Problem Chosen	2022 MCM/ICM Summary Sheet	Team Control Number
C		2202238

A Cross-Market Investment Strategy based on Prediction and Moving Average Method

Summary

In recent years, the digital currency has been a heated discussion. People use their spare money to purchase volatile assets such as gold and bitcoin to get more returns. However, many investors cannot maximize their profit or even lose money due to the improper investment strategy. To help traders optimize their strategy, our team proposed a model consisting of prediction and decision-making process.

Firstly, we designed a predictor to predict the price of the following day. According to the theory of statistics, the price change of assets can be seen as a random process. We introduce **Geometric Brownian Motion (GBM)** to describe this process. The output of GBM is the expectation and variance of predicted price. With the variance being small, we use the expectation to denote the predicted price. The parameters of GBM are obtained through a fixed length of previous data, using Logistic Regression and empirical method.

Next, we put forward **Cross-Market Model** for decision making. The model is divided into two parts. One is the asset transfer between markets, and the other is the investment activity within one market. For the first part, we evaluate and reallocate our assets every 28 days. At the beginning of the investment, we divide cash equally into two markets. Then the monthly rate of return is calculated, and we compare the profitability of each market. If one market has greater earning potential, we reallocate more cash into it or sell the asset of the other market next month. The loss of transaction cost is considered and the reallocation function is based on the overall return rate. Whats more, we give strict proof of the optimality of this model.

For the second part, we invest in each market independently with the **Moving Average Method**. In the first 60 days, we make conservative decisions and observe the price trend. Decisions are made based on the error between the actual price and the predicted price, as well as the daily rate of return. After 60 days, we make bolder decisions to gain more profit. Then the moving average method is applied. Through the difference between the 10-day average and the 60-day average, we derive the decision curve. According to **Ganvles Rules**, we can decide when to buy, hold, or sell each asset with the decision curve. Some limitations are given to avoid multiple transactions which may cause loss.

The result of our model is reliable. The predicted price follows the actual price well, with a reasonable delay. Through the calculation of our decision-making model, the final cash we have is **62513.5 $**. During the 5 years, we made **21 transactions** in total. Besides, we test the sensitivity of our model to transaction cost and analyze its advantages and limitations. Finally, we conclude the paper and write a memorandum to share our model with traders.

Keywords: Moving Average; Geometric Brownian Motion; Logistic Regression; State Transition

A Cross-Market Investment Strategy based on Prediction and Moving Average Method

February 22, 2022

Contents

1 Introduction **3**
1.1 Background . . . 3
1.2 Our Work . . . 3
2 Problem Restatement **3**
3 Notations and Assumptions **4**
3.1 Notations . . . 4
3.2 Assumptions . . . 4
4 A Price Predicting Model based on GBM **5**
4.1 Geometric Brownian Motion Model . . . 5
4.2 Parameter Estimation by Logistic Regression . . . 6
4.3 Implement of Prediction Model . . . 7
5 Decision Making Strategy **8**
5.1 Cross-Market Analysis . . . 8
5.2 Independent Market Analysis . . . 9
5.3 Optimality Proof . . . 11
5.4 Implement of Decision Making Model . . . 12
6 Sensitivity Analysis **13**
7 Strengths and Weaknesses **15**
7.1 Strengths . . . 15
7.2 Weaknesses . . . 15
8 Conclusion **15**
A Memorandum to Traders **17**

Appendices 18

Appendix A Code for Prediction Model 18

Appendix B Code for Decision Model 19

Appendix C Parameters and Data 21

1 Introduction

1.1 Background

Looking at the yellow lines on the screen nervously, you have no idea which asset you should buy or sell. When it goes upwards, you put all your money in with excitement. On the next day, however, it drops madly as if it is an unhappy boy. In the financial currency market, it is always hard to predict the price trend and make a right decision. With more and more people interested in buying volatile assets, it is necessary to make the predicting process and decision making process more scientific. Thus, our team proposed a model to help investors out.

Among all the volatile assets, bitcoin and gold enjoy the most popularity within the globe. Our team use the historical price data of gold and bitcoin to make the most accurate prediction of the future. According to the prediction result, we decide when and which asset should we purchase or sell in order to maximize the total return.

1.2 Our Work

We first designed a price predictor using Geometric Brownian Motion (GBM) Model. Each day, we load the data of a fixed length up to that day into our predictor to predict the future price trend. We use Logistic Regression to train our model and optimize the parameters. Then, we designed a decision maker considering all the related constraints. Next, we proved that our model is one of optimal solutions to the problem.

After that, we applied our model to the given data to compute the final gain, and when to buy or sell the assets. Besides, we tested the sensitivity of our model to transaction costs and analyzed the strengths and weaknesses of it. Finally, we concluded the paper and wrote a memorandum to share our model with traders.

2 Problem Restatement

Our team is required to finish tasks as follows:

1. Devise a model to decide whether the trader should buy, hold or sell the property they have each day. The decision should base only on price data before that day. We need to predict the price first and then make decisions.
2. Prove that our model provides optimal strategy.
3. Analyse the sensitivity of our model to transaction costs and find out how transaction costs affect the strategy and results.
4. Write a memorandum to traders to share our model, strategy, and result.

3 Notations and Assumptions

3.1 Notations

In this section, we list the nomenclatures that are used in the passage. Some may have different subscripts based on its users. Other symbols that are not frequently used will be introduced later.

Table 1: Symbols and Meanings

Symbol	Meaning
α	The commission proportion for each transaction
S_k	The actual price on day k
$\hat{S}_k$	The predicted price of day k.
$R_k^d / \hat{R}_k^d$	The actual/predicted daily earning yield rate of day k.
R_G^m / R_B^m	The gold/bitcoin monthly earning yield rate.
k	The transfer coefficient of the Cross-Market
L	The length of data we use for price prediction.
E_{rr}	Relative error, the difference between S_k and $\hat{S}_k$ divided by S_k
β	The difference between the short term moving average and the long term moving average
Q	The estimated profit in one month
A	Total assets of bitcoin, gold, and cash in USD.
B/G	The number of the bitcoin/troy ounce
C	The amount of cash (U.S. dollars)

3.2 Assumptions

- **The price trends of gold and bitcoin are combined with trend and fluctuation.** Because there are too many factors affecting the financial market, the price will fluctuate in a small range in the short term. In the long term, however, the financial market's low will be dominant, thus the price will show a certain trend which is the basis of our prediction.
- **We do not take major emergencies into our main model, such as war and COVID-19.** Because the emergencies occur randomly in the real life which is unpredictable, so it is reasonable to ignore them and to focus on the model itself. In addition, we develop a special decision strategy to handle with the unexpected events.
- **We assume that the value of cash remains stable, i.e. it cannot be deposited in a bank for interest.** As mentioned in requirements, there is no cost to hold an asset. We

make another assumption that the amount of cash cannot be increased by interest as well.

- **We assume that we make decision on the morning without knowing today's price information and make prediction on the evening.** To avoid further confusion and potential bugs, we promise that we use only historical price information to decide how to invest today.
- **We assume that we will spend all of our money on bitcoin or gold, or sell all bitcoin or gold in each legal transaction.** Due to the high transaction cost rates (1% for gold and 2% for bitcoin), we constrain our buying and selling strategy to avoid high losses due to unnecessary multiple transactions.
- **We do not make any transaction for a small period of time at the beginning, in order to obtain some basic data.** We set the value of this window period to 10 days based on the theory [1].
- **We assume that the gold market and the bitcoin market do not influence each other.** There is no convincing evidence [2] showing either bitcoin price influence gold price or gold price influence bitcoin price, so we do not consider the interaction between gold market and bitcoin market.
- **We assume the value of gold remains stable on weekends.** Gold market does not open on weekends, so there is no transaction during weekends. Thus, we treat the gold price on Friday as the gold price on that weekend.

4 A Price Predicting Model based on GBM

4.1 Geometric Brownian Motion Model

At first, let's give a brief introduction to Itô process, which is a stochastic process expressing Brownian motion in the form of a stochastic differential equation. Let X be an Itô process, it has the following stochastic differential equation form:

$$\mathrm{d}X(t) = \mu(t)\mathrm{d}t + \sigma(t)\mathrm{d}B_t \tag{1}$$

where $\mu(t)$ denotes the expectation of X, $\sigma^2(t)$ denotes the variance of X, and B_t is the Standard Brownian Motion. Assume that $F(t,x)$ is a continuous function in $[0,T]\times\mathbb{R}$, then it has continuous partial derivatives F_t, F_x, F_{xx}. We define $Y(t) \triangleq F(t,X(t))$. Then $Y(t)$ is also an Itô process. According to Itô Fomula, we have:

$$\begin{aligned}\mathrm{d}Y_t =& \mathrm{d}F(t,X(t)) \\ =& F_t(t,X(t))\mathrm{d}t + F_x(t,X(t))\mathrm{d}X(t) + \frac{1}{2}F_{xx}(t,X(t))\mathrm{d}X(t)\mathrm{d}X(t)\end{aligned} \tag{2}$$

We use S_t to denote the price, and apply Itô Fomula to $\ln S_t$, we have:

$$\begin{aligned}\mathrm{d}\ln S_t &= \mathrm{d}F(t,S_t) \\ &= F_t(t,S_t)\,\mathrm{d}t + F_s(t,S_t)\,\mathrm{d}S_t + \frac{1}{2}F_{ss}(t,S_t)\,\mathrm{d}S_t^2 \\ &= F_s(t,S_t)\,\mathrm{d}S_t + \frac{1}{2}F_{ss}(t,S_t)\,\mathrm{d}S_t^2 \\ &= \frac{1}{S_t}\mathrm{d}S_t - \frac{1}{2}\frac{1}{S_t^2}\mathrm{d}S_t^2\end{aligned} \tag{3}$$

Equation (3) will be used in Geometric Brownian Motion [3].

A geometric Brownian motion (GBM) is a continuous-time stochastic process in which the logarithm of the randomly varying quantity follows a Brownian motion with drift [4]. It satisfies the following stochastic differential equation:

$$\begin{cases} \mathrm{d}S_t = S_t \mu \mathrm{d}t + S_t \sigma \mathrm{d}B_t \\ S_0 = s_0 \end{cases} \tag{4}$$

Further we have:

$$\frac{1}{S_t}\,\mathrm{d}S_t = \mu \mathrm{d}t + \sigma \mathrm{d}B_t \tag{5}$$

Substitute it into equation (3), we obtain the stochastic differential equation of S_t:

$$\mathrm{d}\ln\left(S_t\right) = \left(\mu - \frac{1}{2}\sigma^2\right)\mathrm{d}t + \sigma \mathrm{d}B_t \tag{6}$$

Solve the stochastic differential equation [5], we derive the expression of S_t:

$$S_t = s_0 \exp\left[\left(\mu - \frac{1}{2}\sigma^2\right)t + \sigma B_t\right] \tag{7}$$

Here R.V. B_t follows a standard normal distribution, which means $B_t \sim N(0,1)$. Parameters μ and σ determine the expectation curve of our prediction and B_t determines the fluctuation based on expectation curve. If we can estimate parameters μ and σ, we can use expectation curve to predict the future price trend with equation (7).

4.2 Parameter Estimation by Logistic Regression

Only considering bitcoin market, on day k, we need to predict its price trend in the near future using the data of past days only. If we use all the previous data to make prediction, the result may be mislead by data from a long time ago. Thus, to obtain locally optimal prediction, we use a fixed length (L) of previous data to make prediction. The price on day k is S_k^B, it works as s_0 in equation (7). In addition, we only predict the price of next one day which is enough for us to make decision. That is to say,

$$\hat{S}_k = S_{k-1} \exp\left[\left(\mu - \frac{1}{2}\sigma^2\right) + \sigma B_t\right]. \tag{8}$$

For the estimation of parameter σ refer to [6], the following formula is generally used in practical:

$$\sigma^2 \approx \frac{1}{T_2 - T_1}\sum_{j=0}^{L-1}\left[\ln\left(\frac{S_{t_{j+1}}}{S_{t_j}}\right)\right]^2 = \frac{1}{L}\sum_{j=0}^{L-1}\left[\ln\left(\frac{S_{k-L+j+1}}{S_{k-L+j}}\right)\right]^2 \tag{9}$$

To estimate parameter μ, we apply Logistic Regression to data on previous L days. The fitted function is equation (7) with σ already estimated. We find out that the order of magnitude of μ is 10^{-3}, while that of σ^2 is 10^{-5}. Therefore, when predicting the future price, we can ignore the influence of σ. Hence, we can use $\hat{S}_k = S_{k-1}e^{\mu}$ as the approximation of predicted price. The predicted rate of return is $\hat{R}_k = e^{\mu} - 1$.

For the gold market, we have the similar strategy. According to gold trading rules, we do not trade gold and its price remains on weekends.

Besides, we choose 10 days as a window period refer to [1], which means $L = 10$. Length L can not be too short or too long. If L is too short, the prediction may be very high or very low. If L is too long, the local optimality of prediction will decrease. In addition, the parameter L can be adjusted during distinguish scenarios such as stock market, which increases the scalability of our model to a great extent.

4.3 Implement of Prediction Model

Applying our prediction model to the given data, we derive a series of pictures of actual price and predicted price. To emphasize again, the prediction model only predict the price of one day. We store the prediction of each day and derive figure 4.3. As is shown in figure 4.3, we can see that the overall actual price and predicted price is almost overlapped which means we have a very good prediction in the local realm.

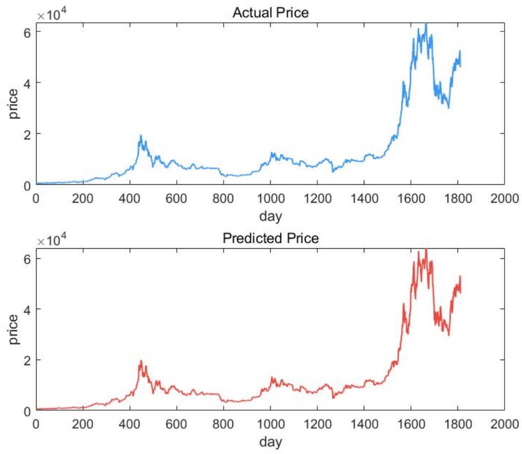

Figure 1: The Actual Price and Predicted Price of bitcoin

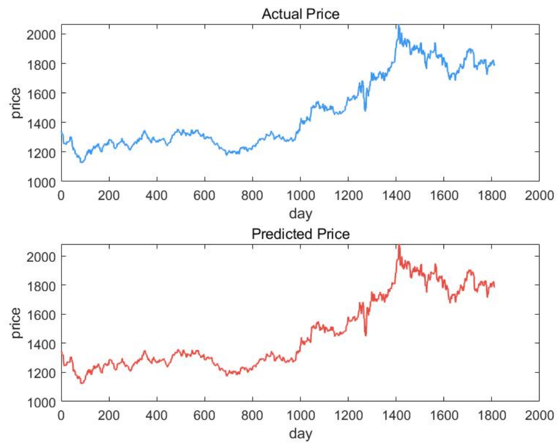

Figure 2: The Actual Price and Predicted Price of gold

Zoom in on figure 1 and 2, we obtain figure 3 and 4. From figure 3 and 4, we tell that the general trend of predicted price and actual price is the same, but prediction has a little delay. We think such delay is reasonable because it accords with the forecast rule [7]. For example, while the price keeps rising in a period of time, the prediction will also be rising. Only when the drop appears will the prediction starts to drop. There is always a delay in prediction.

We then compute the curve of 97.5% confidence level and 2.5% confidence level to see the boundary of our prediction. Figure 5 and 6 show the expectation curve of prediction, 97.5% confidence intervals of prediction and 2.5% confidence intervals of prediction. The boundary between 2.5% and 97.5% is small, which means it is reasonable to use expectation curve as the value of predicted price. It is also consistent with our analysis that μ determines more than σ.

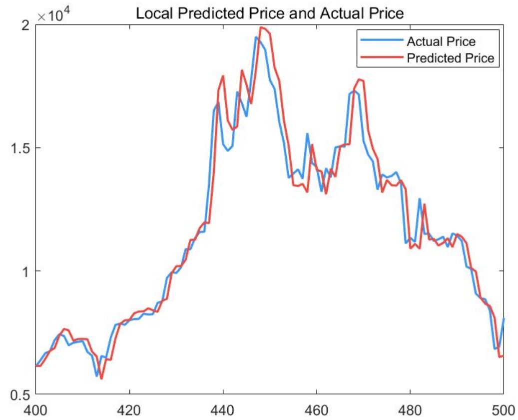

Figure 3: Local Actual Price and Prediction of bitcoin

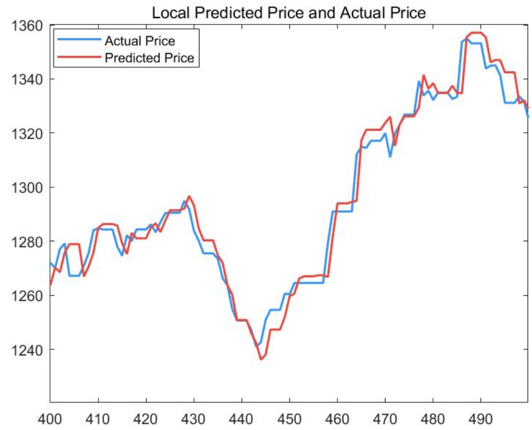

Figure 4: Local Actual Price and Prediction of gold

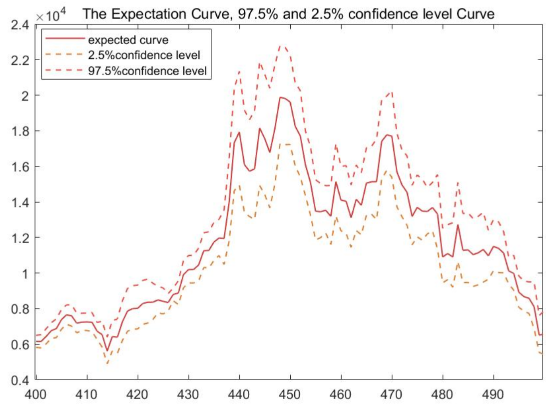

Figure 5: The Expectation Curve and its fluctuation range of bitcoin

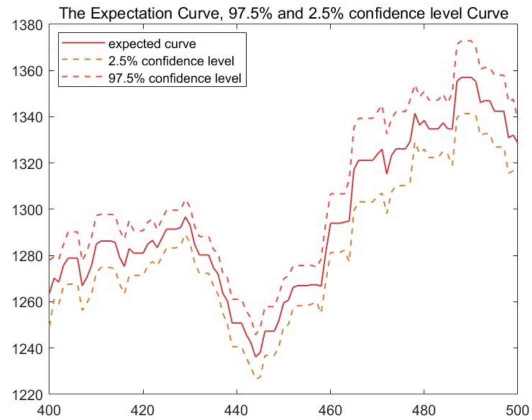

Figure 6: The Expectation Curve and its fluctuation range of gold

5 Decision Making Strategy

5.1 Cross-Market Analysis

We first separate gold market and bitcoin market into two independent markets, and divide cash C into two parts as well. One part is used to invest gold, represented by C_G, while the other part is used to invest bitcoin, represented by C_B. For traders, they do not know which asset has more potential at the beginning. Thus, we treat gold market and bitcoin market equally, giving them 500 $ each. During the investment period, operations within each market are independent and markets do not influence each other. The bond between markets is only cash.

The profitability of two markets are different, so it is necessary to adjust the proportion of investment according to market performance. We set one month as an evaluation period, every month the performance of two markets will be evaluated. Then the portfolio will be reallocated based on evaluation results. Each market has a monthly rate of return R^m, which can be used as an indicator to measure the market performance. If one market behaves worse than the other one, its portfolio will be sold into cash and flow into the other market.

Here are some detailed rules about investment between markets. At the end of each month, the rate of return is calculated:

$$R^m = \frac{S_{28} - S_1}{S_1} \tag{10}$$

We only consider the price on the start of month and the end of month, because it reflects the overall performance of market. Then we compare the monthly rate R_B^m and R_G^m. If $R_B^m > R_G^m$, we move all the existing C_G into C_B. The transfer between cash costs nothing. Then define

$$\Delta R = R_B^m - R_G^m \tag{11}$$

if $R_B^m > R_G^m + \alpha_{gold}$, that means the expected income of bitcoin next month can make up for losses from selling gold. As a result, we can sell some amount of the gold and transfer C_G from gold sale to C_B. The amount is decided by $\Delta R - \alpha_{gold}$ and the proportion of R_B^m. The proportion is defined by

$$k = \frac{R_B^m}{R_B^m + R_G^m} \tag{12}$$

Likewise, if $R_B^m < R_G^m$, we apply the same strategy. The pseudocode 1 reflects the asset transferring process.

Algorithm 1 Assets Transfer between Markets

procedure ASSETS TRANSFER
repeat
  monthly calculate R_B^m, R_G^m
  if $R_B^m > R_G^m$ **then**
    $C_B \leftarrow C_G$
    if $R_B^m > R_G^m + \alpha_{gold}$ **then**
      sell $f(\Delta R - \alpha_{gold})$ amount of gold
      $C_B \leftarrow C_G$
    end if
  end if
  if $R_G^m > R_B^m$ **then**
    $C_G \leftarrow C_B$
    if $R_G^m > R_B^m + \alpha_{bitcoin}$ **then**
      sell $f(\Delta R - \alpha_{bitcoin})$ amount of bitcoin
      $C_G \leftarrow C_B$
    end if
  end if
until End of Investment

5.2 Independent Market Analysis

In this section, the investment strategy within a single market is discussed. The investment strategy is divided into the first 60 days and the days after that. In the first 60 days, for traders who are not familiar with the market, it is wise to observe price trend and make conservative decisions. After 60 days, knowing the probable regularity, traders can make bolder decisions to get more return. The decision within a single market is made everyday. For the first 60 days, we introduce relative error E_{rr} as an indicator:

$$E_{rr} = \frac{S_{k-1} - \hat{S}_{k-1}}{S_{k-1}} \tag{13}$$

Another indicator is the daily rate of return R^d.

On day k we make the decision with knowing the predicted price and the actual price of yesterday. As is shown in figure 7, for some time the price keeps rising, so the prediction model will produce high prediction output $\hat{S}_{k-1}$. However, if the actual price S_{k-1} is smaller than S_{k-2}, it means a drop in price is likely to happen. In this situation, $E_{rr} < 0$ and $S_{k-1} < S_{k-2}$, or $R^d_{k-1} < 0$. When this happens, we should sell all the asset to keep safe. Even if we may lose money, we can avoid losing much more money in the next few days. At the same time, if $E_{rr} > 0$ and $R^d_{k-1} > 0$, it means the market is rising rapidly and we buy the asset with all the money in that market. If $E_{rr} > 0$ and $R^d_{k-1} < 0$ or $E_{rr} < 0$ and $R^d_{k-1} > 0$, do not make any move to avoid extra cost caused by transaction rate. The first 60 days is the conservative period, it is acceptable to lose little or earn little.

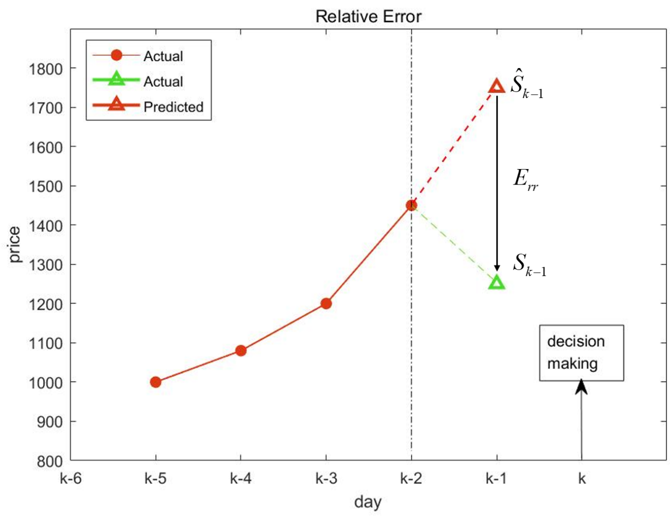

Figure 7: Relative Error

After 60 days, we use Moving Average Method to make decision. Moving Average Method is a economical method widely used in stock investment. We define the N-day average of day k as the mean value of N days before day k. Jogepsb Ganvle, a U.S. investment expert, put up 8 rules to optimize trading time [1]. Half of the eight rules are used to determine when to buy, while the other half are used to determine when to sell. In general, when the moving average is below the stock price and it is trending upwards, we should buy. On the contrary, when the moving average is above the stock price line and it is trending downwards, we should sell.

According to the Moving Average Theory [8], the recent stock price can be represented by a short term moving average. Meanwhile, the overall trend of the stock is mainly reflected by a long term moving average. We set 10 days as a short short moving average and 60 days as a long moving average. We choose 10 days also because the input of our prediction model is 10. Then define: $\beta =$ 10-day average $-$ 60-day average as the measure of our investment. When β is high and going upwards, it means the market is in a good condition and we can wait the asset to appreciate. When β reaches its extreme and the first time it drops, we need to judge whether β is larger than the threshold value γ_{max}. If β is relative small, we think the market tend to saturate and it is the time to sell. For β is low, we adopt the same approach using the turning point and γ_{min} to determine when to buy.

One thing that needs emphasize is how to deal with gold on weekends. According to trade regulations, the gold market cannot be open on weekends. Thus, we assume that the actual price of gold on weekends is the same as that on Friday. Besides, no trade can be allowed to proceed. In this way, the curve of gold assets is continuous and it is convenient

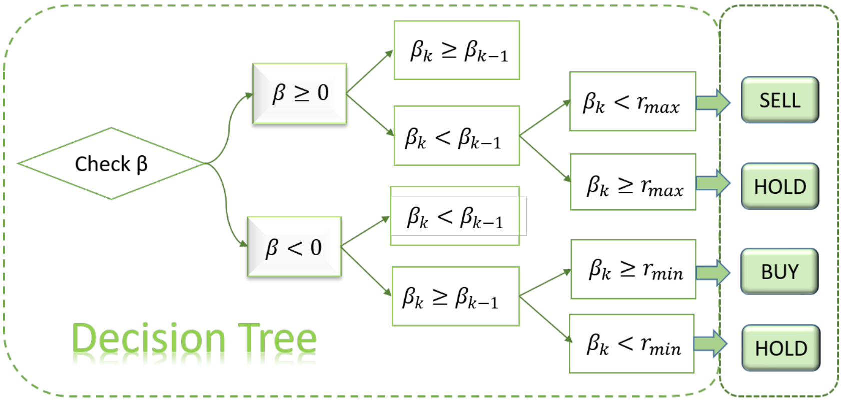

Figure 8: Decision Tree of β

to make comparison with bitcoin.

Combing the Cross-Market analysis and Independent market Analysis, we derive the general framework of decision making process in figure 9.

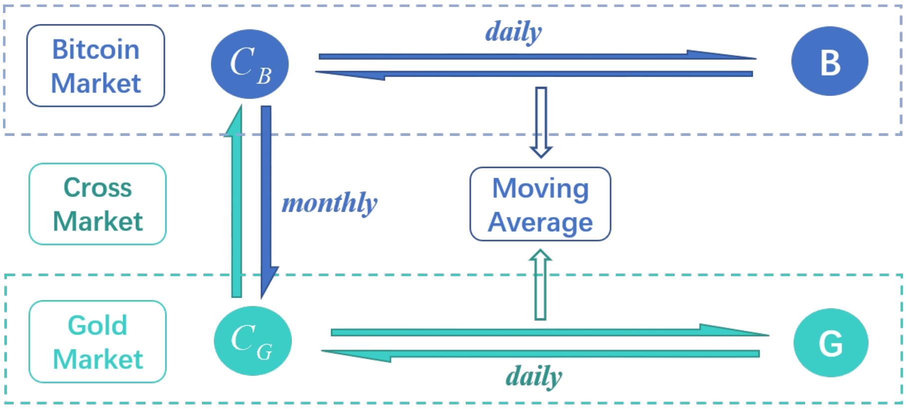

Figure 9: Framework of Cross-market Analysis

5.3 Optimality Proof

This section provides evidence that the Cross-Market model produces the optimal strategy. Below are the strategies and the proof of optimization.

- **Rule 1**: Compare month return rate R_G^m, R_B^m, then transfer all the cash of that market to the other market if it has a lower R^m. Assume that the price of this market grows at the same rate $R_i^m, i = G, B$ next month.

  Proof.

$$\begin{aligned} C &= C_G + C_B \\ Q_C &= C_G\left(1 + R_G\right) + C_B\left(1 + R_B\right) - C_G - C_B \\ &= C_G R_G + C_B R_B \leqslant \left(C_G + C_B\right) R_B = C R_B \end{aligned} \tag{14}$$

  Equal if and only if $C_B = C, C_G = 0$. That means transferring all the C_G into C_B is the optimal strategy. □

- **Rule 2**: If $R_B > R_G + \alpha_{gold}$, then sell gold into C_G and transfer it to C_B. The same is true with the condition $R_G > R_B + \alpha_{bitcoin}$.

Proof. Assume that $R_B > R_G + \alpha_{gold}$. If we do not transfer gold to bitcoin, the estimated profit next month is Q_1. If we transfer $k\%$ of gold to bitcoin, the estimated profit next month is Q_2. We have:

$$\begin{aligned}
&Q_1 = G\left(1+R_G\right) + B\left(1+R_B\right) - G - B \\
&= GR_G + BR_B \\
&Q_2 = G(1-k\%)\left(1+R_G\right) + \left(B + k\%G\left(1-\alpha_{gold}\right)\right)\left(1+R_B\right) \\
&= GR_G + BR_B - k\%G\left(1+R_G\right) \\
&= Q_1 + k\%G\left(1 + R_B - \alpha_{gold} - \alpha_{gold}R_B - 1 - R_a\right) \\
&= Q_1 + k\%G\left(R_B - \alpha_{gold} - R_G - \alpha_{gold}R_B\right) \quad \alpha_{gold}R_B \approx 0 \\
&= Q_1 + k\%G\left(R_B - \alpha_{gold} - R_G\right) > Q_1
\end{aligned} \tag{15}$$

That means any transfer from gold to bitcoin creates better return. □

5.4 Implement of Decision Making Model

We draw the moving average line of gold and bitcoin in figure 10 and 11. For each asset, subtract the 60-day average line from 10-day average line, we derive the β line. The exchange point is plotted with solid dot. The red dot is the buying point, the blue dot is the selling point. As is shown in result, there are **6** buy in and **3** sell out in bitcoin, **8** buy in and **4** sell out in gold. For bitcoin, it buys in on day **184, 364, 615, 1164, 1294, 1452**, and sells out on day **250, 449, 1571**. For gold, it buys in on day **295, 536, 592, 1130, 1184, 1278, 1660, 1744**, and sells out on day **557, 1212, 1311, 1678**.

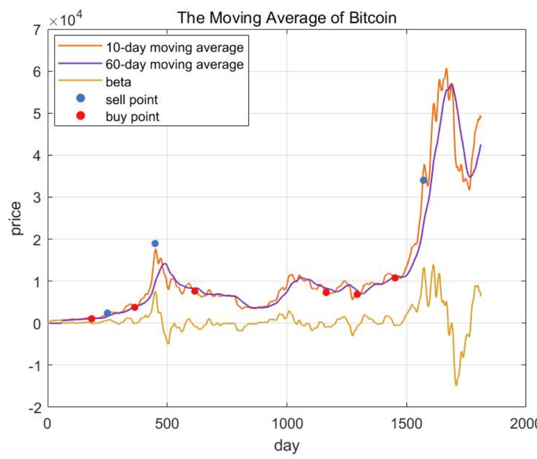

Figure 10: The Moving Average of Bitcoin

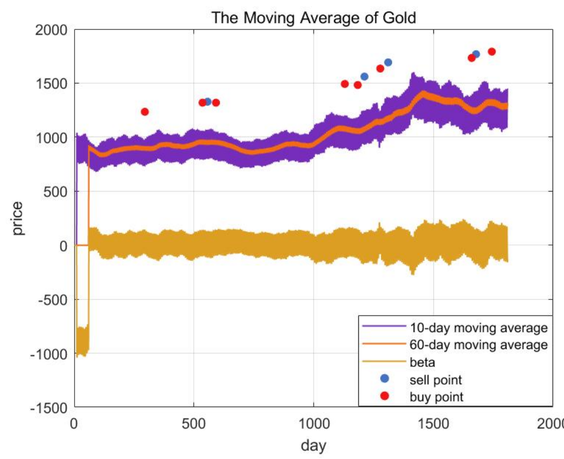

Figure 11: The Moving Average of Gold

The value of buy in point and sell out point is the actual price. Since the actual price curve almost overlaps the 10-day moving average curve, ew do not plot the actual price. Seen from figure 11, we know that the price of gold is quite stable, its actual price fluctuates between its average. That is why it looks dense.

Then, we draw the asset of each currency during the whole investment period. It reflects the portion we hold of each currency. We use USD as their unit uniformly. From figure 12, we can see that bitcoin is more risky than gold. At about day 500, the bitcoin are sold and transferred to gold because it is safer. However, bitcoin has greater earning ability. Once the cash is transferred into bitcoin market, they are all used to purchase bitcoin. At about day 1600, bitcoin starts to shock violently and we sell all the bitcoin into gold. We missed a higher profit but gained more safety.

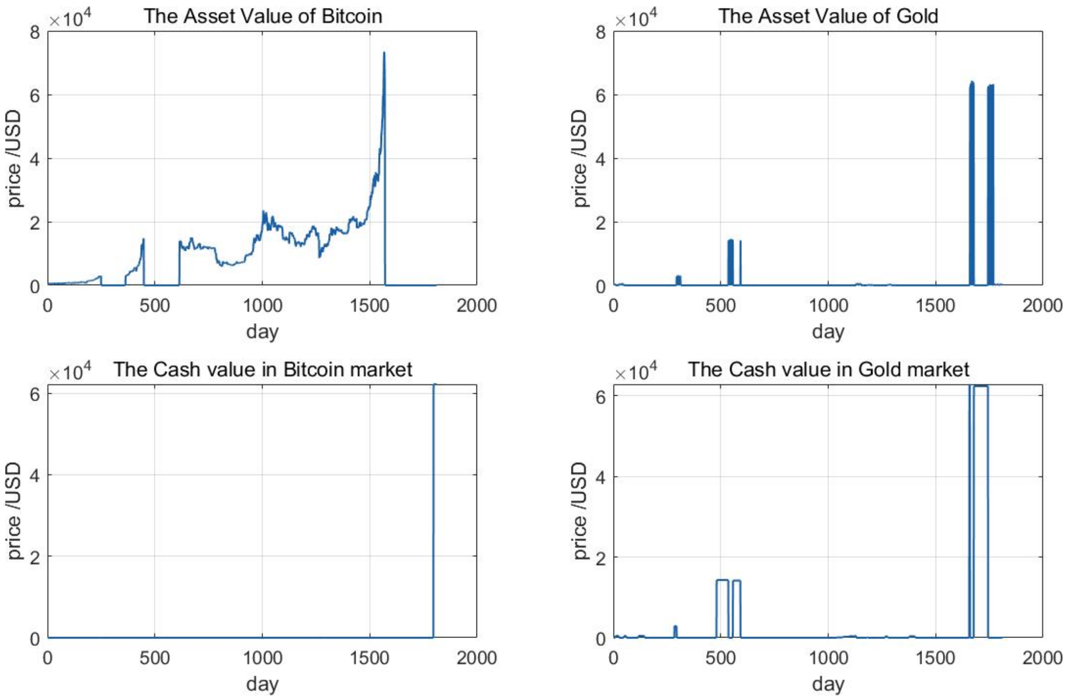

Figure 12: The Asset Flow through the whole Investment

Figure 13 is the total assets we have. That is the combination of gold, bitcoin and cash. The final asset we have is **62513.5 $**, that is 63 times of the initial assets. The unit is USD.

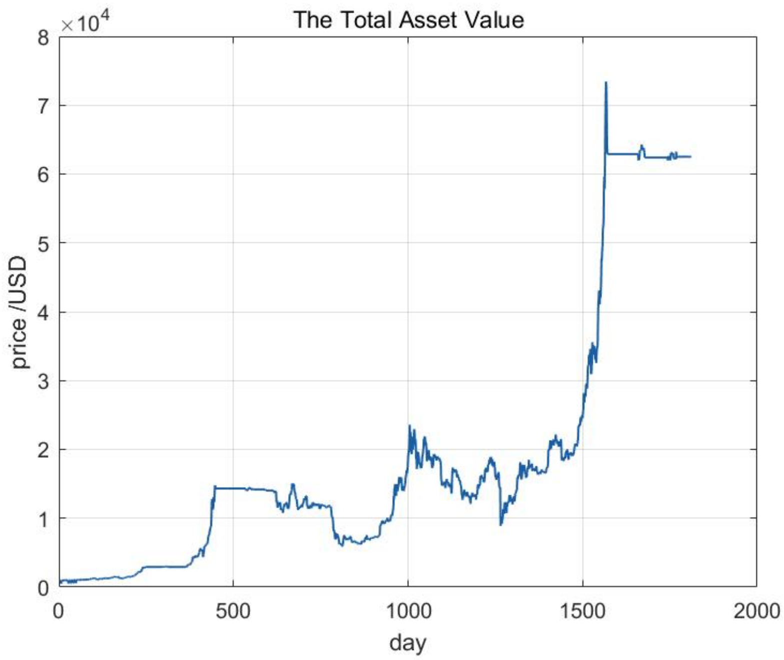

Figure 13: Total Assets

6 Sensitivity Analysis

To prove that our model is stable, we conduct a sensitivity analysis in this section. In the previous sections, the transaction rate of gold and bitcoin is $\alpha_{gold} = 1\%$ and $\alpha_{bitcoin} =$

2%. We modify the value of α to see whether the output will produce large fluctuations. We first fix α_{gold}, then change the value of $\alpha_{bitcoin}$. The original total assets and fluctuation range is shown in figure 14. Next,we fix $\alpha_{bitcoin}$ and test the influence of α_{gold} to the total assets. The result is presented in figure 15. Finally, we modify the value of α_{gold} and $\alpha_{bitcoin}$ simultaneously and the result is shown in figure 16 and 17.

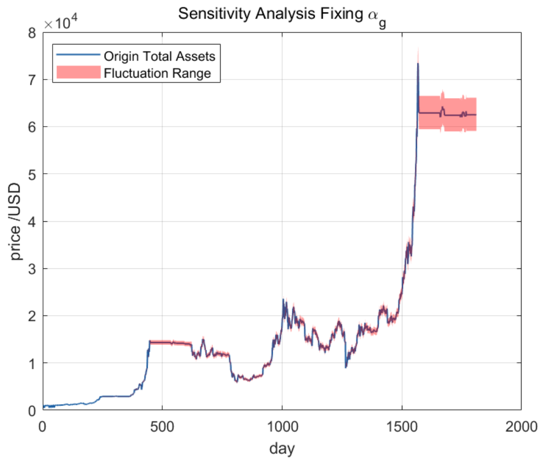

Figure 14: Sensitivity Analysis Fixing α_{gold}

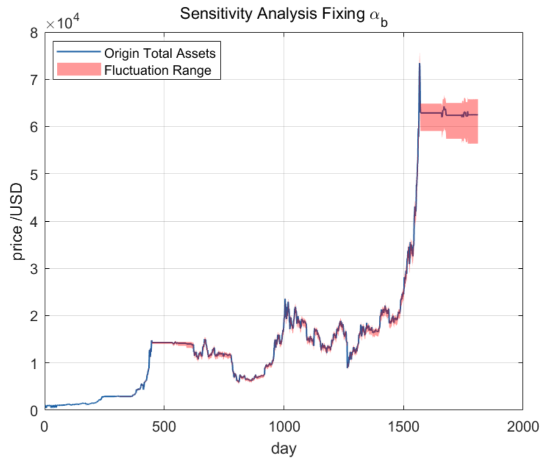

Figure 15: Sensitivity Analysis Fixing $\alpha_{bitcoin}$

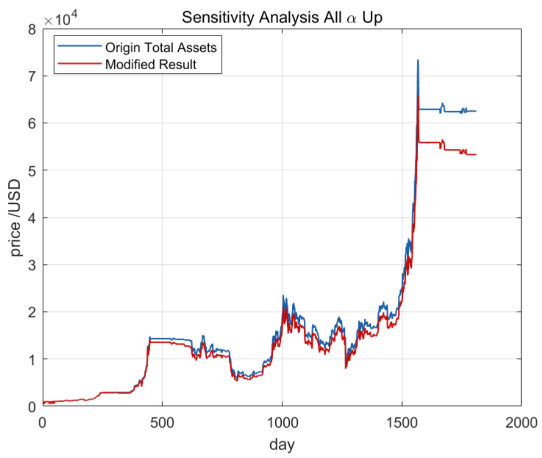

Figure 16: Sensitivity Analysis All α Up

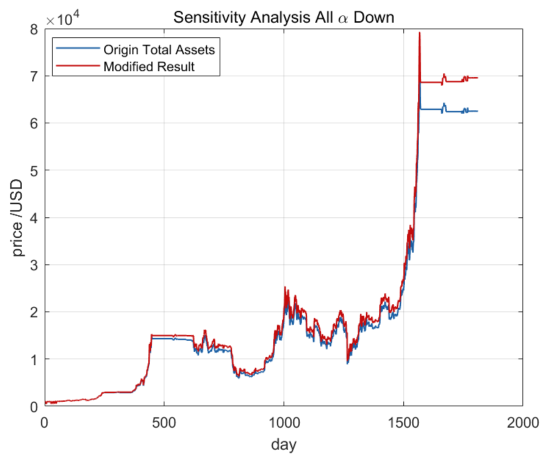

Figure 17: Sensitivity Analysis All α Down

2 a table to include max % and min % and final %

Define the fluctuation rate by $(A_{new} - A_{old})/A_{old}$, we obtain the values in table 2. Table 2 presents the fluctuation rate of different situations. We consider the total asset under $\alpha_{gold} = 1\%$ and $\alpha_{bitcoin} = 2\%$ as a base reference. The new total asset with modified α_{gold} or $\alpha_{bitcoin}$ is computed, and we further compute the fluctuation rate. The first row is the new final assets after we modify the parameter, ignoring the process. The second row is the fluctuation rate of the final assets. The third row and the fourth row is the maximum fluctuation rate and minimum fluctuation rate. The last row is the mean fluctuation rate.

From the figures and table above, we can see that our model is not sensitive to α during most of the invest process. But at the end of investment, the fluctuation rate is a little bit high due to the wild fluctuation of bitcoin. Seen as a whole, our model is not sensitive.

Table 2: Data of Sensitivity Analysis

	$\alpha_G = 1\%$		$\alpha_B = 2\%$		$\alpha_B = 3\%$	$\alpha_B = 1\%$
	$\alpha_B = 3\%$	$\alpha_B = 1\%$	$\alpha_G = 2\%$	$\alpha_G = 0.5\%$	$\alpha_G = 2\%$	$\alpha_G = 0.5\%$
Final	59087.94	66103.49	56402.54	65788.27	53313.3	69567.27
Final rate	-5.48%	5.74%	-9.78%	5.24%	-14.72%	11.28%
Max	0	0.0574	0	0.0524	0	0.1128
Min	-0.0548	0	-0.0978	0	-0.1472	0
Average	-0.0357	0.037	-0.046	0.0239	-0.0797	0.0621

7 Strengths and Weaknesses

7.1 Strengths

- We developed two general decision strategy based on the situation of lack-of-data and rich-of-data respectively, which can help investors get the best return in **both the short term and the long term**.
- Combining the feature of **high quality, easy-to-understand and low computational complexity**, our model is useful for every one who wants to invest in gold and bitcoin markets to make money.
- Our model is **explainable** and can be proved theoretically. We proof that our Cross-Market model can produce the optimal strategy mathematically.
- Our model is **highly portable**. Because the principles behind our model are highly accepted and widely used in economical fields, we can easily transfer our model to another finance market such as stock market by adjusting some parameters.
- Our model is **strong and stable**. The change of exchange rates will cause minor impact on our model.

7.2 Weaknesses

- Countless tiny factors affect the real financial market, which are not even considered exhaustively by economists. We only take explainable and theoretically-based factors into account, **thus can not completely coincident with the real situation, which will bring acceptable investment risk into the model**. As a result, invest with caution.
- Due to the requirement of using only the past stream of daily prices to date, we have to **do some proper assumption to constrain our model and may bring some bias into the model**.
- Considering the potential huge risk in bitcoin, **our model may not fall into the condition of getting the most earnings with extremely low probability**.

8 Conclusion

In this paper, we proposed a Cross-Market Investment Strategy composed of the prediction model and decision model. The prediction model predicted the price of gold and bitcoin of the following day with data-driven methods. For the decision making part, our strategy separated the bitcoin market and the gold market and invested in them respectively. We also considered the transfer of the assets between two markets. Apart from that, we tested the sensitivity of our model to transaction rate and analyzed the strengths and weaknesses of our model. Generally, our model solved the problems of price prediction and strategy optimization.

There are still expectations to our model. How to get better parameters such as using deep learning, and how to integrate the two markets better are still open problems waiting to be solved in the future.

References

[1] "Joseph granville," Wikipedia. [Online]. Available: https://en.wikipedia.org/wiki/Joseph_Granville

[2] Ö. Korkmaz, "The relationship between bitcoin, gold and foreign exchange returns: The case of turkey," *Turkish Economic Review*, vol. 5, no. 4, pp. 359–374, 2018.

[3] W. Rodi and N. Fueyo, *Engineering turbulence modelling and experiments 5*. Elsevier, 2002.

[4] K. Reddy and V. Clinton, "Simulating stock prices using geometric brownian motion: Evidence from australian companies," *Australasian Accounting, Business and Finance Journal*, vol. 10, no. 3, pp. 23–47, 2016.

[5] A. S. Poznyak, *Advanced mathematical tools for automatic control engineers: Stochastic techniques*. Elsevier, 2009.

[6] L. B. Hailin Feng, *Stochastic Processes: Computation and Applications*. Xidian University Press, 2012.

[7] J. Wang and J. J. Wu, "The taylor rule and forecast intervals for exchange rates," *Journal of Money, Credit and banking*, vol. 44, no. 1, pp. 103–144, 2012.

[8] J. E. Granville, *Granville's new strategy of daily stock market timing for maximum profit*. Prentice-Hall, 1976, vol. 6.

A Memorandum to Traders

Dear Traders,

Are you still puzzled by the trading strategies of digital currency? Here we provide you with a fantastic solution. Our model has the function of price predicting and decision making and can maximize the return with money you have. Now we are writing to share it with you.

When making predictions of the price, it is unwise to forecast too far because lots of things can influence the price change. Our suggestion is to focus on tomorrow's prediction. In addition, do not use too much previous data to make the prediction. If you use too much previous data, it will be a long-term prediction. Certainly, we want short-term prediction because it is more accurate. Our model uses 10 days to predict the next day, the result turns out quite well. You can also try to dynamically adjust the length of data for prediction, which are not completed in this contest. Geometric Brownian Motion is a perfect model for price prediction which is generally used in the stock market. About the parameters, just apply Logistic Regression or other machine learning methods to historical data.

As for making decision, if you want to invest in a new asset, we recommend you to make conservative decisions at first. Spend a little time observing, and record its price data. Our strategy is to consider the daily rate of return, and the error between the predicted price and the actual price. For a rising currency, if its return rate becomes negative and the actual price blows the predicted price, it is likely that there will be a drop in the near future. At this time, youd better sell the currency.

After a period of time, for example, 60 days, if you find your currency is safe and even has the potential to make benefits, then you can make bolder decisions. Our strategy is the Moving Average method. It simply calculates the average price of the last period of time. Choose a short-term moving average and a long-term average. According to Ganvles 8 rules, when the moving average is below the stock price and it is trending upwards, it is time to buy. On the contrary, when the moving average is above the stock price line and it is trending downwards, it is time to sell. That is the investment strategy in a single market. What's more, if there are different kinds of assets such as gold and bitcoin, you can monthly reallocate your money according to their behavior in that month. If bitcoin earns more in one month, then invest more money in bitcoin, or even sell gold and give money to bitcoin next month. That is our Cross-Market methodology.

Starting on 9/11/2016 with 1000 $, our total assets is 62513.5 $ on 9/11/2021. With different transaction rates, the income does not vary too much. That means our model is quite stable and worth a try. We do not have too many transactions so you do not need to worry about the transaction fee. By the way, our prediction model is excellent, too. Look at our prediction curve and the true curve, the prediction tightly follows the actual price.

Finally, we have to say that market is affected by various factors. Be sure to keep an eye on the news! Some people can change the market tremendously with a sentence on their social media. The sharp increase and decrease of bitcoin after 2020 is almost attributed to Elon Musk. So read the news and that may bring you wealth.

Yours sincerely,

Team 2202238

Appendices

Appendix A Code for Prediction Model

```
import math
import matplotlib.pyplot as plt
from sklearn.datasets import load_iris
from sklearn.linear_model import LogisticRegression
from sklearn import linear_model
'''
whether gold or bitcoin: 1 for gold, 2 for bitcoin
'''
ch = 2
G_or_B = gold_data['USD (PM)']
G_or_B_date = gold_data['date']
G_or_B_Date = gold_data['Date']
filename = ''
if ch == 1:
    G_or_B = gold_data['USD (PM)']
    G_or_B_date = gold_data['date']
    G_or_B_Date = gold_data['Date']
    filename = 'gold'
elif ch == 2:
    G_or_B = bitcoin_data['Value']
    G_or_B_date = bitcoin_data['date']
    G_or_B_Date = bitcoin_data['Date']
    filename = 'bitcoin'
step_interval = 10 # window size
G_or_B_len = len(G_or_B) # length of bitcoin_data
G_or_B_E_St = [] # prediction
G_or_B_sigma2 = [] # sigma ^ 2: variance of ln_St
G_or_B_mu = [] # mu: exchange_rate of return
next_interval = 1 # how many days to predict
next_E_St = [] # prediction values from current day to 'next_interval' days later
next_St_va = []
R_list = []
B_t = 0.0
filename = '-'.join([filename, str(step_interval), str(next_interval)]) + '.csv'
for day in range(0, step_interval):
    G_or_B_E_St.append(G_or_B[day])
for day in range(step_interval, G_or_B_len):
    X, y = [[0]], [0]
    start_day = day - step_interval
    ln_S0 = math.log(G_or_B[start_day])
    for end_day in range(start_day + 1, day):
        ln_St = math.log(G_or_B[end_day])
        t = end_day - start_day
        # compute sigma * sigma using formula
        sigma2 = 0
        for i in range(start_day, end_day):
            sigma2 = sigma2 + math.pow(math.log(G_or_B[i + 1] / G_or_B[i]), 2)
        sigma2 = sigma2 / t

        X.append([t])
        y.append(ln_St - ln_S0 + 0.5 * sigma2 * t - B_t * math.sqrt(sigma2))
    '''
    linear regression
    '''
    clf = linear_model.LinearRegression()
    clf.fit(X, y)
    # compute sigma * sigma using formula
    t = day - start_day
    sigma2 = 0
    for i in range(start_day, day):
        sigma2 = sigma2 + math.pow(math.log(G_or_B[i + 1] / G_or_B[i]), 2)
    sigma2 = sigma2 / t
    G_or_B_sigma2.append(sigma2)
    # mu
    mu = clf.coef_[0]
    G_or_B_mu.append(mu)
    # R
    R = math.exp(mu) - 1
    R_list.append(R)
    # next_St_mean
    next_St_mean = G_or_B[day - 1] * np.vectorize(math.exp)(mu * \
                   np.linspace(1, next_interval, num=next_interval))
    next_E_St.append(next_St_mean.tolist())
    G_or_B_E_St.append(next_St_mean[0])
    # next_St_va
```

```
    next_St_va = [math.pow(G_or_B[day - 1], 2) * math.exp(2 * mu) \
                * (math.exp(sigma2) - 1)]
    for i in range(day + 1, day + next_interval):
        next_St_va.append(math.pow(next_St_va[i - day - 1], 2) \
        * math.exp(2 * mu) * (math.exp(sigma2) - 1))
    next_St_va.append(next_St_va[0])
X = np.linspace(0, G_or_B_len - 1, num=G_or_B_len).tolist()
y = G_or_B.tolist()
y_pred = G_or_B_E_St
plt.plot(X, y, 'r-', X, y_pred, 'b-')
```

Appendix B Code for Decision Model

```
% This is prediction model for the first 60 days
ex_rate = [1, 0.99, 0.98]; % cash, gold, bitcoin
short_term_gap = 10;
long_term_gap = 60;
B_St = table2array(bitcoin101(:, 1));
B_E_St = table2array(bitcoin101(:, 2));
B_sigma2 = table2array(bitcoin101(:, 7));
B_R = table2array(bitcoin101(:, 6));
G_St = table2array(gold101(:, 1));
G_E_St = table2array(gold101(:, 2));
G_sigma2 = table2array(gold101(:, 7));
G_R = table2array(gold101(:, 6));
len = length(B_St); % 1811
begin_days = 60;
% Handle first 60 days
ptn_60 = [500, 0, 500, 0]; % gold_cash, gold, bitcoin_cash, bitcoin
data_list = []
for day = 1 : begin_days
    if day == 1
        ptn_60(2) = (ptn_60(1) / G_St(day)) * ex_rate(1) * ex_rate(2);
        ptn_60(1) = 0;
        ptn_60(4) = (ptn_60(3) / B_St(day)) * ex_rate(1) * ex_rate(3);
        ptn_60(3) = 0;
    else
        % Handle gold
        G_prev = -1;
        if mod(day - 1, 7) == 5 || mod(day - 1, 7) == 6 % Weekends not Work
            G_prev = -1;
        elseif mod(day - 1, 7) == 0 % Today is Monday, so use last Friday's
            G_prev = day - 3;
        else
            G_prev = day - 1;
        end
        if G_prev ~= -1
            if G_R(G_prev) > 0 && G_St(G_prev) > G_E_St(G_prev) ...
                    && ptn_60(1) > 0 % case 1: all in
                ptn_60(2) = ptn_60(2) + (ptn_60(1) / G_St(G_prev)) ...
                    * ex_rate(1) * ex_rate(2);
                ptn_60(1) = 0;
            elseif G_R(G_prev) < 0 && G_St(G_prev) < G_E_St(G_prev) ...
                    && ptn_60(2) > 0 % case 4: all out
                ptn_60(1) = ptn_60(1) + (ptn_60(2) * G_St(G_prev)) ...
                    * ex_rate(1) * ex_rate(2);
                ptn_60(2) = 0;
            end
        end
        % Handle bitcoin
        B_prev = day - 1;
        if B_R(B_prev) > 0 && B_St(B_prev) > B_E_St(B_prev) ...
                && ptn_60(1) > 0 % case 1: all in
            ptn_60(4) = ptn_60(4) + (ptn_60(3) / B_St(B_prev)) ...
                * ex_rate(1) * ex_rate(3);
            ptn_60(3) = 0;
        elseif B_R(B_prev) < 0 && B_St(B_prev) < B_E_St(B_prev) ...
                && ptn_60(3) > 0 % case 4: all out
            ptn_60(3) = ptn_60(3) + (ptn_60(4) * B_St(B_prev)) ...
                * ex_rate(1) * ex_rate(3);
            ptn_60(4) = 0;
        end
    end
    tot = ptn_60(1) + ptn_60(3) + ptn_60(2) * G_St(day) + ptn_60(4) * B_St(day);
    data_list(end + 1, :) = [ptn_60(4), ptn_60(3), ptn_60(2), ptn_60(1), tot];
end

%This is the function for the indepent market analysis
function [newB,newCB,trade_information,trade_node, daily_store_bit]=...
```

```
    bitcoin_operation(startday, endday, S_t_true,  B, CB)
% Input:
% startday: starting time of the program  an integer between 1 and 1811
% endday: end time of the program         an integer between 1 and 1811
% category: 0 represent bitcoin, 1 represent gold
% bitcoin_S_t_actual 1811 * 1 double
exchange_rate = [1, 0.99, 0.98];
short_term_gap = 10;
long_term_gap = 60;
daily_store_bit=[];
sub_store=[];
bitcoin_S_t_actual = S_t_true;
len = length(bitcoin_S_t_actual); % 1811
trade_information = [];
trade_node = [];
threshold_pos = 0.3;
threshold_neg = 0.03;
portion = [CB, 0, B];  % cash, gold, bitcoin
trade_point = 0;
for i = startday: endday-1
    short_term_average(i) = sum(bitcoin_S_t_actual(i-9:i))/10;
    long_term_average(i) = sum(bitcoin_S_t_actual(i-59:i))/60;
    delta(i) = short_term_average(i) - long_term_average(i);
    k = (short_term_average(i) - long_term_average(i))...
        / short_term_average(i); %bitcoin_S_t_actual(i - 1);
    flag = 0;
    if k > threshold_pos && delta(i) < delta(i - 1)
        if portion(3) > 0
            trade_point = trade_point + 1;
            trade_node(end+1)=i;
            flag=2;
            trade_information(end+1)=portion(3);
        end
        portion(1) = portion(1) + portion(3) * exchange_rate(3)...
            * exchange_rate(1) * bitcoin_S_t_actual(i - 1);
        portion(3) = 0;
    elseif k < -threshold_neg && delta(i) > delta(i - 1)
        if portion(1) > 0
            trade_point = trade_point + 1;
            trade_node(end+1)=i;
            trade_information(end+1)=portion(1);
            flag=1;
        end
        portion(3) = portion(3) + portion(1) * exchange_rate(1)...
            * exchange_rate(3) / bitcoin_S_t_actual(i - 1);
        portion(1) = 0;
    end
    sub_store = [sub_store portion(3)];
    sub_store = [sub_store portion(1)];
    sub_store = [sub_store flag];
    daily_store_bit = [daily_store_bit;sub_store];
    sub_store = [];
end
cash = 1 * portion(1) + bitcoin_S_t_actual(i) * portion(3);
newCB = portion(1);
newB = portion(3);

%main function using the cross-market model
for i= startday : endday
    if mod(i,28)==5
        R_G_month = (gold_true(i) - gold_true(i-28)) / gold_true(i-28);
        R_B_month = (bitcoin_true(i) - bitcoin_true(i-28)) / bitcoin_true(i-28);
        R_G_total = (gold_true(i) - gold_true(1)) / gold_true(1);
        R_B_total = (bitcoin_true(i) - bitcoin_true(1)) / bitcoin_true(1);
        if R_G_month > R_B_month
            CG=CB+CG;
            CB = 0;
        else
            CB=CG+CB;
            CG = 0;
        end
        if R_G_month > R_B_month + alpha_B
            k = R_G_total / (R_G_total + R_B_total);
            if k<0
                k=0;
            elseif k>1
                k=1;
            end
            CG = CG + B * bitcoin_true(i) * (1 - alpha_B) * k;
            B = B * (1 - k);
        elseif R_B_month > R_G_month + alpha_G
            k = R_B_total / (R_G_total + R_B_total);
            if k<0
                k=0;
            elseif k>1
```

```
                k=1;
            end
            CB = CB + G * gold_true(i) * (1 - alpha_G) * k;
            G = G * (1 - k);
        end
    end
    if mod(i,28)==5 && i == 61
        [B,CB,a,b,bit_restore] = bitcoin_operation(i, i + 28,...
        bitcoin_true(1 : i + 28),  B, CB);
        bit_info = [bit_info;bit_restore] ;
    elseif mod(i,28)==5
            [B,CB,a,b,bit_restore] = bitcoin_operation(i, i + 28,...
            bitcoin_true(1 : i + 28),  B, CB);
            bit_info = [bit_info;bit_restore] ;
            [G,CG,a,b,gold_restore] = gold_operation(floor(i/28)*20 +5,...
            floor(i/28)*20 + 25,gold_true_nonzero(1 : floor(i/28)*20 + 25)...
            ,  G, CG);
            gold_info = [gold_info;gold_restore] ;
    end
end
```

Appendix C Parameters and Data

Table 3: Gold Data

Date	$S_{tactual}$	E_{St}	$S_{t_{2.5}}$	$S_{t_{97.5}}$	R	$\sigma^2_{\text{formular}}$	μ_{formular}	err
2016/9/26	1340.50	1340.33	1326.37	1354.40	0.00126	2.85E-05	0.00119	0.0000
2016/9/27	1327.00	1343.36	1327.07	1359.81	0.00214	3.86E-05	0.00142	0.0001
2016/9/28	1322.50	1330.01	1313.69	1346.49	0.00227	3.96E-05	0.00046	-0.0123
2016/9/29	1318.10	1325.27	1310.23	1340.43	0.00209	3.38E-05	0.00101	-0.0057
2016/9/30	1322.50	1319.64	1304.50	1334.91	0.00117	3.45E-05	0.00084	-0.0054
2016/10/1	0.00	0.00	0.00	0.00	0.00000	0	0.00000	0.0000
2016/10/2	0.00	0.00	0.00	0.00	0.00000	0	0.00000	0.0000
2016/10/3	1313.30	1322.81	1307.12	1338.64	0.00024	3.70E-05	0.00066	0.0022
2016/10/4	1283.30	1312.15	1287.88	1336.75	-0.00088	9.03E-05	-0.00002	-0.0072
2016/10/5	1269.40	1279.10	1255.03	1303.51	-0.00327	9.35E-05	-0.00361	-0.0225
2016/10/6	1254.50	1262.60	1238.29	1287.26	-0.00535	9.79E-05	-0.00591	-0.0076
2016/10/7	1258.75	1245.85	1221.72	1270.32	-0.00690	9.90E-05	-0.00719	-0.0065
2016/10/8	0.00	0.00	0.00	0.00	0.00000	0	0.00000	0.0000
2016/10/9	0.00	0.00	0.00	0.00	0.00000	0	0.00000	0.0000
2016/10/10	1259.50	1249.31	1225.14	1273.84	-0.00750	9.89E-05	-0.00696	0.0102
2016/10/11	1253.45	1250.38	1227.17	1273.92	-0.00724	9.09E-05	-0.00577	0.0081
2016/10/12	1256.50	1244.47	1221.44	1267.82	-0.00716	9.04E-05	-0.00593	0.0024
2016/10/13	1261.05	1248.34	1225.21	1271.78	-0.00650	9.06E-05	-0.00528	0.0096
2016/10/14	1251.75	1254.44	1230.65	1278.56	-0.00524	9.49E-05	-0.00525	0.0101
2016/10/15	0.00	0.00	0.00	0.00	0.00000	0	0.00000	0.0000
2016/10/16	0.00	0.00	0.00	0.00	0.00000	0	0.00000	0.0000
2016/10/17	1254.80	1247.19	1224.08	1270.63	-0.00364	9.07E-05	-0.00530	-0.0021
2016/10/18	1258.20	1252.66	1237.59	1267.86	-0.00170	3.80E-05	-0.00248	0.0061
2016/10/19	1269.05	1257.45	1243.25	1271.78	-0.00059	3.35E-05	-0.00097	0.0044
2016/10/20	1271.65	1269.69	1258.60	1280.85	0.00050	2.00E-05	0.00129	0.0091
2016/10/21	1266.05	1272.84	1261.50	1284.25	0.00093	2.08E-05	0.00114	0.0015
2016/10/22	0.00	0.00	0.00	0.00	0.00000	0	0.00000	0.0000
2016/10/23	0.00	0.00	0.00	0.00	0.00000	0	0.00000	0.0000
2016/10/24	1265.55	1267.55	1256.26	1278.91	0.00118	2.08E-05	0.00059	-0.0054
2016/10/25	1269.40	1267.29	1256.39	1278.26	0.00138	1.94E-05	0.00108	-0.0016
2016/10/26	1270.50	1271.13	1260.34	1281.98	0.00136	1.89E-05	0.00114	0.0017
2016/10/27	1266.25	1272.27	1261.53	1283.08	0.00140	1.87E-05	0.00084	-0.0005
2016/10/28	1273.00	1267.97	1258.05	1277.95	0.00136	1.60E-05	0.00129	-0.0048
2016/10/29	0.00	0.00	0.00	0.00	0.00000	0	0.00000	0.0000
2016/10/30	0.00	0.00	0.00	0.00	0.00000	0	0.00000	0.0000
2016/10/31	1272.00	1274.35	1264.55	1284.22	0.00106	1.55E-05	0.00161	0.0040
2016/11/1	1288.45	1272.87	1258.98	1286.88	0.00069	3.13E-05	0.00122	-0.0019
2016/11/2	1303.75	1289.78	1274.30	1305.40	0.00103	3.78E-05	0.00170	0.0121

Table 4: Bitcoin Data

Date	$S_{t\text{actual}}$	E_{St}	$S_{t_{2.5}}$	$S_{t_{97.5}}$	R	$\sigma^2_{\text{formular}}$	μ_{formular}	err
2016/9/26	598.98	600.41	590.77	610.17	-0.00221	6.80E-05	-0.00221	0.0000
2016/9/27	605.96	597.70	587.27	608.26	-0.00214	8.02E-05	-0.00215	-0.0024
2016/9/28	605.67	605.07	594.88	615.38	-0.00147	7.47E-05	-0.00147	0.0136
2016/9/29	603.85	605.48	595.26	615.83	-0.00031	7.51E-05	-0.00031	0.0010
2016/9/30	609.39	604.24	593.53	615.09	0.00064	8.28E-05	0.00064	-0.0027
2016/10/1	614.82	610.64	601.09	620.31	0.00206	6.44E-05	0.00206	0.0085
2016/10/2	612.98	616.58	606.92	626.37	0.00287	6.47E-05	0.00287	0.0068
2016/10/3	611.85	614.84	605.41	624.38	0.00303	6.19E-05	0.00303	-0.0059
2016/10/4	609.62	613.32	606.09	620.61	0.00240	3.65E-05	0.00240	-0.0049
2016/10/5	607.18	611.03	603.79	618.33	0.00231	3.69E-05	0.00231	-0.0061
2016/10/6	612.08	608.18	600.57	615.87	0.00165	4.12E-05	0.00165	-0.0063
2016/10/7	617.21	612.70	605.65	619.81	0.00101	3.47E-05	0.00101	0.0064
2016/10/8	614.74	618.04	610.77	625.37	0.00135	3.63E-05	0.00135	0.0073
2016/10/9	615.65	615.48	608.31	622.71	0.00120	3.56E-05	0.00120	-0.0054
2016/10/10	617.54	616.12	609.72	622.56	0.00076	2.82E-05	0.00076	0.0003
2016/10/11	614.77	618.07	612.36	623.82	0.00086	2.24E-05	0.00086	0.0023
2016/10/12	635.01	615.47	602.02	629.15	0.00114	0.000126	0.00114	-0.0054
2016/10/13	635.96	636.93	623.01	651.07	0.00302	0.000126	0.00302	0.0308
2016/10/14	634.02	638.78	624.84	652.94	0.00443	0.000126	0.00442	-0.0015
2016/10/15	637.94	637.16	623.14	651.41	0.00495	0.000128	0.00494	-0.0075
2016/10/16	641.42	641.17	627.25	655.31	0.00506	0.000125	0.00504	0.0012
2016/10/17	638.97	644.88	631.19	658.78	0.00539	0.000119	0.00538	0.0004
2016/10/18	636.29	642.49	628.84	656.35	0.00550	0.000119	0.00549	-0.0092
2016/10/19	629.25	639.27	625.03	653.75	0.00468	0.000131	0.00467	-0.0097
2016/10/20	627.72	631.10	617.05	645.37	0.00293	0.000131	0.00293	-0.0159
2016/10/21	631.92	628.34	614.23	642.69	0.00098	0.000133	0.00098	-0.0054
2016/10/22	655.48	631.22	615.60	647.14	-0.00110	0.000163	-0.00110	0.0057
2016/10/23	653.25	655.94	639.65	672.53	0.00070	0.000163	0.00070	0.0370
2016/10/24	651.39	654.57	638.32	671.12	0.00202	0.000163	0.00202	-0.0041
2016/10/25	655.31	653.12	636.92	669.62	0.00265	0.000163	0.00265	-0.0049
2016/10/26	651.45	657.72	641.38	674.37	0.00368	0.000164	0.00368	0.0033
2016/10/27	682.22	654.33	629.84	679.53	0.00442	0.000375	0.00441	-0.0096
2016/10/28	687.68	687.20	661.32	713.82	0.00730	0.00038	0.00727	0.0409
2016/10/29	685.91	694.23	668.48	720.70	0.00952	0.000368	0.00947	0.0007
2016/10/30	698.00	692.69	665.99	720.17	0.00988	0.000398	0.00983	-0.0121
2016/10/31	702.00	704.96	677.83	732.89	0.00997	0.000397	0.00992	0.0076
2016/11/1	697.01	708.67	686.20	731.68	0.00950	0.000268	0.00946	-0.0042
2016/11/2	733.33	703.87	672.79	736.00	0.00984	0.000525	0.00980	-0.0167
2016/11/3	686.17	742.17	697.98	788.40	0.01206	0.000966	0.01199	0.0402