Problem Chosen	2022 MCM/ICM Summary Sheet	Team Control Number
C		2209354

Grasp Market Changes: a Data-Based Investment Strategy

Varieties of investment has been playing an indispensable role during evolution of big data era. Traders are becoming more preferable to make decisions according to analysis of data. Therefore,aiming to help traders choose better trading strategies, a kind of data-based investment strategy is worth proposing based on prediction model and programming model. Our investment strategy demonstrates extraordinary performance on real gold and bitcoin price from 2016 to 2021.

Firstly, we divide the investment task into prediction and programming. In prediction part, **LSTM**(Long Short-Term Memory) model which is limited to extract temporal feature of previous gold and Bitcoin price is introduced. To improve the accuracy of its predicting result, we propose a history-data based pretraining process. Besides, in programming part, **HDP** (Heuristic Dynamic Programming) model is established to obtain the best strategy. The HDP model performs heuristic state transitions based on predictions of price changes, and calculate expected reward of different strategy considering trader's preference factor. Experimenting on this model on given data with the initial capital of $1,000$\$, the highest return can reach about $308.7K$ with the idealest parameters, and most of the time it fluctuate between $140K$ and $250K$ stably with reasonable adjusted parameters.

Following that, we seperately compare LSTM and HDP model with other existing models, including prediction models like Grey Prediction, SVM (Support vector machine) and HMM (Hidden Markov Model), among which our pretrained LSTM model stands out when comparing their prediction accuracy. Meanwhile, when comparing against other programming models like Markowitz Mean-Variance Model, Five-day SMA model and Ant Colony Model, our HDP model is also revealed to be the most suitable model in investment mission.

In addition, **sensitivity analysis** of different parameters on our strategy is performed. For transaction costs, experiments are done with fixed parameters on different composition of Gold and Bitcoin transaction costs rate. The result shows a great impact on strategy and reward of the Bitcoin transaction cost rate change, but a much smaller one of Gold's. It is also discovered that the Gold transaction cost rate is proportional to the final value while to whom Bitcoin transaction cost rate is inversely proportional.

Finally, we draft a memorandum to the trader, introducing the general theory, functions and advantages of our strategy, as well as a simple example of the user interface of our model in one day's application. Our code are available at "`https://github.com/Randle-Github/MCM2022G2209354_Repo`".

Keywords: **LSTM,** **HDP,** **Sensitivity Analysis**

Contents

1 Introduction **3**
- 1.1 Problem Background . . . 3
- 1.2 Clarification and Restatement . . . 3
- 1.3 Our Work . . . 4

2 Preparation of the Models **4**
- 2.1 Assumptions . . . 4
- 2.2 Notations . . . 5

3 Models **5**
- 3.1 LSTM . . . 5
  - 3.1.1 Input Module . . . 6
  - 3.1.2 Pretraining Machanism . . . 6
  - 3.1.3 LSTM Module . . . 7
  - 3.1.4 Output Module . . . 8
- 3.2 HDP . . . 8
  - 3.2.1 Model Initialization . . . 8
  - 3.2.2 State Transition Equation . . . 9
  - 3.2.3 Evaluation Indicators . . . 10
  - 3.2.4 Dynamic Programming Model . . . 12
- 3.3 Adjustment and Experiment . . . 13

4 Comparison **14**
- 4.1 Comparison with Other Prediction Models . . . 14
- 4.2 Comparison with Other Programming Models . . . 16

5 Sensitivity Analysis of Transaction Cost **17**

6 Strengthes and Weaknesses **17**
- 6.1 Strengthes . . . 17
- 6.2 Weaknesses . . . 18

1 Introduction

1.1 Problem Background

With the continuous improvement and development of financial markets, gold and bitcoin market trading is also expanding day by day. Moreover, a growing number of people are hoping to achieve asset preservation and appreciation through financial markets. However, market trading risks are becoming increasingly complex and the volatility of trading assets is getting bigger and bigger, which directly affects the realization of the goal of maximizing asset returns. Therefore, it is a greater responsibility for market traders to determine a reasonable investment data model to achieve an optimal portfolio and to opportunistically buy and sell volatile assets in real time, which is the way to maximize the return on clients' assets.

Considering only the daily price flow factors to date, it is worth studying and exploring to determine portfolio data models for USD, gold and bitcoin assets, as well as to determine whether a trader should buy, hold or sell the output of his portfolio to maximize asset portfolio benefits.

1.2 Clarification and Restatement

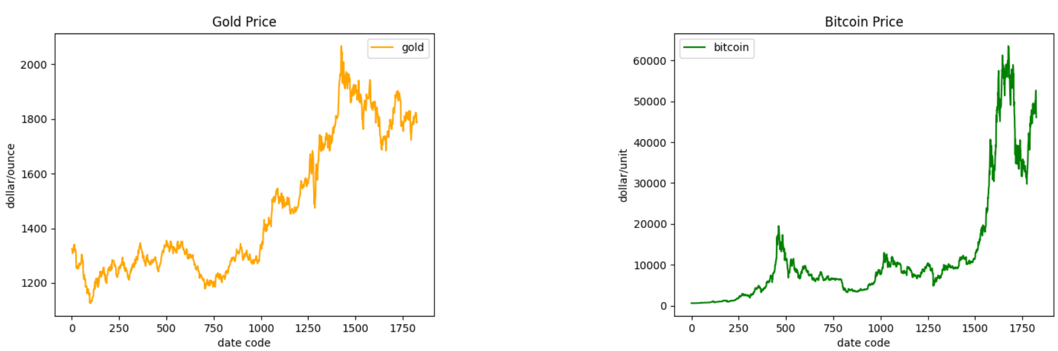

Figure 1: Gold and Bitcoin Price Volatility Curves

In this question, we are only given the exchange rate between goldbitcoin and the US dollar for each day from September 11, 2016 to September 10, 2021, respectively, as shown in **Figure 1**. Considering only the interchange of these three kinds of currency, and using $1,000\$$ dollars as start-up capital, several questions are addressed as follows.

1. Build a model that gives the best daily trading strategy based only on price fluctuations and derives the total value of the initial $1,000$ after five years.

2. Clarify why the strategy given in the above model is the optimal strategy.

3. Test the sensitivity of the model to commissions and determine how the presence of commissions will affect the trading strategy and the final outcome.

In order to solve the above problem, the following two situations require special attention.

- Bitcoin can be traded every day, but gold can only be traded on open market days.

- Each time the relevant currency is traded the trader has to pay a certain percentage of commission, corresponding to one percent for gold and two percent for bitcoin.

1.3 Our Work

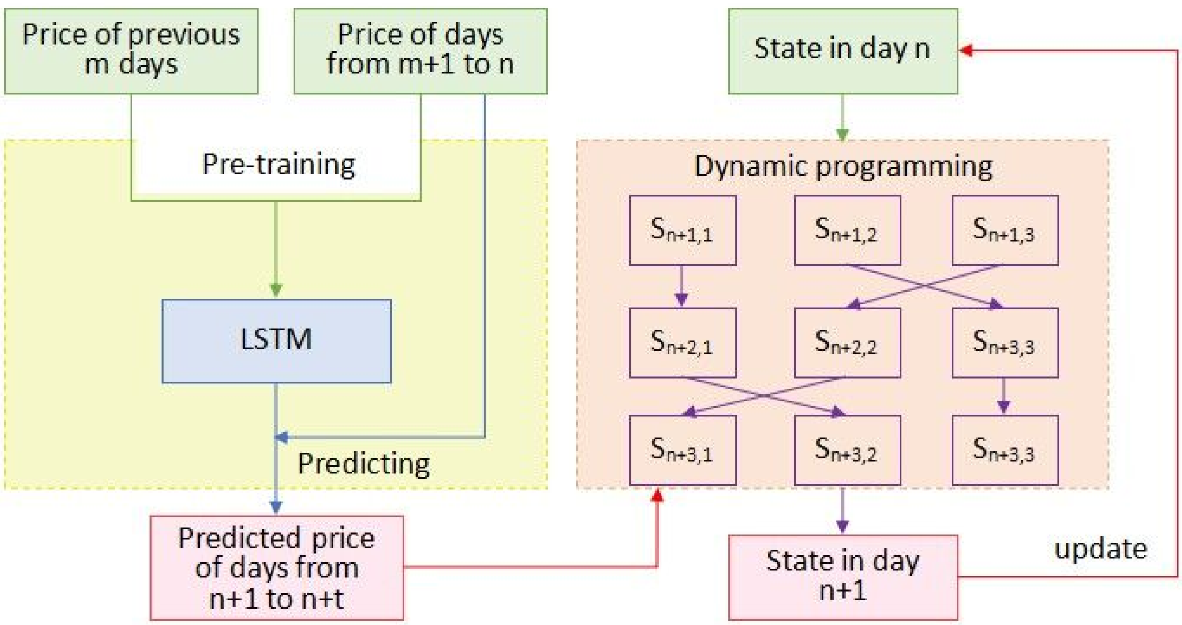

Figure 2: Flow chart of our strategy

2 Preparation of the Models

2.1 Assumptions

1. Price is related to a number of factors, but past data can predict future data to some extent.
2. Each conversion can only transfer an integer multiple of 1% of the asset to reduce the amount of calculations.
3. When the gold market is not open, the price of gold defaults to the price of the previous trading day.

2.2 Notations

Table 1: Declaration of notations and signs

Symbols	Descripton
C	Cash (dollar)
G	Gold possession (troy ounce)
B	Bitcoin possession
s	Asset allocation state
p	Price
ra	Transaction rate
O	Operation to transfer state
d	The date code

3 Models

3.1 LSTM

The prices of gold and bitcoin are constantly changing every day as the time series grows, and the trend of both prices over time plays a crucial role in our investment decisions. The ability to accurately predict price trends determines whether we can take the initiative in our investment choices. Therefore, we use long and short-term memory (LSTM) to predict the change patterns of simulated price series.

LSTM model is a variant of recurrent neural network (RNN). It is well-suited to classifying, processing and making predictions based on time series data. A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell. Such design makes the model perform better at long-term temporal prediction. Compared with other temporal model, LSTM is both accurate and easy to train. The structure of our nested two-layer LSTM model is shown in **Figure 3**. The LSTM model can be divided into three separate modules: input module, LSTM module and output module.

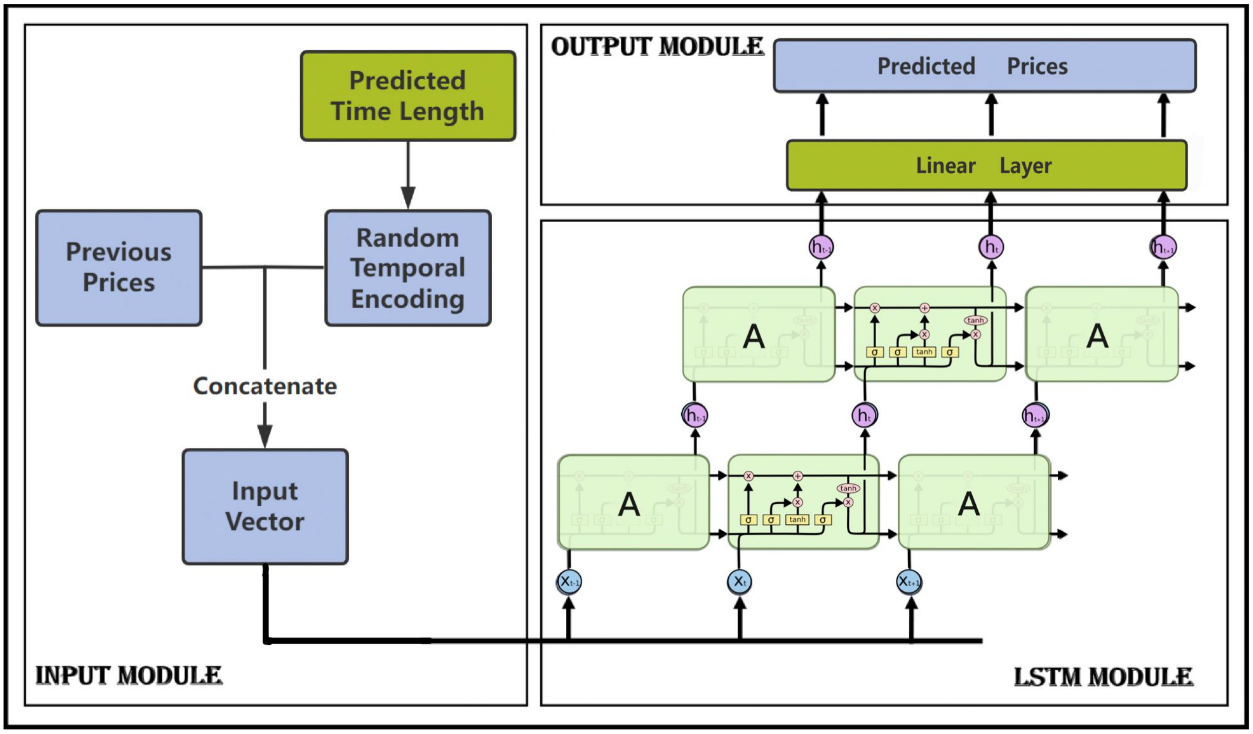

Figure 3: The structure of nested two-layer LSTM model

3.1.1 Input Module

The random temporal encoding can be first constructed based on the predicted time length obtained in the previous calculation, which is represented as a 10-dimension vector. Meanwhile, the previous prices are represented as a one-dimension vector. Therefore, the input vector for the LSTM model is gained by concatenating the above two.

3.1.2 Pretraining Machanism

According to our expirement results, we propose a kind of pretraining machanism that can notably improve the prediction accuracy. To predict t days based on previous data, we first train our model in a bi-directional pattern with all known data to learn an approximate feature of the curve. However, it normally ignore the importance of the nearby data and have a prediction of an average flat line. We then train our model in a single-directional with only the nearby data and overfit it. The final prediction will be comprehensive and pay enough attention on recent data.

Algorithm 1 Predict t days on day n

Input: n, t model of LSTM
Output: prediction of t days
pretraining the model
for $i = 0$ to $n-1$ **do**
train model for small number of times with price data in $(i, n-1)$
end for
train model for large number of times with price data in $(n-kt, n-1)$
predict t days with model with price data in $(n-kt, n-1)$ =0

3.1.3 LSTM Module

The workflow of one-layer LSTM unit is shown below:

$$\begin{cases} i = \sigma(W_{ii}x + b_{ii} + W_{hi}h + b_{hi}) \\ f = \sigma(W_{if}x + bif + W_{hf}h + b_{hf}) \\ g = tanh(W_{ig}x + b_{ig} + W_{ho}h + b_{ho}) \\ o = \sigma(W_{io}x + b_{io} + W_{ho}h + b_{ho}) \\ c' = f * c + i * g \\ h' = o * tanh(c') \end{cases} \tag{1}$$

The notations for equations above is shown in Table 2.

Table 2: Notations for one-layer LSTM

Symbols	Definition
x$\in \mathbb{R}^{11}$	The concatenated input vector for the LSTM
h$\in \mathbb{R}^{10}$	Hidden state, containing encoded information for the sequence flow
x$\in \mathbb{R}^{10}$	Cell state, tracking dependencies between the elements in the input sequence
x$\in \mathbb{R}^{10}$	Input gate, controlling the extent to which a new value flows into the cell
x$\in \mathbb{R}^{10}$	Forget gate, controlling the extent to which a value remains in the cell
x$\in \mathbb{R}^{10}$	Gathered input value from input and current hidden state
x$\in \mathbb{R}^{10}$	Output gate, controlling the extent to which the cell is used to compute outputs
W	The weight matrix for transitions
b	The bias for transitions
σ	The sigmoid function
tanh	The hyperbolic tangent function

As for our nested two-layer LSTM, a LSTM unit is overlaid with another LSTM unit. We feed the hidden state of the bottom LSTM into the upper LSTM again and use the hidden state from the upper state as the input for the output module.

3.1.4 Output Module

In the output module, we simply feed the hidden state of the upper LSTM into a linear layer to make predictions on the feature vector, which has the same structure as v_{input} in the input module.

$$v_{pred} = W_{vh} + b \tag{2}$$

Training on data of prices in figured recent days, the model is then able to predict the prices in next few days as shown in **Figure 4**.

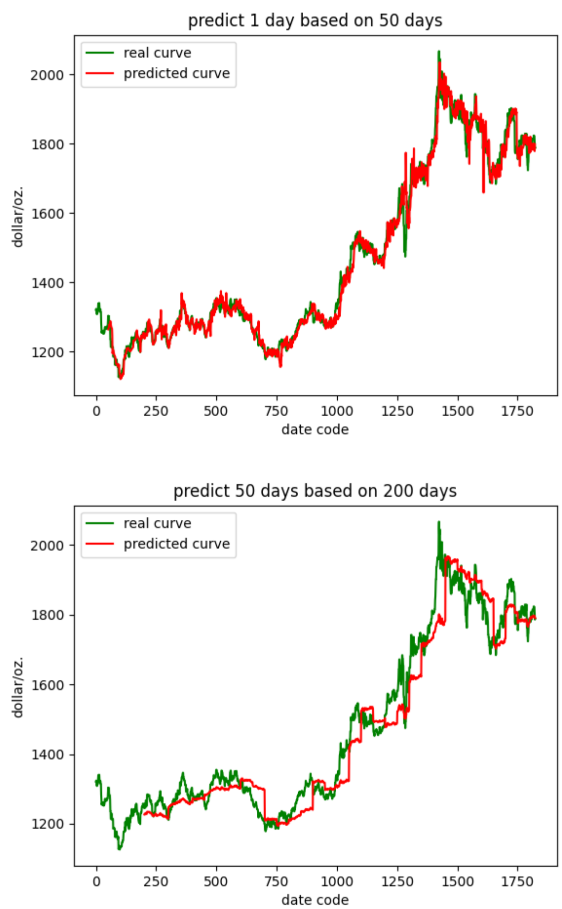

Figure 4: Graph of forecasts with different forecast days

3.2 HDP

3.2.1 Model Initialization

Obtaining the predicted price of the next t days, we propose a kind of HDP (heuristic dynamic programming) to figure out the trading strategy.

On day d, we define the trader's daily assets state vector as $s_d = (C_d, G_d, B_d)^T$, where C, G, B represent the user's cash, gold, and bitcoin holdings on that day respectively. The daily market price is $p_d = (1, G_d, B_d)^T$, where the unit value of cash is constant 1 at $p[1]$, while $p[2]$ and $p[3]$ separately represent the unit value of gold and bitcoin. Then the total value of the trader's assets on a single day for a figured state is $F_d(s) = s^T p_d$. We also define the function o as the state transition equation that transfers the state s_{d_1} to s_{d_2}.

To reduce computational complexity and investment risk we set the minimum unit for operating each asset to be 10% of that asset. To avoid wasting transaction costs, we only consider 3 kinds of operation that converts two of cash, gold and Bitcoin into the third asset. In addition, traders can neither sell nor buy gold when gold market is not open.

3.2.2 State Transition Equation

In an operation from d_1 to d_2, we aim to figure out the total value change when we make such operation in d_1, and maintains the state till d_2. In an operation, we input s_{d_1}, ra and p_{d_1} into the equation, where s_{d_1} represents initial state, ra the transfer ratio and $p1$ the price vector on the this day. To be noted, $ra[1]$, $ra[2]$ are the transfer ratio of the two converted assets in order respectively.

1. **Convert to cash from s_{d_1}:**

$$s_{d_2} = \begin{pmatrix} s_{d_1}[1] + 0.99ra[1] \cdot s_{d_1}[2] \cdot p_{d_1}[2] + 0.98ra[2] \cdot s_{d_1}[3] \cdot p_{d_1}[3] \\ (1 - ra[1]) \cdot s_{d_1}[2] \\ (1 - ra[2]) \cdot s_{d_1}[3] \end{pmatrix} \tag{3}$$

The assets value of the trader at this point:

$$F_{d_2}(s_{d_2}) = (0.99 s_{d_1}[2] p_{d_1}[2] - s_{d_1}[2] p_{d_2}[2]) ra[1] + (0.98 s_{d_1}[3] p_{d_1}[3] - s_{d_1}[3] p_{d_2}[3]) ra[2] + C$$

- When $0.99 p_{d_1}[2] - p_{d_2}[2] > 0$: F_{d_2} is positively correlated with $ra[1]$.
- When $0.98 p_{d_1}[3] - p_{d_2}[3] > 0$: F_{d_2} is positively correlated with $ra[2]$.

2. **Convert to gold from s_{d_1}:**

$$s_{d_2} = \begin{pmatrix} (1 - ra[1]) s_{d_1}[1] \\ s_{d_1}[2] + 0.99ra[1] \frac{s_{d_1}[1]}{p_{d_1}[2]} + 0.99 \times 0.98 ra[2] s_{d_1}[3] \frac{p_{d_1}[3]}{p_{d_1}[2]} \\ (1 - ra[2]) s_{d_1}[3] \end{pmatrix} \tag{4}$$

- When $0.99 \frac{p_{d_2}[2]}{p_{d_1}[2]} - 1 > 0$: F_{d_2} is positively correlated with $ra[1]$.
- When $0.99 \times 0.98 \frac{p_{d_1}[3]}{p_{d_1}[2]} p_{d_2}[2] - p_{d_2}[3] > 0$: $F_{d_2}$is positively correlated with $ra[2]$.

3. **Convert to bitcoin from s_{d_1}:**

$$s_{d_2} = \begin{pmatrix} (1 - ra[1]) s_{d_1}[1] \\ (1 - ra[2]) s_{d_1}[2] \\ s_{d_1}[3] + 0.98ra[1] \frac{s_{d_1}[1]}{p_{d_1}[3]} + 0.98 \times 0.99 ra[2] s_{d_1}[2] \frac{p_{d_1}[2]}{p_{d_1}[3]} \end{pmatrix} \tag{5}$$

- When $0.98\frac{p_{d_2}[3]}{p1[3]} - 1 > 0$: F_{d_2} is positively correlated with $ra[1]$.
- When$0.99 \times 0.98\frac{p_{d_1}[2]}{p_{d_1}[3]} p_{d_2}[3] - p_{d_2}[2] > 0$: F_{d_2} is positively correlated with $ra[2]$.

Considering the precision of prediction, it is not accurate but heuristic for us to approach the theoretical optimal solution.

3.2.3 Evaluation Indicators

1. **Profit Rate**

   As our requirement is to maximize the final return, we set the profit rate that limits too many short-term trading according to trader's ambition. If the current is the n^{th} day, the price is p_{d_n}, and the state is s_i. In prediction, the price is p_{d_1}, and the state is converted to s_i.the expected profit rate r_{d_i} is:

$$r_i = \frac{F_i(s_i)}{F_n(s_n)} = s_i^T p_{d_i} / s_n^T p_{d_n} \tag{6}$$

2. **Accuracy coefficient**

   As our experiment demonstrates that the precision of prediction greatly affects the final reward that even varies in different magnitude, we take the accuracy coefficient into consideration. Cross-validation is a useful procedure to help machine learning models obtain optimal hyper-parameters. In our model, when predicting t days after n previous days, we use cross-validation to estimate the accuracy of our prediction model.

   The traditional k-folder cross-validation is not applicable in the temporal model. As shown in **Figure 5**, we propose an extended-window cross-validation which is considered to be the best choice. From day d_0, we tested our model on validation set trained with the training set. Then, each time we slides the validation set window to the end of known data. Similarly, the accuracy coefficient is calculated as $A_{d_n} = (a_{d_{n+1}}, a_{d_{n+2}}, \cdots, a_{dn+t})$. The accuracy coefficient for prediction set a could be estimated as the mean of validation sets.

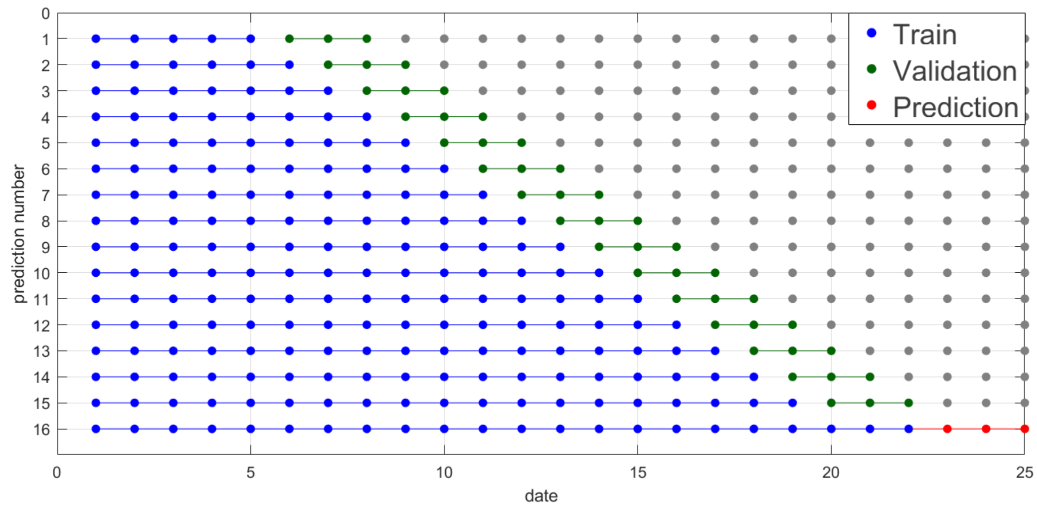

Figure 5: Cross-Validation Schematic

3. **Maximum withdraw rate**

   As a metric to estimate risk in Economics, maximum withdraw rate describes the maximum loss for an investor. The bigger the **maximum withdraw rate**, the smaller risk. It denotes the greatest decreasing rate in most pessimistic situation according to the prediction.

   In this model, the maximum withdraw rate is defined as the maximum value of assets value decline in the same condition from day d_i to day d_{n+t}. On day d_n, the price is s_n, the withdraw rate is calculated as:

$$D_i(s_i) = max(\frac{(F_i(s_i) - F_j(s_i))}{F_i(s_i)} - 1), (d_n \le i \le j \le d_{n+t}) \tag{7}$$

4. **Human subjective factors**

   Actually, the model cannot be completely accurate, and the market prices of gold and bitcoin may change greatly. Meanwhie,different traders have different risk tolerance and different levels of confidence in the prediction results of the model. Therefore, it is of vital importance to add the subjective factors of traders to simulate the results of different types of traders using the model. Define h as the the degree of braveness. The smaller $h(s_i)$ is, the more willing to take risks traders are.

5. **Expectation** The expectation is a subjective estimation, and it is not proper for us to just take the profit rate r_i as the expectation $E(s)$. The accuracy coefficient of the model may lead to great bias, and as our experiments demonstrate that expectation is strongly proportional to the accuracy. Therefore, we perform a linear transform on profit rate r_i with a_i.

   Also the model needs to consider about the trader's tendency. The more adventurous the trader is, the less the model will pay attention to the loss risk caused by the maximum withdraw rate. Here we perform a linear transform on withdraw rate D with h.

   Therefore, by integrating the above four indicators, expectation $E(s_i)$ equals is:

$$E(s_i) = r_i \cdot a_i - h_i \cdot D_i) \tag{8}$$

And our target is:

$$max(E(s_{n+t,i})) \tag{9}$$

3.2.4 Dynamic Programming Model

Since that gold and bitcoin prices vary greatly and lack regularity, our prediction may be not that accurate and there maybe a lag of change of price, and it is not suitable to use the greed model, which will make every day's operation is the optimal operation, but may miss profit opportunities or increase the risk of loss. Therefore, we use the idea of dynamic programming to create a heuristic strategy search dynamic programming model, hoping to make full use of the price prediction information of $n+1$ to $n+t$ days, and determine the operation strategy of the n*th* day with the highest evaluation score of measures as the goal.

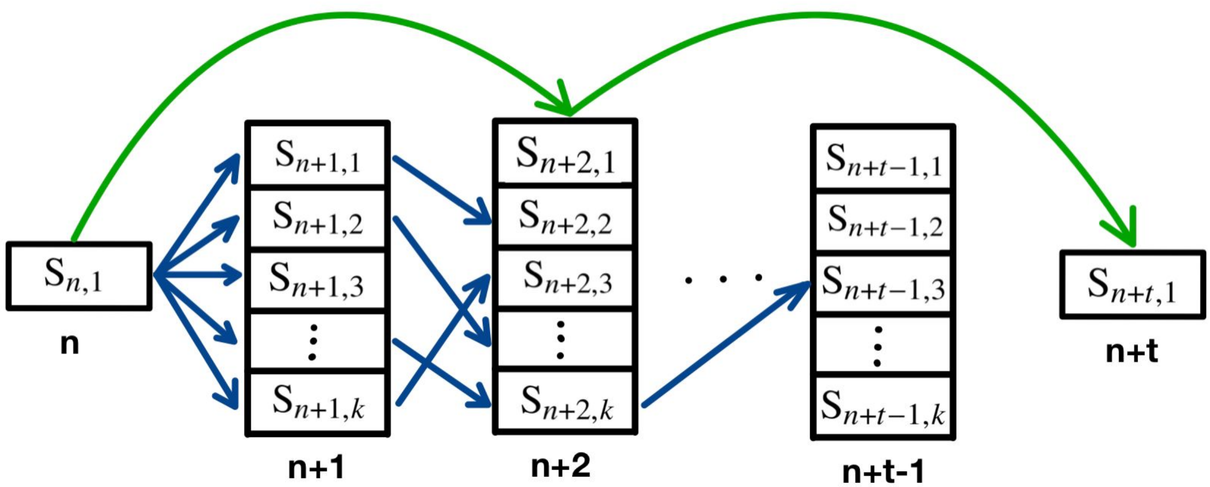

Figure 6: Heuristic Dynamic Planning Schematic

As shown in **Figure 6**, the first grid is the definite state s_n that the trader is in on the n*th* day. The 5 grids in the second column are converted from s_n. They are top k E_{n+1} biggest states after one operation from s_n and they are used as the alternative states on the n+1*th* day. There are now six initial states. Similarly, starting from each state, the top k E_{n+2} biggest states are selected after one operation, which are regarded as the alternative states on the n+2*th* day. O() is used to represent an operation, then the alternative states from the n+1 to the n+t-1 day is obtained by the following formula.

$$s_{n+i} = Top_k(E_{n+i}(O(s_{n+w}, s_{n+i}))), (1 \le i \le t-1, 1 \le w \le i-1) \tag{10}$$

Therefore:

$$s_{n+t} = Top(E_{n+t}(O(s_{s_{n+w}, s_{n+t}}))), (1 \le w \le t-1) \tag{11}$$

For example, the green line shows the change process to get the biggest $E_{n+t}(s_{n+t})$. In the prediction, we hold the assets on d_n day, and reach $s_{n+2,1}$ after one operation on n+1*th* day, finally reach s_{n+t} after one operation on day d_{n+t-1}.

3.3 Adjustment and Experiment

In our model, we need to confirm some basic parameters. According to the trader's tendency and our experiment results, The properties of these parameters and their effect on the results were investigated.

For the LSTM model, prameters are n (number of previous training days), pretrained or not. For the programming model, parameters are t (number of predicted days), h (tolerance for risk, ambition for reward), k (the states number of every predicted day). We denote our parameters as $(n, True/false, t, h, k)$. Here we give two example of t and k.

As for t, we test $t = 5, 10, 20, 50, 100$, when the parameter vector is $(50, True, t, 1.01, 20)$. We obtain results in **Table 3**. Limited with our computing power, we are not able to test t greater than 100.

Table 3: Final value when t changes

t	5	10	20	50	100
Final value	121166	177793	248156	267235	308690

As for k, we test $k = 5, 6, 7, 10, 20$, when the parameter vector is $(50, True, 20, 1.01, k)$. We obtain results in **Table 4**. The results denote that k does not influence the final results too much.

Table 4: Final value when k changes

k	5	6	7	10	20
Final value	151293	179505	180357	181265	248156

After adjusting the parameters, we came up with the best strategy whose final value is 308,690 dollars, as shown in **Figure 7**.

Figure 7: Value curve of the best strategy

Besides all parameters above, we observe that the accuracy of the prediction model makes great influence on the final value. In the extreme case, we directly used the real data as the prediction data and got a gain of 10^9. We used models that were barely trained and even ended up with huge losses.

4 Comparison

4.1 Comparison with Other Prediction Models

- **With Grey Prediction:**

  Gray predictiona method widely used in Operational Research, is a prediction method for gray systems within which only part of the information is known, and there is an uncertain relationship between the factors. Moreover, gray prediction is used to analyse the degree of dissimilarity of development trends among system factors, like correlation analysis and gray generation of the original data. Then, it is able to find the pattern of system changes and establish the corresponding differential equation model, so as to predict the future development trend of things.

  The Grey Prediction method is prominent at short-term prediction. It reaches the same level of LSTM to capture momentary change in curve and gain equal performance at 1-day prediction. However, as it lacks the ability to summarize a universal law of prices change but only focus on nearby data, its long-term prediction accuracy is much worse than pretrained LSTM.

- **With SVM:**

  The SVM (Support Vector Machine) model is a kind of traditional model of machine learning. It is a common model to predict the rise and fall of scation. tocks in real application. As shown in **Figure 8**, this classifier maps the input vector into a higher dimension one with the kernel function ϕ, and classifies the data.

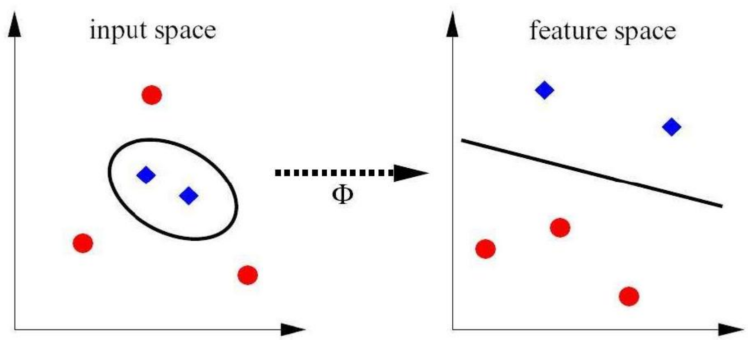

Figure 8: Visualisation of SVM classifier

  In our mission, it lacks the ability to predict accurate numbers of price. In our experiment, we take the data of previous 10 days as the input vector. However, the temporal message is seriously missing and the result is not ideal.

- **With Lagrange Interpolation Polynomial:**

  The LIP(Lagrange Interpolation Polynomial) is a method to obtain polynomial curve to fit the previous curve. If given $X = [x_0, x_1, \cdots, x_n]$ and its corresponding $Y = [y_0, y_1, \cdots, y_n]$, it is easy to get the basis function:

$$l_i(x_j) = \frac{(x_j - x_0)\cdots(x_j - x_{i-1})(x_j - x_{i+1})\cdots(x_j - x_n)}{(x_i - x_0)\cdots(x_i - x_{i-1})(x_i - x_{i+1})\cdots(x_i - x_n)}(j \neq i) \tag{12}$$

  Multiply each basis function by the corresponding y_i and add them together to get the Lagrangian interpolation polynomial:

$$L_n(x) = \sum_{i=0}^{n} y_i \frac{(x - x_0)\cdots(x - x_{i-1})(x - x_{i+1})\cdots(x - x_n)}{(x_i - x_0)\cdots(x_i - x_{i-1})(x_i - x_{i+1})\cdots(x_i - x_n)} \tag{13}$$

  Compared with LSTM, it is too easy to overfit the previous data. In long-term prediction, the results shown departure

- **Mean Squared Error(MSE) Evaluation System:**

  The mean squared error in mathematical statistics is the expected value of the squared difference between the parameter estimate and the true value of the parameter, denoted as MSE.

$$MSE = \frac{1}{n}\sum_{i=1}^{m}(y_i - y_{predi})^2 \tag{14}$$

MSE is applied to measure the average error of the above methods, and their respective values are shown in **Table 5**.

Table 5: Evaluation of each method

MSE \ Predicted/Training Days — Method	1/30	5/50	50/200
LSTM(pretrained)	423.6	**1170.3**	**7512.3**
LSTM(unpretrained)	**384.7**	1320.4	10353.2
Gray Prediction	407.4	2686.7	34714.5
LIP	966.5	16483.2	$\times 10^8$

rise/fall \ Predicted/Training Days — Method	1/30	5/50	50/200
SVM(rise/fall)	68.2%	57.7%	57.0%

4.2 Comparison with Other Programming Models

- **With Markowitz Mean-Variance Model**

  Asset allocation using the Markowitz mean-variance model assumes that the investor knows the probability distribution of each expected asset return. Therefore, to use this method, it must be assumed that there is a good fit between expected returns and historical returns and that no large changes will occur. However, this assumption certainly has significant limitations, especially for volatile assets such as gold and bitcoin, and using the historical return approach can lead to a model that is very sensitive to the interval sampled, resulting in large changes in the parameters of the resulting model.

- **With Five-Day SMA (Simple Moving Average) Model**

  The five-day SMA model, as the name implies, reflects the price trend of a currency by using the average of its trading price or index over the previous five days as an indicator. The biggest advantage of our model over the five-day average method is the introduction of a pre-training mechanism. Before predicting the price trend, some characteristics of the previous trend have been roughly grasped, which can largely improve the accuracy of our prediction, as shown in **Table 5**.

- **With GA(Genetic Algorithm) Model**

  The GA is a kind of traditional heuristic algorithm that widely used in industry. The main idea of it is to encode the states and develop a method to generate new states from existing excellent states. Compared with our HDP model, it lacks a clear optimal direction. In same computaion ability, GA could not figure out a great strategy. Our model also possesses greater ability to preserve efficient solutions, and it is more strict in mathmetic strategy.

5 Sensitivity Analysis of Transaction Cost

To determine how sensitive the strategy and return is to transaction costs, we change the transaction rate α_{gold} from 0.0 to 0.06 and $\alpha_{bitcoin}$ from 0.0 to 0.05. At the the same time, we keep other parameters as the best combination.

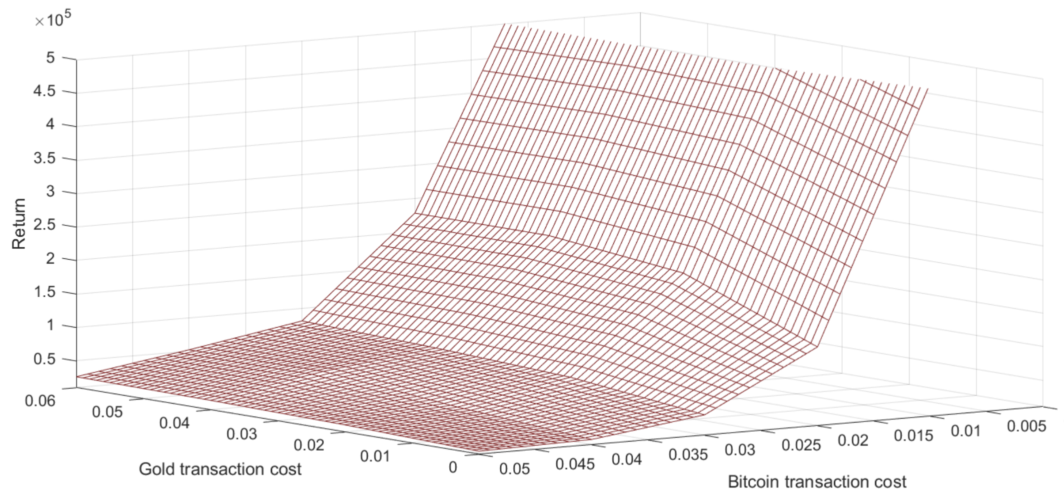

Figure 9: Sensitivity to Transaction Cost

As shown in **Figure 9**, when α_{gold} increases and $\alpha_{bitcoin}$ keeps the same, the final value increases. Specificly, when α_{gold} increases from 0 to 0.02, return increases more obviously. When $\alpha_{bitcoin}$ increases from 0.02 to 0.06, the return increases gently. In comparison with α_{gold} , the return is more sensitive to $\alpha_{bitcoin}$. when $\alpha_{bitcoin}$ increases from 0 to 0.05, the final reward decrease greatly. However, for every 0.1 decrease in $\alpha_{bitcoin}$, the reward increases even more than ten times.

From the price data, we can see sharpen increase in the price of bitcoin around day 600 and around day 1600. In most period of time, the model tends to transfer into bitcoins than gold, which could explain the reason why the return is more sensitive to $\alpha_{bitcoin}$. Furthermore, when α_{gold} is smaller, the model will buy more gold, and reduce bitcoin holdings, which result in decrease of return. Otherwise, owing to that the bitcoin price growers faster, there is many times to buy or sell bitcoins, so that when $\alpha_{bitcoin}$ increases from 0 to 0.05, return decrease greatly.

The results also demonstrates the change of strategy in model. Times of trading into gold and bitcoin remains around 5 : 1 with the original ratio. However, as we increase α_{gold} or decrease $\alpha_{bitcoin}$, the times of trading into gold and bitcoin changes into 7 : 1, which denotes that our model adapts well from different transaction rate and develop new trading strategy to obtain higher return.

6 Strengthes and Weaknesses

6.1 Strengthes

- Our model not only considers the total value of assets in the following day ,but also makes full use of multi-day prediction price to make the expectation higher after a period of time.

- Our pretrained LSTM could remain the structure and parameters for the next training data, which greatly decreases the computational costs.
- Compared with other models, our model takes better strategy and obtain the highest result of $304.9K$.

6.2 Weaknesses

- Our model is strongly depending on the accuracy of prediction. In our experiment, with prediction model in low accuracy, the final reward will be deeply decreased.
- The paramters in our model are difficult to adjust. Our model requires great computation resources and too many experiments.
- The stabability of our model could not be ensure and the test results of our model vary widely.

Memorandum

To: Traders

From: Team#2209354

Subject: Investment Strategy

Date: 2022-02-21

Considering today's growing quantitative trading market, more and more people are looking to profit from the market by building mathematical models and using computer technology to replace human subjective judgment. As response to your requirement, we are here pretty glad to have the opportunity to introduce our research and strategies to you, with the hope that it may maximize your profits.

We have carefully and comprehensively compiled and analyzed the data related to the price fluctuations of gold and bitcoin over the last five years. Among many models such as grey forecasting, support vector machine (SVM), Lagrange interpolation, etc., we finally determined the LSTM model to predict the trend of both and proved its superiority in comparison.

Subsequently, we introduced a heuristic dynamic programming algorithm to give the optimal trading scheme. Compared to empirical-based financial analysis (e.g., five-day averaging method, etc.) and common heuristics, our algorithm is able to achieve better results with the same arithmetic power.

In particular, the personal and subjective factors of different investors are taken into account in the development of this program for you. Specifically, we need to determine exactly what investment strategy to adopt based on your personality and needs. Afterwards, we will make some minor adjustments to the model parameters based on your preferences. For those investors who are somewhat aggressive, our model prefers long-term investment returns, a scenario that carries some risk, but probably more reward. In our tests, the highest return can reach about 308.7k, and most of the time the return fluctuates between 140k and 250k. For those investors who are somewhat conservative, our model will prefer to choose relatively short-term investments to ensure that each transaction does not lose too much money. The final return for this scenario may be relatively less, but more stable.

We will use the following two tables to show you our user interface. As shown in **Table 6,7** our user interface will present the various asset holdings, the expected return over time, the risk assessment and the corresponding strategies as you use our model specifically for each day.

Table 6: User status interface

	assets / info.	Cash	Gold	Gold
Day 38	possession	1328.2	52.1	327.4
	price	1.0	36.2	154.2

Table 7: User suggestion interface

	Expected reward (20days)	hazard	strategy	$\cdots$
Suggestion$_1$	3800	1.2	convert 50% cash to bitcoins	$\cdots$
Suggestion$_2$	3400	1.3	convert 70% gold to bitcoins	$\cdots$
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$

We really appreciate this opportunity to assist you in building an investment strategy. We believe that our advice can be used to make you more profitable. Please feel free to contact us for further information about this project.

Sincerely yours
Team#2209354

References

[1] Mengya Li,Juan He,Rong Zhou,Li Ning,Yan Liang. Research on prediction model of mixed gas concentration based on CNN-LSTM network[C]//Proceedings of 2021 3rd International Conference on Advanced Information Science and System (AISS 2021).,2021:344-348.DOI:10.26914/c.cnkihy.2021.052241.

[2] Greff K, Srivastava R K, Koutník J, et al. LSTM: A search space odyssey[J]. IEEE transactions on neural networks and learning systems, 2016, 28(10): 2222-2232.

[3] Gers F A, Schmidhuber J, Cummins F. Learning to forget: Continual prediction with LSTM[J]. Neural computation, 2000, 12(10): 2451-2471.

[4] Shi X, Chen Z, Wang H, et al. Convolutional LSTM network: A machine learning approach for precipitation nowcasting[J]. Advances in neural information processing systems, 2015, 28.

[5] Fu Zhongjie,Wu Qingqiang. Quantitative Trading Strategy of Market States Prediction Based on HMM[C]//Proceedings of 2017 2nd International Conference on Materials Science,Machinery and Energy Engineering(MSMEE 2017).,2017:1333-1343.

[6] Changjie Chen. Using Index model and Markowitz Model to analyze the Results of Retail Industries Before and After the Covid-19 Virus Outbroke[C]//Proceedings of 3rd International Conference on Global Economy and Business Management (GEBM 2021).[],2021:240-249.DOI:10.26914/c.cnkihy.2021.049922.

[7] Wang Yuan,Luo Chao. An intelligent quantitative trading system based on intuitionistic-GRU fuzzy neural networks[J]. Applied Soft Computing,2021(prepublish):

[8] Van-Dai Ta,CHUAN-MING Liu,Direselign Addis Tadesse. Portfolio Optimization-Based Stock Prediction Using Long-Short Term Memory Network in Quantitative Trading[J]. Applied Sciences,2020,10(2):

[9] Andrada-Félix, J., Fernández-Rodríguez, F., 2008. Improving moving average trading rules with boosting and statistical learning methods. Journal of Forecasting 27 (5), 433–449.

[10] Arévalo, R., García, J., Guijarro, F., Peris, A., 2017. A dynamic trading rule based on filtered flag pattern recognition for stock market price forecasting. Expert Systems with Applications 81, 177–192.

[11] Vogel S, Ney H, Tillmann C. HMM-based word alignment in statistical translation[C]//COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics. 1996.

[12] Chen K, Zhou Y, Dai F. A LSTM-based method for stock returns prediction: A case study of China stock market[C]//2015 IEEE international conference on big data (big data). IEEE, 2015: 2823-2824.

Appendix

The file tree structure and information of all files, at: https://github.com/Randle-Github/MCM2022G2209354_Repo

```
root/
|--datasets/ --"datasets"
|    |--Gold.npy --"Gold data after preprocessing"
|    |--Bitcoin.npy --"Bitcoin data after preprocessing"
|    |--BCHAIN-MKPRU.csv
|    |--LBMA-GOLD.csv
|    |--preprocess.py --"to preprocess original .csv data"
|    |--date_map.json --"the map from date to date encode"
|    |--fill_missing_date.m --"to fill the missing data of Gold"
|    |--position_encoding.npy --"the date encode"
|    |--utils/
|        |--visual.py --"visualise the results"
|--dataloader.py --"a tool to load data from datasets/"
|--predict.py --"the predictor of LSTM model"
|--run.py --"the runner of LSTM model"
|--lstm.py --"the LSTM model"
|--dynamic_programming/
|    |--DP.py --"the Heuristic Dynamic Programming to give a final solution"
|    |--utils.py --"the calculator of expectation for DP"
|    |--dataloader.py --"a tool to load data from prediction/"
|    |--prediction/ --"the prediction of t days based on previous n days"
|        |--0.npy
|        |--1.npy
|        |--..
|        |--1937.npy
|        |--gold_allow.npy --"the trading date of gold"
|--comparison/ --"different prediction model to beat with"
     |--LIP.py --"with Lagrange Interpolation Polynomial model"
     |--SVM.py --"with Support Vector Machine model"
     |--Grey.m --"with Grey Prediction model"
     |--MAE_judger.py --"Mean Average Error calculator"
```