz model are used to find the effective weights of asset combinations, and Dynamic Programming strategy is applied to create an Optimal Action Model for finding the best trading dates. The overall model is found to be sensitive to transaction commission change.

Keywords—Long Short-Term Memory; Monte Carlo Simu- lation; Discrete Optimization

1. INTRODUCTION

1.1 Problem Background

A profitable volatile assets trading strategy is vital to Market traders. It is always applied to optimize capital allocation to maximize the overall performance, such as expected return. Return maximization is based on the estimates of a stock's potential return and risks. In general, investors make stock in- vestment decisions by predicting the future direction of stocks' ups and downs. In the modern financial market, successful investors are good at using high-quality information to make investment decisions, and they can make quick and effective decisions based on the information they have already had. Thus, the field of stock investment attracts the attention of financial practitioners and ordinary investors, and researchers in academics.

1.2 Restatement of the problem

In this problem, we are given two data sets: the Bitcoin and the daily Gold prices from 9/11/2016 to 9/10/2021 and are asked to develop a model using only the price data up to that day to decide if the trader should buy, hold, or sell their possessions in each day.

The initial possession we will start with on 9/11/2016 is $1, 000, and we are trying to maximize the total return in our portfolio, which consists of cash, Gold, and Bitcoin in U.S dollars, on 9/10/2021. We will accomplish the following tasksaccording to the given data:

ICBBEM 2023, May 19-21, Hangzhou, People's Republic of China

DOI 10.4108/eai.19-5-2023.2334301

- Develop predicted models for the price of Gold and Bitcoins.
- Develop a trading strategy using our predicted model tomaximize total return.

2. DATA PREDICTION

2.1 Data Preprocessing

First, we preprocess both Bitcoin and Gold data sets. When looking at the Gold data, we find that there are missing values (NA value) in the price column of the Gold data. Considering that the Gold data has only two dimensions (date and price) and the original Gold data has only ten missing values for 1265 rows, we simply ignore the missing values. In other words, we only calculate 1255 Gold data.

Based on our initial data analysis, we can consider the daily floating values of Bitcoin and Gold as two stocks. As predictions in stock trading require the consideration of previous data, we choose to use time series models as our primary forecasting models. We fit, analyze the long short- term memory (LSTM) models. To better compare and evaluate the strengths and weaknesses of the models, we chose to use the Bitcoin data, which is more variable than the Gold data, as our training and testing data. We used the first 80% of this data (index numbers from 1 to 1460) as training data and the last 20% (index numbers from 1461 to 1826) as testing data to visualize the comparison between predicted and true values. Also, in order to come up with a more accurate model, we choose not to do any trading operations in the first five days but just record the data. In other words, our predicted values start from the sixth day.

3. LONG SHORT-TERM MEMORY (LSTM)

3.1 Introduction to the Model

Long Short-Term Memory (LSTM) is a special type of Recurrent Neural Network (RNN) capable of learning long- term dependencies. In RNN, because there is a recursive effect, the state of the hidden layer at the last moment is involved in the computation process at the present moment. That is to say, the selection and decision are made regarding the previous state. LSTM inherits this advantage.

3.2 Adjustment of the Model

Before feeding the data into the model as training data, we need to normalize the original data. There are two reasons for doing the normalization. First, normalization can improve the speed of the gradient descent method to solve the optimal solution. Since LSTM is developed based on RNN, the essence of LSTM is to minimize the loss by gradient descent method to obtain the optimal solution. Applying normalization to the data in the gradient descent method can help the model reach the convergence state faster. Second, normalization has the potential to improve accuracy. According to the characteristics of the original data (the values are relatively concentrated), we do not consider standard deviation normalization and nonlinearnormalization. Instead, we choose linear normalization.

Before training a model for machine learning, we need to choose the right hyperparameters. In LSTM models, a few essential hyperparameters are shown below:

- Epoch: this is the total number of model forward or backward propagation iterations.

- Number of hidden layers: the number of hidden layers of the neural network. Although our input data is of low dimensionality (Value is the only dimension), we still choose the number to be 10 to get better predictions.

- Batch Size: this is the number of training samples during one forward or backward propagation before the weights are updated. The batch size must be the common factor of the training and test sets. Since the length of our training and test sets is 1455, we choose a factor of 97.

- Time step: the lag length between the training and test sets. In this case, we choose 5, which means that the overall data is considered with a lag of 5 days.

Figure 1 is when we take Epoch equal to 15 and when Epoch is equal to 20. We can see that the training loss and validation loss are almost equal. When Epoch exceeds 20, the model will be overfitting, so we finally determine Epoch to be20.

In the time series, cross-validation is not easy to do. We cannot choose random samples and assign them to either the test set or the train set. The reason is that we may choose a value from the future to test a value from the past. That situation makes no sense. In simple words, we want to avoid future-looking when we train our model.

3.3 Model Outcome

The visualization of the final LSTM model is shown on the left graph of Figure 2. In the plot, "raw" represents the original data, "train" means the fitted values of original training data, and "test" indicates the prediction values of the model. For the LSTM model, we can hardly see any original data, which means the overall performance of the model is great. The right graph of Figure 2 shows the partial enlargement. Although there are still some subtle differences between the original and predicted data from day to day, the overall trend is successfully simulated.

4. MODEL ASSESSMENT

4.1 Evaluation with MSE

To evaluate the model more logically and accurately, we use Mean Square Error (MSE) to evaluate the model, an essential metric of the model accuracy because it calculates the mean of the sum of squared difference between all predicted values and true values. We then apply LSTM to the Gold data, fit the model and visualize the predictions (Figure 3). The MSE for this model is quite low, which indicates that the accuracy of the predicted Gold prices is high.

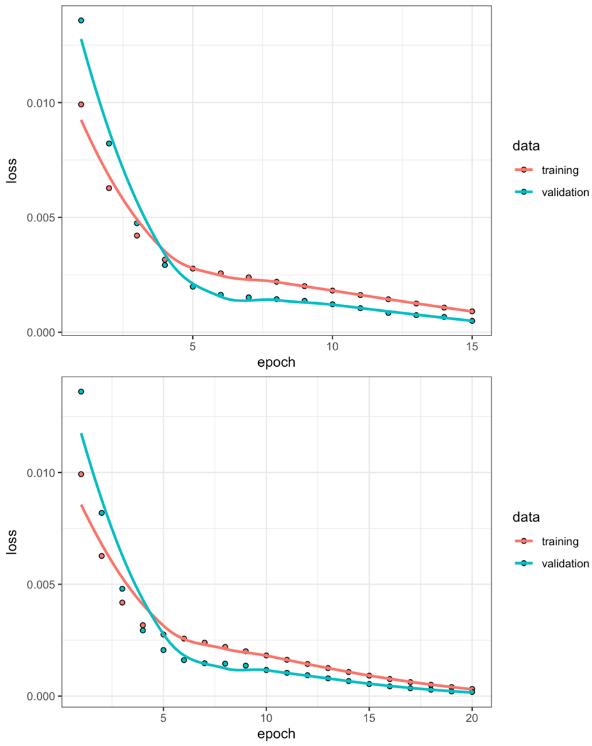

Fig. 1. Learning Curves of the LSTM Model when Epoch = 15 and Epoch =20

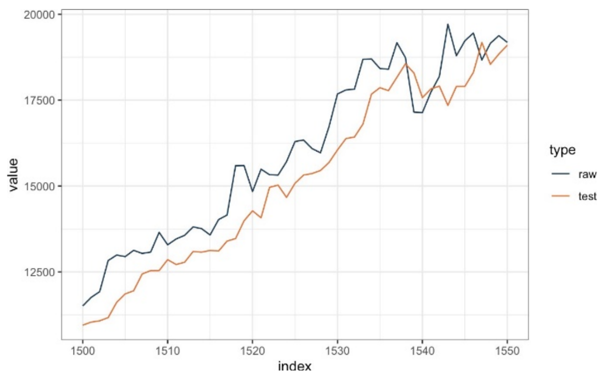

Fig. 2. Plot of Predictions using the LSTM model

5. OPTIMAL PORTFOLIO MODEL

Financial markets are fraught with uncertainty. Since we have to decide the time to trade stocks based on the results of our forecasting models, thus we should consider increasing return rates while reducing the risk of our investments. The core task in our investment optimization is to find out how an investor can allocate the assets to achieve maximizing (cumulative) returns for a given risk. In this section and next section, we will introduce our optimal portfolio model for predicting optimal weights and optimal action model for predicting time to trade.

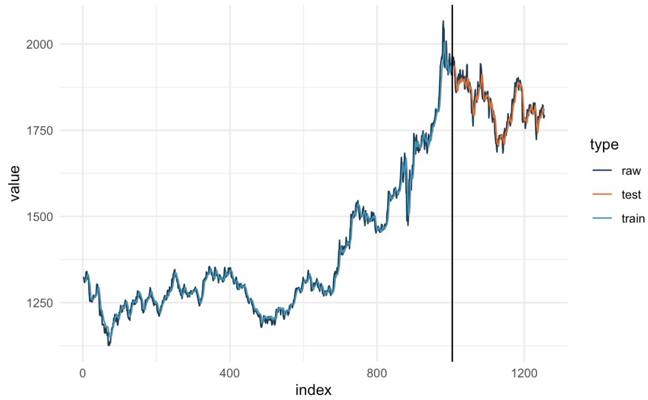

Fig. 3. Plot of Predictions of the Gold Data using the LSTM model

We first get the predicted price of each stock to get the daily return in these five years. When deciding how to allocate the funds and trade stocks, we need to set the appropriate weights for each stock trading. In other words, when trading stocks, we should allocate money by using 50% for Gold and 50% for Bitcoins or using 30% for Gold and 70% for Bitcoins. Regarding this problem, we have the following three weightingschemes.

5.1 Exploring the optimal portfolio of stocks

Considering how to balance the returns and risks when choosing the investment combination, we can introduce the Markowitz model to analyze the data and determine the best weights for predicted data.

1) Monte Carlo Simulation: Markowitz model [5] works well here because the background of the problem perfectly fits the prerequisite of the model. The investor considers each investment choice based on the probability distribution of the returns of the securities over a given holding time. The investor estimates the risk of a portfolio of securities based on the expected rate of return. The investor's decision is simply a sentence about the risk and return of the security. At a certain level of risk, the investor expects to benefit the most. Moreover, we decide to introduce Monte Carlo Simulation to create random weights to compare the outcome from different weights. [4]

When we use Monte Carlo Simulation to analyze, we randomly create a set of weights to calculate the returns and standard error of the return and repeat this progress many times(10000 times in our model). [3] Through this method, we take in each return and standard error into the model as a point to construct the scatter plot, Figure 4.

The nature of investment is to choose the balance between risks and returns, and Figure 4 depicts the two elements. Each point in the graph shows a portfolio combination, the x-axis shows the standard deviation of risks, and the y-axis is return rates. [2] Markowitz investing combination rule considers the wise investor is always maximizing the returns given fixed risk, or minimizing the risk given fixed return. It is shown in the graph as red edge, whose points on edge are the most effective investment combinations. Now we find out a series of most-effective investment weight combinations. However, we need to choose a strategy to find a final weight for Gold and Bitcoins. Here we introduce and compare two strategies: Investment risk minimization portfolio (minimize the risks) and Optimal portfolio (maximize the returns with uncertain risks).

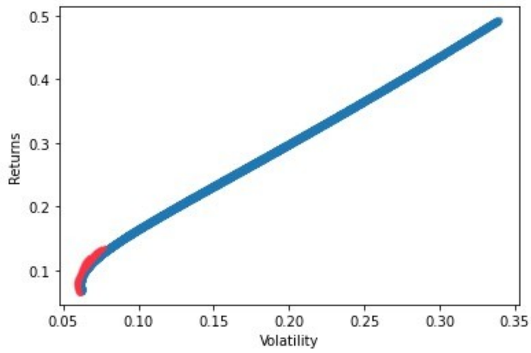

Fig. 4. Simulation Result

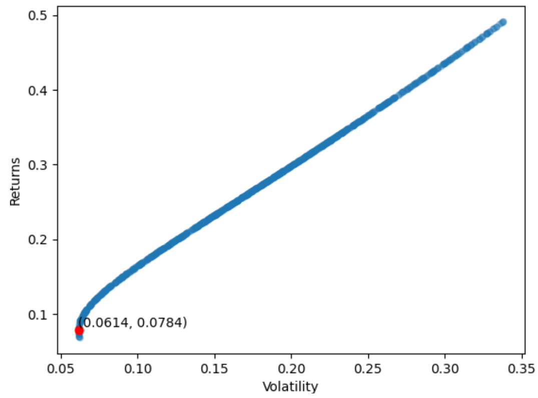

Fig. 5. GMV Portfolio

2) *Investment risk Minimization Portfolio:* One strategy is to find the highest return in the lowest risk situation, which is called Global minimum volatility(GMV) portfolio [1]. We successfully find this combination and draw it in the Figure 5.

3) *Optimal portfolio of Investments:* Since we use the optimal portfolio of investments here, we have to admit that certain risks will show up, while An wise investor can always burden certain risks to strive for a higher return. Thus, we will introduce Sharpe Ratio[1] here to help us balance return and risks for each investment combination.

[1]The ratio is the average return earned in excess of the risk-free rate perunit of volatility or total risk.

$$R_s = \frac{R_p - R_f}{\sigma_r}$$

where R_s is the Sharpe Ratio, R_p is expected rate of return, R_f is interest Rate with no risk, σ_r is the standard deviation of excess returns. The numerator calculates the spread, the excess return of an investment compared to a benchmark representing the entire investment portfolio. The denominator standard deviation represents the return volatility and responds to the risk, as higher volatility predicts higher risk. We can simply divide the mean of the excess return by the standard deviation, which is the Sharpe ratio measuring return and risks, and multiply it by 252 (there are 252 trading days in a year) to get the annualized Sharpe Ratio.

Then, we add Sharpe Ratio as the third variable into the return-risk scatter plot, Figure 6, and we use color to show Sharpe Ratio here. We find that the upper edge has higher Sharpe Ratio here. Thus, we should figure out the combination with greatest Sharpe Ratio in the scatter plot and find the weight of that combination.

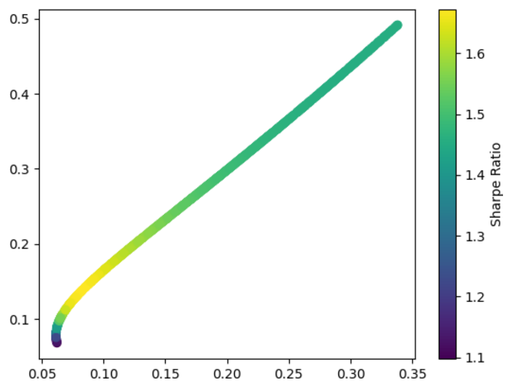

Fig. 6. Sharpe Ratio Plot

5.2 Model Outcome

In this section, we introduce and compare three weighting schemes: portfolio with given weights, portfolio with equal weights, and optimal portfolio weights. We utilize Monte Carlo Simulation, Markowitz model, and Sharpe Ratio to determine the best weights for predicted data. Through these steps, we get the optimal weights for Gold and bits is 0.188 : 0.812 from Table I. That is to say; we decide to allocate $188 for Gold investment and $812 for Bitcoins investment.

TABLE I OPTIMAL WEIGHT TABLE

	gold	bits
weight	0.188	0.812

6. OPTIMAL STRATEGY MODEL

6.1 Introduction to Dynamic Programming Model

From the optimal portfolio model result, we decide to use $188 to make investments in Gold and $812 to make investments in Bitcoins. In this way, we can avoid the dif- ference in trading days since only Bitcoins can be traded at weekends. When we decide to separately invest the stocks, in order to get greater return, if we find a model that assists us maximize either investment, the final return rate will definitelythe highest return rate.

We decide to choose the dynamic programming model to find out the best time to trade since this problem meets the optimality principle[2] with the overlap of subproblems[3] and no posteriority[4]. In this model, in order to maximize the return rate for each day, we need to compare and choose the the higher return rate for two choices: buy the assets or sell the assets. From this guideline, we can gradually find the optimal operation from start day to the last day.

In this model, M is as the cash, α is as transfer fee, S is as the stock. We first set the start of cash (M_0) in hand as 1, and the start of stock as the maximum stock (S_0) can be bought using M_0.

$$M_0 = 1 \quad S_0 = \frac{1 \cdot (1-\alpha)}{P_0}$$

The state transfer function will be as follow:

$$M_t = \max(M_{t-1}, S_{t-1} \cdot P_t(1-\alpha))$$

$$S_t = \max(S_{t-1}, \frac{M_{t-1} \cdot (1-\alpha)}{P_t})$$

6.2 Model Outcome

We predict the best time to trade from our Dynamic Pro- gramming (DP) Model and the predicted stock price from the LSTM model. After applying the time information given by our model, we make our investment change from 1,000.00 dollars to 14,140,234.70 dollars, which is a big success.

---

[2] Regardless of the past states and decisions, the remaining decisions must constitute the optimal strategy for the state formed by the previous decisions.
[3] Thesub-strategiess of an optimal strategy are always optimal
[4] Each state is a complete summary of the past history

7. SENSITIVITY ANALYSIS

7.1 Method

To test the sensitivity of transaction, we decide to test our model by changing the transaction fee rate from 50% to 150%. In other words, if the real Gold transaction fee rate is 1%, we will test how will the return rates change when we change the transaction fee rate from 0.5% to 1.5%. We will decide whether the model is sensitive to transaction cost according to the change of the final return.

7.2 Result

After making the transaction cost rate change from 50% to 150%, we create two plots, Figure 7, to show the influence of transaction fee rate on the final returns for these two stocks.

From the figure above, the left one shows Bitcoins' change of return rate caused by increasing transaction cost, and the right one shows Gold's change of return rate caused by increasing transaction cost. The return rate change for Bitcoins changes from 26000 to 12000, decreasing for about 54%, while the return rate change for Gold changes from about 1.45 to 1.15, decreasing for about 21%. Although comparing the two stocks, Bitcoins is more sensitive to the change of transaction cost, the change for both stocks shows that they are all sensitive to the transaction cost. This result is caused mainly by the dynamic programming model. Since the dynamic programming model is to optimize the final return by catching each likely chance to trade. Thus, even if the change of transaction cost from 0.01 to 0.03 does not seem like agreat number, it causes a great change to the final return.

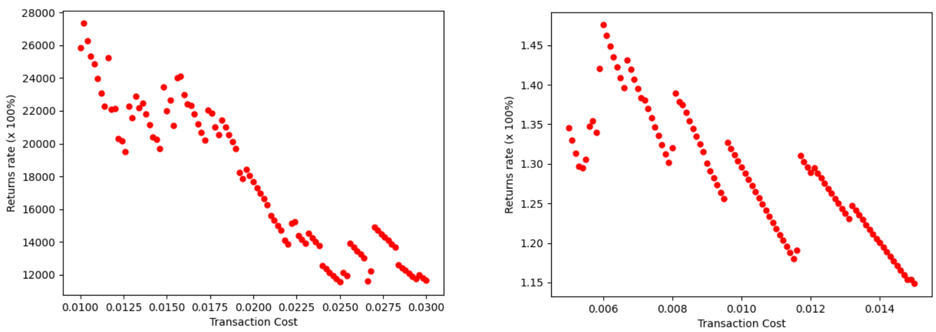

Fig. 7. Sensitive Plot (Bitcoin / Gold)

8. STRENGTHS AND WEAKNESSES

8.1 Strengths

- The Optimal Portfolio Model can both fit for two types of volatile assets allocation but also more types of volatile assets allocation.

- Dynamic Programming Model can accurately predict each possible profitable chance to invest.

8.2 Weaknesses

- Our model results may be over-fitted, because the number of real data used for validating the model is too small.

- Our model might be simple and not realistic since it is only based on the closing price of Gold and Bitcoin without the highest and lowest daily price of these two volatile assets.

- The output from Dynamic Programming Model changes corresponding to change in transaction commission.

9. CONCLUSION

To predict the trend of Bitcoin and Gold, we choose time series models LSTM. We perform parameter tuning and final result comparison for LSTM models. We will choose the LSTM Model for prediction since the results of our LSTM model are much better than those of other models. After accessing the predicted asset daily prices, we use Dynamic Programming to predict the optimal dates to trade and the type of asset to trade. After applying the prediction to theraw data, we appreciate the investment from $1000.00 to $14,140,234.70.

REFERENCES

[1] Taras Bodnar, Stepan Mazur, and Yarema Okhrin. Bayesian estimation of the global minimum variance portfolio. European Journal of OperationalResearch, 256(1):292–307, 2017.

[2] Rene D. Estember and Michael John R. Maraña. Forecasting of stock prices using brownian motion monte ..., Mar 2016.

[3] Xing Jin and Allen X. Zhang. Decomposition of optimal portfolio weight in a jump-diffusion model and its applications. Review of Financial Studies, 25(9):2877–2919, 2012.

[4] YAREMA OKHRIN and WOLFGANG SCHMID. Estimation of optimal portfolio weights. International Journal of Theoretical and Applied Finance, 11(03):249–276, 2008.

[5] Samik Raychaudhuri. Introduction to monte carlo simulation. In 2008 Winter Simulation Conference, pages 91–100, 2008.