Problem Chosen	2022 MCM Summary Sheet	Team Control Number
C		2217069

Abstract

With the continuous prosperity of the financial market, the rapid development of computer technology, and the participation of a variety of users, the market has put forward higher requirements for futures and spot trading. The application of computer technology to trading strategy has gradually become an innovative and mainstream way of trading. So, the quantitative investment comes into being and is constantly improved in use, which relies on mathematical models and trading rules to generate trading signals. By inputting market information into the program, quantitative investors can make decisions about investment targets and investment timing.

In this paper, with only the daily price data of Bitcoin and gold from 2016 to 2021, we integrate and improve the ARIMA algorithm, the Stochastic Oscillator, and the turtle trading principle. By giving consideration to risk control, higher investment return, and capital utilization rate at the same time, we provide investors with an absolutely dominant quantitative trading strategy with high yield and stability.

The results show that our improved integrated trading strategy model is more efficient and stable than the traditional single strategy model. After the optimization of multiple parameters, it has demonstrated strong profitability and risk control ability. It has the ability to provide investors with reliable trading strategies.

Keywords: ARIMA; Stochastic Oscillator; Improved Turtle Trading Principle; Trading Strategy

Contents

1 Introduction

1.1 Information and Notations About Bitcoin and Gold

Gold and bitcoin are two investment products. Gold has been used as a major currency for thousands of years. Gold's monetary function in modern financial markets declined, while its investment function developed rapidly. Bitcoin is a cryptocurrency which has become popular since about 2011. Its price fluctuations make it a good investment product.

In order to describe this research more briefly, the following notations are used.

Table 1: Notations of Gold and Bitcoin

Variable	Notation
Price of Gold (per troy ounce)	GP ($)
Bitcoin	BTC
Price of One Bitcoin	BTCP ($)

1.2 Analysis of Given Data and Basic Strategies to Invest

Table 2: Historical Mean and Variance of Prices of Gold and Bitcoin

	BTCP	GP
Mean (μ)	12,206.07	1,464.55
Variance (δ^2)	197,230,892.03	62,146.41

By computing the means and variances of BTCP and GP in Table 2, bitcoin is more volatile than gold.

Bitcoin prices can be very volatile and is likely to achieve high yields. In order to gain better returns, investors should focus on bitcoin more than gold when bitcoin hasn't seen a large drop. Large drops in the value of bitcoin are often accompanied by instability in global financial markets. Therefore the inventors can choose cash or gold as a safe haven. After predicting the price of bitcoin and gold, investors need to apply the following brief basic strategies.

- BTCP $\nearrow \Longrightarrow$ hold BTC.
- BTCP $\searrow$, GP $\nearrow \Longrightarrow$ sell BTC, hold gold.
- BTCP $\searrow$, GP does not increase greatly $\Longrightarrow$ sell BTC, hold cash.

2 Model 1: Predict GP and BTCP for Next Day Based on Previous Days' Price Data

2.1 Application Objective of Model 1

Accurately predict the direction of GP and BTCP for the next day based on the price data of the previous days. The data obtained by the prediction algorithm can be compared with the actual price data. The closer the two data curves are, the more suitable the algorithm is. The data generated by the precise algorithm can serve Model 2 using the predicted growth to make decisions.

2.2 Construction of Algorithm Applied to Model 1 — ARIMA

Time series provide opportunity for forecasting value of future. Based on the old time series, we can predict the changing trend of BTCP and GP.

The continuity of gold and BTC data in time can be regarded as a time series model. For the time series model, we hope that some characteristics of the time series can continue over time in the future. Therefore, we expect the time series data to be (weak) stationary in a certain period, that is, its expectation and covariance remain unchanged to some extent.

When using ADF test, we can find that gold and bitcoin data hardly have stationarity locally. Therefore, we use the difference method to obtain the d-order difference of the original local data, so that the data can show the characteristics of stationarity continuing its data.

We hope to perform regression on the data to get its future properties. Since we only have time series data, we choose the autoregression model (AR) to perform regression with p continuous time intervals:

$$y_t = \mu + \sum_{i=1}^{p} \gamma_i y_{t-i} + \varepsilon_t$$

We believe that there exists a model which can reduce the influence of $_t$ on y_t, that is, it is offset by the errors in q continuous time intervals, that is, the moving average model (MA):

$$y_t = \mu + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t$$

To combine difference, autoregressive and moving average, we choose the AutoRegressive Integrated Moving Average (ARIMA) model.

In the process of selecting order (p, d, q), we choose AIC (Akaike information criterion) and BIC (Bayes information criterion) as standards:

$$\text{AIC} = 2k - 2\log(L)$$
$$\text{BIC} = k\log(n) - 2\log(L)$$

L is the maximum likelihood function, replaced by $\hat{\delta} = \frac{RSS}{n-k-1}$.

In order to ensure that the time series can continue the previous trend after difference, white noise test (model residual test) is performed on it, and then ARIMA prediction is performed.

In BTC's and gold's ARIMA model, the first 50 days are out of prediction. The reason is that without enough previous price data, the prediction's accuracy cannot be guaranteed.

2.3 Results of ARIMA

2.3.1 Analysis of BTC

After applying ARIMA to BTC, the data for predicted BTCP is outputted. In order to directly observe the accuracy of the predicted data, we draw the predicted and original BTCP on the same graph and compare them.

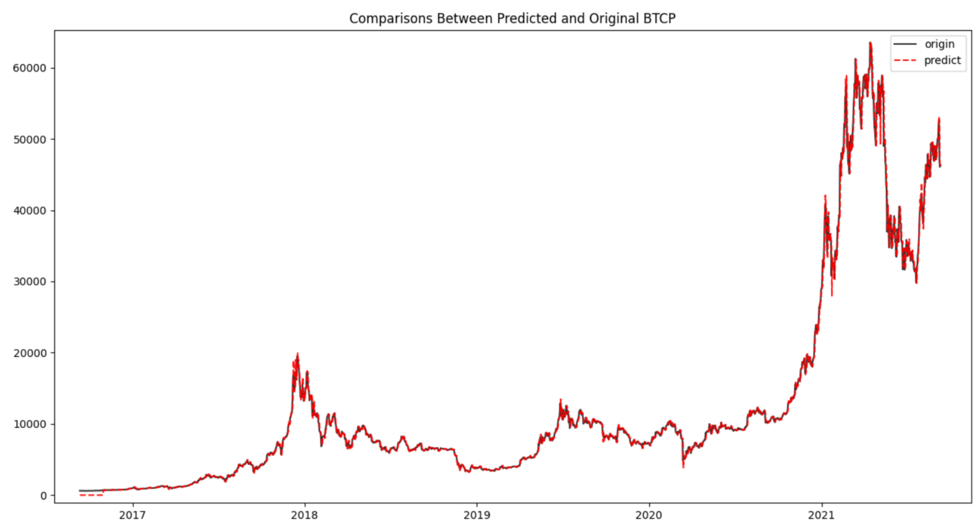


Figure 1: Comparisons Between Predicted and Original BTCP

In *Figure 1*, we believe that the data prediction under this algorithm has a high degree of accuracy. Its prediction accuracy is very high no matter when the data fluctuation is smooth (*Figure 2*) or the fluctuation is violent (*Figure 3*).

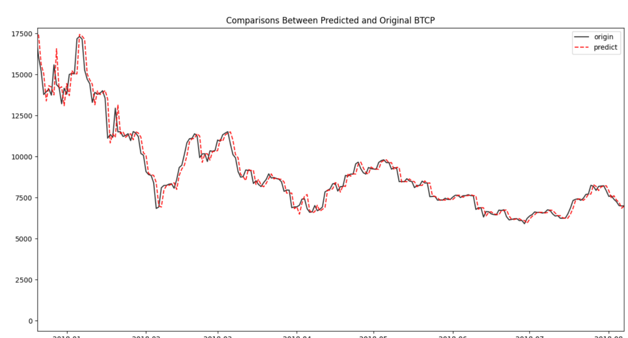


Figure 2: Smooth Example (Jan.–Aug., 2018)

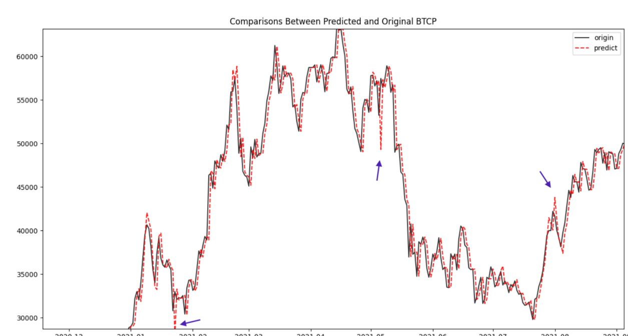


Figure 3: Violent Example (Jan.–Sept., 2021)

One flaw: when the data are extremely volatile, the predictions are sometimes much more volatile (blue arrays in *Figure 3*). Although the overall tendency predicted by ARIMA is correct, this flaw may lead to over-investment or oversold and relatively serious economic losses.

2.3.2 Analysis of Gold

Similarly, after completing the prediction of GP, we observed the original and predicted GP in the same graph.

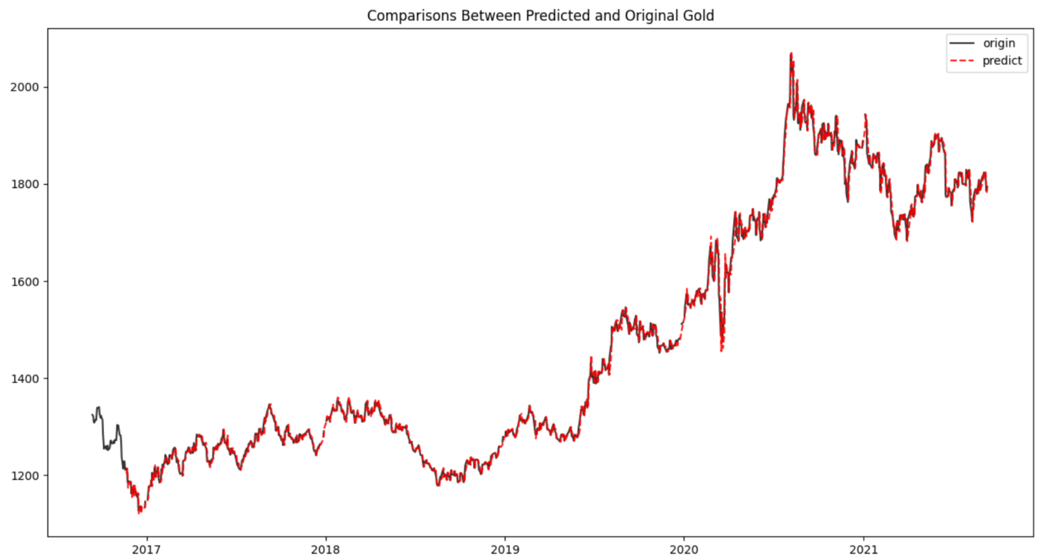


Figure 4: Comparisons Between Predicted and Original GP

Also, we focus on performance of ARIMA on smooth part and volatile part separately.

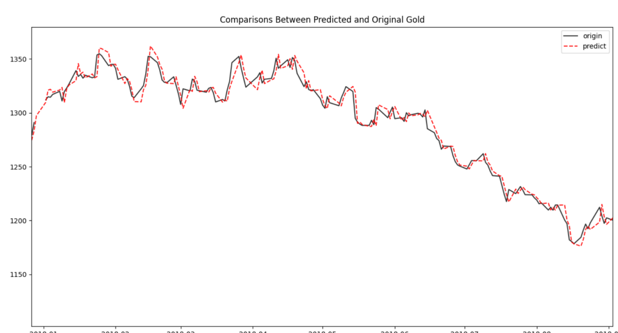


Figure 5: Smooth Example (Jan.–Sept., 2018)

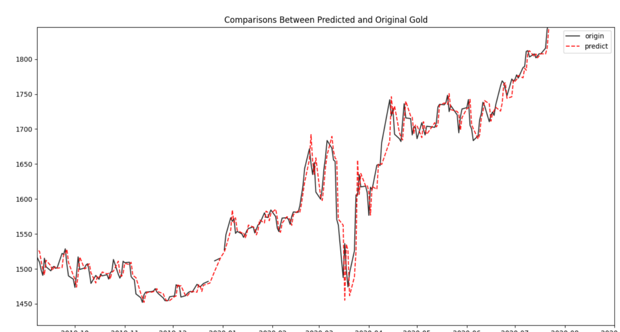


Figure 6: Violent Example (Jan.–Aug., 2020)

Because gold as a whole has a much smaller volatility than BTC, ARIMA estimates gold perfectly.

2.4 Improvement Idea of Model 1

The advantage of ARIMA in using time series for prediction is quite obvious. However, the downside is that there must be a lag of about a day. If time series are used, this lag is unavoidable.

Therefore, the improvement of Model 1 should focus on how to approach the accuracy of prediction brought by time series without using time series. Here's an approach worth trying. The prediction results are weighted by mean, variance and ROC, and then the new prediction results are used to buy and sell when a certain threshold value is reached. This idea will not be used for further prediction in this paper because the turtle trading principle in Model 2 will also have a lag time. Accuracy of forecast data will be more important in Model 1 than timeliness. This new idea could provide more timely predictions. It can work well in conjunction with other trading strategies.

3 Model 2: Decide the Investment Strategy Based on the Predicted GP and BTCP and Trend-following

3.1 Application Objective of Model 2

Based on the price data of the days before the trade, we can get the trend of price movement and use this to determine the trading strategy. After that, the size of the specific transaction volume is determined based on the results of Model 1. For example, when the BTC projected increase in Model 1 is greater than a certain threshold set, buy the BTC corresponding to that threshold price.

3.2 BTC Trading Strategy

3.2.1 Stochastic Oscillator

Stochastic Oscillator is a momentum indicator comparing a particular closing price of a security to a range of its prices over a certain period of time. The sensitivity of the oscillator to market movements is reducible by adjusting that time period or by taking a moving average of the result. It is used to generate overbought and oversold trading signals, utilizing a 0–100 bounded range of values. [1]

Stochastic Oscillator is mainly used in security market. It can also be applied to BTC, as BTC is also used as an investment product similar to a security.

The Raw Stochastic Value (RSV) of BTC in the cycle shall be determined first to calculate the KDJ index of bitcoin, and the cycle shall be set to 5 days in consideration of the large fluctuation of BTC. RSV of 5 days = $\frac{C5-L5}{H5-L5} \times 100$, and $C5$ in the formula is the price of the fifth day. L5 is 5-day low and H5 is 5-day high. Then calculate K and D: K of the current day $\frac{2}{3} \times K$ of the previous day $+\frac{1}{3}\times$ RSV of the current day, D of the current day = $\frac{2}{3} \times D$ of the previous day $+\frac{1}{3} \times K$ of the current day, if there is no K and D of the previous day, it is replaced by 50, $J = 3K$ of the current day $-2D$ of the current day.

The Stochastic Oscillator is sensitive and fast. It is suitable for short and medium term trend band analysis. Among these indicators, both K and D values are in the range of 0–100, while the J is not limited. However, in this situation, we set the study range to 0–100 with sensitivity $J > K > D$ and opposite security. Overbought and oversold signal: when the value of K line is higher than 90, D is higher than 90, and J is higher than 90 for three consecutive days, overbought appears and BTC has a high probability of callback in a short term. At this time, we should sell BTC to avoid risks. When K and D are lower than 40 and J are lower than 40 for three consecutive days, oversold appears, and BTC is weakened in the short term with a high probability of rebound. We can choose to buy.

From *Figure 7*, it makes sense that BTC has almost equal overbought days and oversold days in the first 200 days. This fits with BTC's relatively stable price trend over the first 200 days.

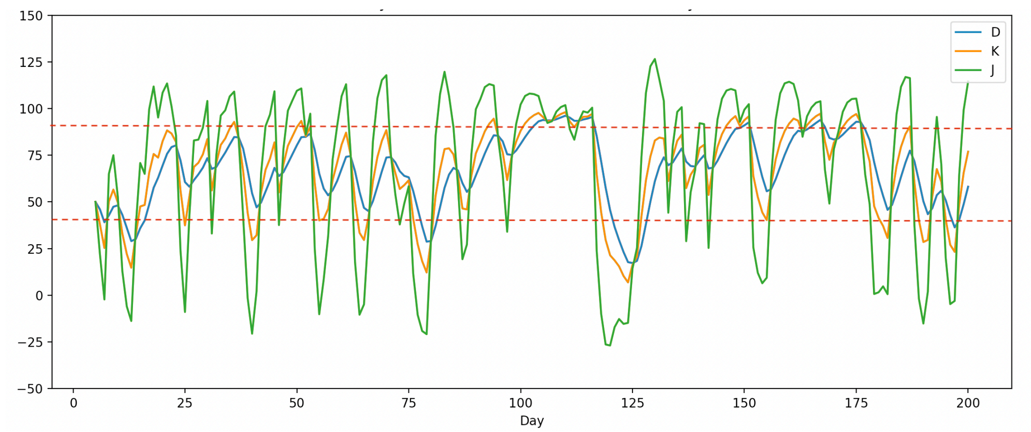


Figure 7: Stochastic Oscillator Curve of BTCP in the first 200 days

3.2.2 Turtle Trading Principle

So far, the next day price prediction is obtained using the ARIMA model, and the buy and sell signal is obtained using the Stochastic Oscillator. For specific trading strategies, such as position building and addition methods, we refer to the Turtle principle. ATR (Average True Range) was used to add or subtract positions in batches, and stop-profit and stop-loss were carried out dynamically. Since we did not know the daily price fluctuation at this time, THE ATR was extended to the mean of volatility within five days.

Turtle trading strategy is also a trend trading strategy, the significant characteristic is to capture the mid - and long-term trend, the whole market on the resonance of the market, can obtain greater profits in the short term. The criteria of turtle trading strategy in market selection are good liquidity and long/short mechanism, which can be basically adapted to most futures markets. Easy to deal due to the smaller market swings, which produce high transaction costs because of frequent trading, therefore, generally choose high liquidity large markets as the turtle trading strategy of trading places, but here the two market, BTC and gold, conform to the characteristics, so the turtle trading rules shall apply.

Turtle trading rules define a set of strict position control, stop profit and loss rules. Money management involves the position size setting, position adjustment, account growth expectations and the market transaction, etc., under the circumstance of deviating from the expected turtle trading money management is to take positions based on position size, the total available funds are divided into several small, initial position position must be controlled within a certain percentage of available funds. The most important indicator is volatility N, also known as Average True Range (ATR), which is the Average amplitude of the maximum volatility of the index in a single day. It is measured by the intraday points of the market and the history. The actual amplitude of the index points in a single day is the maximum amplitude, so as to ensure that the risk of capital is kept within the absolute control Range.

However, in this case, we don't have one-day price changes, so we use five-day bitcoin price changes to measure the amount of volatility, so $TR = Max(H5 - L5,\ H5 - C5,\ C5 - L5)$, $N = \frac{(19N')+TR}{20}$. The N value is at the heart of the turtle's trading rules and is associated with the size of each open position.

First of all, mentioned in some super, we consider selling BTC and buying BTC, combining with the principle of sea turtles. When the market is oversold, predict the market price of the breakthrough in the direction of profit 0.5N, accumulating 25%, stop-loss and accumulating using N. When the market is overbought, predict the market price of the breakthrough towards loss of N, underweight 25%, Each time a position is added, the stop loss level rises 0.5N.

Pseudo-code about trading strategy in BTC transactions.
This part is going to describe how to determine transaction volume of BTC.

```
def isOB(date):  this date is a overbought day.
def isOS(date):  this date is a oversold day.
loss_limit = 2 * N
for date = 6 to 1827:  # the predicted date
    # Ct = BTCP (t days before the predicted date)
    C = [C1,C2,C3,C4,C5]
    H5 = max(C), L5 = min(C)
    TR = max(H5-L5,H5-C1,C1-L5)
    if date==6:
        N=TR
    else:
        N=((19*N)+ TR)/20
    #N is the unit gained by previous 5 days' price data
    M = the corresponding unit price
    if predicted price-today's price >= 0.5*N and isOS(date) = True:
         Times = (predicted price - today's price)/(0.5*N)
         Purchase amount = M *Times
         if loss_limit - 0.5*N < 0.5*N:
              loss_limit = 0.5*N
         else:
             loss_limit = loss_limit - 0.5*N
    elif today's price - predicted price >= 2*N and isOB(date) = True:
         Times = (today's price - predicted price)/(2*N)
         Selling amount = M *Times
         if loss_limit + 0.5*N > 2*N:
              loss_limit = 2*N
         else:
             loss_limit = loss_limit + 0.5*N
    else:
         continue
         # No operation is needed.  Directly go to next day.
```

3.3 Gold Trading Strategy

In the Stochastic Oscillator plate, we change the cycle to 15 days, with the 15-day $RSV = (C15 - L15)/(H15 - L15) * 100$. In the formula, $C15$ is the 15th day price, $L15$ is the 15–day low, and $H15$ is the 15–day high.

In the last part, we prioritized the trading strategy of BTC, and when the holding of BTC was low, we considered buying gold to form a portfolio.

Buying gold requires both low BTC holdings and a potential upside based on our price forecasts and trend. Here we continue to use the Stochastic Oscillator and Turtle trading principles as the criteria for buying signals. However, considering the low volatility of gold and the relatively stable market, there are changes in the cycle and coefficient of index calculation. When changing the period and coefficient, we determine the best measurement index and strategy by analyzing the historical data and the oscillation of the period and coefficient within the range.

In the Stochastic Oscillator plate, we change the cycle to 15 days, with the 15-day $RSV = \frac{C15-L15}{H15-L15} \times 100$. The formula for K, D, and J is the same as for BTC.

Overbought and oversold signal: when the value of K line is higher than 80, D is higher than 80, and J is higher than 80 for three consecutive days, overbought appears and gold has a high probability of callback in a short term. At this time, we should sell gold to avoid risks. When K and D are lower than 20 and J are lower than 20 for three consecutive days, oversold appears, and gold is weakened in the short term with a high probability of rebound.

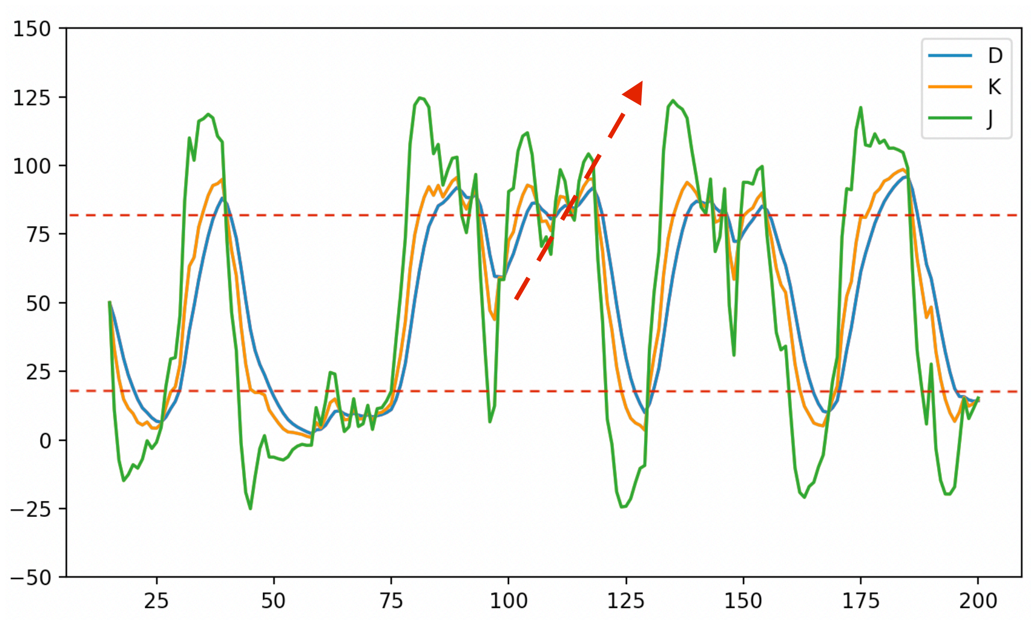


Figure 8: Stochastic Oscillator Curve of GP in the first 200 days

From *Figure 8*, gold's overbought and over sold days in the first 200 days are significantly more than BTC's. This fits with the fact that gold is of greater stability than BTC.

Using the turtle trading principle, again without a one-day price curve, we use a 15-day gold price change to measure volatility and change the total window to 30 days. Therefore:

$$TR = max(H15 - L15,\ H15 - C15,\ C15 - L15)$$
$$N = \frac{29N' + TR}{30}$$

N' is the previous day's N. The 15th day's N equals TR.

4 The Results

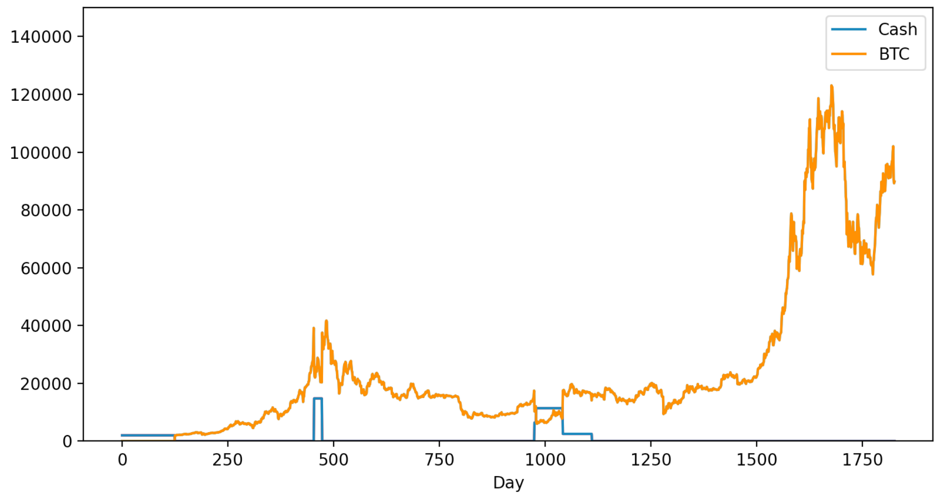


Figure 9: Results Only Considering Cash and BTC

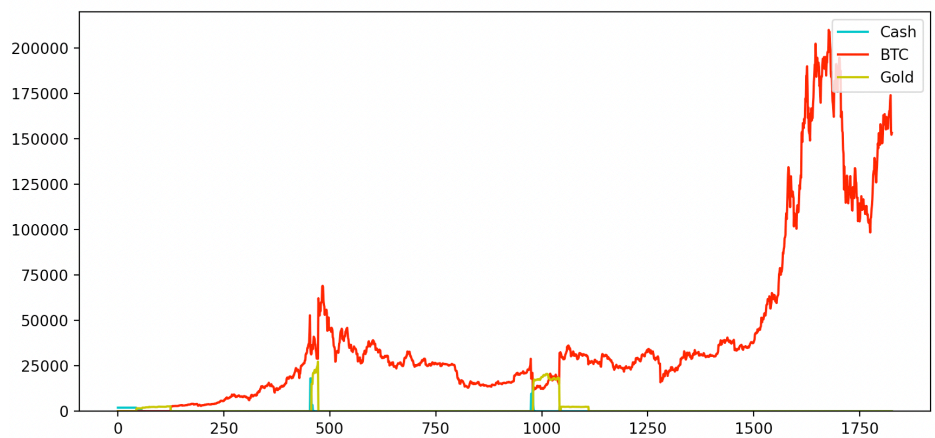


Figure 10: Complete Results

Considering only BTC and cash transactions, this model can obtain assets with a final value of 89,809.80, achieving an annual percentage rate (APR) of 145.847% (*Figure 9*).

After gold's addition, final transaction result increased to 153,295.97 with an APR of approximately 173.59% (*Figure 10*).

5 Sensitivity Analysis

5.1 Sensitivity Analysis to α_{BTC}

When changing α_{BTC}, we can be more profitable if we change the investment strategy. For example, if the transaction cost for BTC increases, we can increase the overbought value and decrease the oversold value to some degree. This will reduce total transaction fees by decreasing transaction times. The detailed results are in the following table:

Table 3: Sensitivity Analysis to α_{BTC} (α_{gold} = 1%)

α_{BTC}	Results(No Strategy Changing)	Results(With Strategy Changing)	New Critical Value
1%	162,182.08	163,057.41	86 44
2%	153,295.97	153,295.97	90 40
3%	139,549.64	141,633.46	91.5 38.5
4%	129,117.31	133,451.40	92.5 37.5

5.2 Sensitivity Analysis to α_{gold}

Changing strategy also lead to more profit when α_{gold} is changing. The detailed results are in the following table:

Table 4: Sensitivity Analysis to α_{gold} (α_{BTC} = 2%)

α_{gold}	Results(No Strategy Changing)	Results(With Strategy Changing)	New Critical Value
0.5%	157,923.66	159,149.25	75 25
1.0%	153,295.97	153,295.97	80 20
1.5%	149,687.85	151,558.46	82 18
2.0%	145,196.58	147,376.48	85 15

6 Strengths and Weaknesses

6.1 Strengths

In our model, we combine the ARIMA algorithm, Stochastic Oscillator, and turtle trading principles to develop trading strategies, determine portfolios, refine existing rules with data types, and utilize BTC and gold daily price data from 2016 to 2021 to analyze and backtest to optimize the trading model.

Finally, the five-year APR=173.59% was achieved. We can demonstrate that our model is optimal in many ways: it controls the trading risk while ensuring the maximum return rate.

Firstly, we analyze the market characteristics of BTC and gold, and determine BTC as the first consideration to increase the yield while Gold as a tool to cover positions to reduce the main direction of risk. This makes up for the lack of other market information to price portfolios through CAMP, MPT (Modern Portfolio Theorem) and other methods. At the same time, it also simplifies our algorithm and model, making the trading strategy more concise and controllable.

Second, we use ARIMA algorithm to predict the daily price, and select 50 days as a moving window to achieve better fitting effect through comparison and analysis of the accuracy of prediction achieved by different window lengths. From the final fitting results, the fitting degree is very high, which provides a good basis for judging our trading signals and effectively reduces risks.

Third, in order to make the Stochastic Oscillator, which is more commonly used in the stock market, more consistent with the bitcoin and gold markets, the RSV cycle and the critical value of the overbought and oversold interval are oscillated to find the best criterion. After comparing, the cycle of bitcoin is located at 5. The period of gold is set at 15. The overbought and oversold critical values are 90/40 and 80/20, which make our model fit the actual situation and have a high reliability. From another side, the oversold and overbought judgment as a necessary condition for buying and selling, effectively avoid too many transactions to increase the transaction cost is too high and reduce the phenomenon of yield.

Four, we through the use of the turtle trading model to implement strict position control, and the two changes of innovation, one is to use the cycle index of maximum index fluctuation amplitude instead of the average days maximum average amplitude fluctuations, in does not affect the accuracy of instruction cases solved the problem of the intraday data missing. The other is, after obtaining the unit N for opening, adding and reducing positions, the next day's price predicted by ARIMA algorithm is used to replace today's price to determine the final trading volume, so that the whole model is compact and has better prediction effect, resulting in achieving a better return rate than the original turtle strategy.

To sum up, this trading strategy model has the advantage of high return and low risk compared with most trading strategies.

6.2 Weaknesses

Our model uses three approaches: ARIMA, Stochastic Oscillator, and Turtle Trading rule to develop portfolio and trading strategies for BTC and gold. While integrating the benefits brought by these methods, we need to admit that their limitations and the imperfections of the comprehensive method make our model have some weaknesses.

First of all, when using ARIMA method, we use the data of 50 days as a window to predict the price on the fifty-first day, which makes it impossible for us to build positions and control positions in the first fifty days. In addition, there is a lag phenomenon in price prediction, and the price fluctuation cannot be captured timely enough. As a result, buy and sell orders may be delayed 1–2 days from the best time, reducing the potential gains.

Second, although the Oscillator is a better technical indicator, there are two problems with its application: The destruction of the Oscillator by large oscillations (BTC does not have the stock mechanism limit, so it is significantly affected) and the passivation of the Oscillator (when the Oscillator response is too sensitive, It will lead investors to purchase too early or ship too early, and passivation is more obvious in short-term analysis, which affects the accuracy of BTC strategy).

Third, there are two obvious problems with the turtle strategy: excessive retracement in the forecast, and recovery (the recovery from the rally is likely to be wiped out by the subsequent crash). As a result, the risk coefficient of our model is not low enough and the yield rate is difficult to achieve the best effect.

References

[1] Adam Hayes. (June 25, 2021) Stochastic Oscillator. In *Investopedia*. Retrieved February 20, 2022 from https://www.investopedia.com/terms/s/stochasticoscillator.asp.

First Appendix: Important Codes

Model 1: ARIMA

```
import pandas as pd
import numpy as np
import datetime
import statsmodels.api as sm
import matplotlib.pyplot as plt
from statsmodels.tsa.arima.model import ARIMA
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.stats.diagnostic import acorr_ljungbox
from statsmodels.tsa.stattools import adfuller
import itertools

class myArima:
    def __init__(self, df):
        self.timeStart = pd.Timestamp('2016-09-11') ## start pt of time series
            windows
        self.timeAdd = pd.Timedelta('1 day')
        self.window = 50 ## windows size
        self.timeEnd = self.timeStart + self.timeAdd * (self.window-1) ## end pt
        self.df = df ## total time series data
        self.diff = pd.DataFrame([]) ## data after diff
        self.data = df.loc[self.timeStart: self.timeEnd] ## slide the data
        self.order = (0, 0, 0) ## ARIMA's order
        self.predict = [] ## result

    def dataChange(self): ## slide the data
        self.timeEnd = self.timeStart + self.timeAdd * (self.window-1)
        self.data = self.df.loc[self.timeStart: self.timeEnd]

    def ADF_test(self): ## test the stability
```

```
            timeseries = self.diff
            x = np.array(timeseries.iloc[:, 0])
            adftest = adfuller(x, autolag='AIC')
            if adftest[0] < adftest[4]["1%"] and adftest[1] < 10 ** (-5):
                print("stable")
                return True
            else:
                print("unstable")
                return False

        def random_test(self): ## test white noise
            timeseries = self.diff
            p_value = acorr_ljungbox(timeseries, lags=1)
            if p_value['lb_stat'][1] < 0.05:
                print("Nonrandom sequence")
                return True
            else:
                print("White noise sequence")
                return False

        def stationarity(self): ## diff
            self.diff = self.data.diff(1).dropna()
            # print(self.data)
            # print(self.diff)
            self.order = (0, 1, 0)
            if not self.ADF_test():
                self.diff = self.diff.diff(1).dropna()
                self.order = (0, 2, 0)

        def detetminante_order_BIC(self): # AICBICHQIC
            self.stationarity()
            # self.random_test()
            timeseries = self.diff
            p_min = 0
            q_min = 0
            d_min = 0
            p_max = 4
            q_max = 4
            d_max = 0
            results_bic = pd.DataFrame(index=['{}'.format(i) for i in range(p_min,
               p_max + 1)], ## AR
            columns=['{}'.format(i) for i in range(q_min, q_max + 1)]) ## MA

            for p, d, q in itertools.product(range(p_min, p_max + 1), ## search
                                              range(d_min, d_max + 1),
                                              range(q_min, q_max + 1)):
                if p == 0 and d == 0 and q == 0:
```

```
                    results_bic.loc['{}'.format(p), '{}'.format(q)] = np.inf
                    continue
                try: ## find the best
                    model = sm.tsa.SARIMAX(timeseries, order=(p, d, q),
                       freq=self.timeAdd)
                    results = model.fit()
                    results_bic.loc['{}'.format(p), '{}'.format(q)] = results.bic
                except:
                    continue
            results_bic = results_bic[results_bic.columns].astype(float)
            point = results_bic.stack().idxmin()
            p = int(point[0])
            q = int(point[1])
            d = self.order[1]
            self.order = (p, d, q)

    def ARIMA_model(self): ## predict data one day after
        self.detetminante_order_BIC()
        train_data = self.data
        arima_model = ARIMA(train_data, order=self.order)
        result = arima_model.fit()
        pred = result.predict(start=self.timeEnd, end=self.timeEnd+self.timeAdd)
        dfPred = pd.DataFrame(pred)
        print(dfPred)
        results = dfPred.iloc[-1][0]
        print(results)
        return results

    def mypredict(self): ## form the predicted data
        for i in range(50):
            self.predict.append(np.nan)
        for i in range(len(self.df)-49):
            temp = self.ARIMA_model()
            self.predict.append(temp)
            self.timeStart += self.timeAdd
            self.dataChange()
        print(self.predict)
        return pd.DataFrame(self.predict, columns=['Value'])
```

Model 2: Strategy

```
import pandas as pd
import matplotlib.pyplot as plt
import openpyxl
import math
```

```
df = pd.read_excel('BTCP.xlsx')
dfl=df.values.tolist()
alpha=0.02 #2% trading fee
beta=0.01
date=5

df2 = pd.read_excel('predict_bc.xlsx')
dfl2=df2.values.tolist()

df3 = pd.read_excel('BTCP.xlsx')
dfl3=df3.values.tolist()

pf={'Cash':2000,'BTC':0,'Gold':0}

def detective(action, pf, alpha, times): ## scan the situation of my wallet and act
    'buy' or 'sale'
    temp = pf
    total=(temp['Cash']+temp['BTC']+temp['Gold'])*0.25*times
    if action == 'buy':
        need = total + total*alpha
        if temp['Cash'] >= need:
            temp['Cash'] -= need
            temp['BTC'] += total
        else:
            temp['BTC'] += temp['Cash']/(1+alpha)
            temp['Cash'] = 0

    else:
        need = total + total*alpha
        if temp['BTC'] >= need:
            temp['BTC'] -= need
            temp['Cash'] += total
        else:
            temp['Cash'] += temp['BTC']/(1+alpha)
            temp['BTC'] = 0
    return temp

def goldop(action, pf, beta):
    temp=pf
    total=total=temp['Cash']*0.8
    if action == 'buy':
        need=total+total*beta
        temp['Cash'] -= need
        temp['Gold'] += total
    else:
        temp['Cash'] = temp['Gold']/(1+beta)
        temp['Gold'] = 0
```

```
def btcchange(pf, C5, C4): #BTC change price every day
    temp=pf
    temp['BTC'] = temp['BTC']*C5/C4
    return temp

def gchange(pf,a,b):
    temp=pf
    temp['Gold'] = temp['Gold']*a/b
    return temp

def isOB(K,D,Jlist): ## sale
    if K>=80 and D>=80 and Jlist[-1]>=80:
        return True
    else:
        return False

def isOS(K,D,Jlist): ## purcheas
    if K<=40 and D<=40 and max(Jlist[-3:-1])<=40:
        return True
    else:
        return False

while date<=1826:
    C1=dfl[date-5][0] #BTCP 4 days ago
    C2=dfl[date-4][0] #BTCP 3 days ago
    C3=dfl[date-3][0] #BTCP 2 days ago
    C4=dfl[date-2][0] #BTCP yesterday
    C5=dfl[date-1][0] #BTCP today
    print(date)
    pf=btcchange(pf, C5, C4)
    if date%7!=0 and 6:
        pf=gchange(pf,dfl3[math.floor(date*5/7)][0],dfl3[math.floor(date*5/7-1)][0])
    L5=min(C1,C2,C3,C4,C5)
    H5=max(C1,C2,C3,C4,C5)
    RSV=(C5-L5)/(H5-L5)*100

    if date==5:
        K=50
        D=50
    else:
        K=(2/3)*K+(1/3)*RSV
        D=(2/3)*D+(1/3)*K

    J=3*K-2*D
    TR=max(H5-L5,H5-C5,C5-L5)
```

```
if date == 5:
    N=TR
    loss_limit = 2 * N
else:
    N=(19*N+ TR)/20
if date>=51:
    if isOB(K,D,Jlist):

        if dfl2[date-1][0]-C5>=1.1*N:
            m=m+1
            times=(dfl2[date-1][0]-C5)/(1.1*N)
            pf = detective(action='sell', pf=pf, alpha=alpha, times=times)
            if loss_limit - 0.5*N < 0.5*N:
                loss_limit = 0.5*N
            else:
                loss_limit -= 0.5*N
    if isOS(K, D, Jlist):
        if C5-dfl2[date-1][0] >= 0.43*N:
            m=m+1
            times=(C5-dfl2[date-1][0])/(0.43*N)
            goldop('sell',pf,beta)
            pf = detective(action='buy', pf=pf, alpha=alpha, times=times)
            if loss_limit + 0.5*N > 2.0*N:
                loss_limit = 2.0*N
            else:
                loss_limit += 0.5*N
date=date+1

if pf['Cash']>0:
    if dfl3[math.floor(date*5/7)][0]-dfl3[math.floor(date*5/7-1)][0]>20:
        goldop('buy', pf, beta)
print(pf)
```

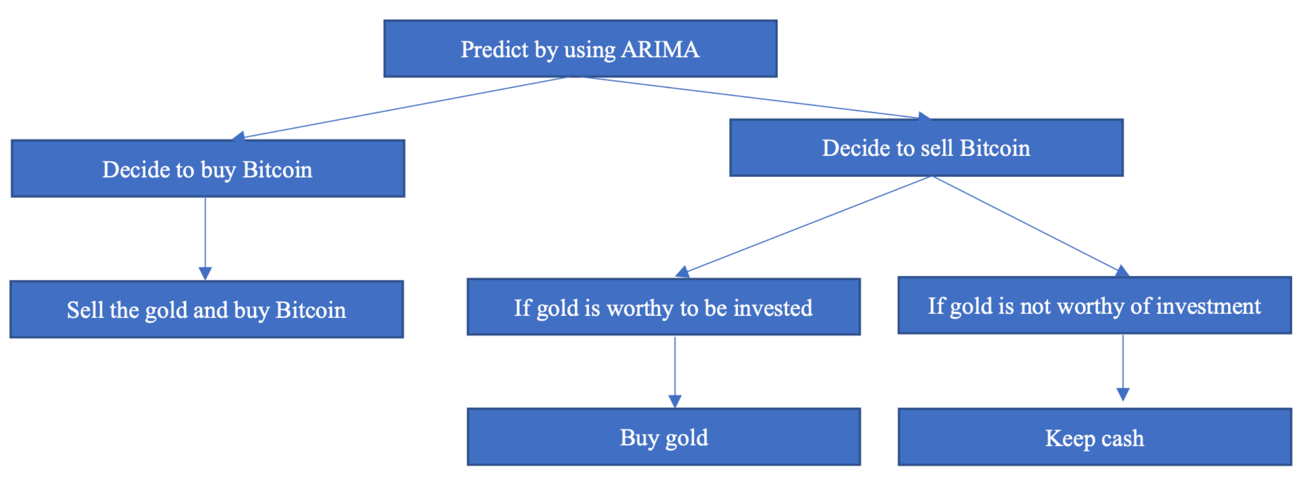
Second Appendix: A Memorandum Including Our Strategy, Model, and Results

Strategy

The core of the strategy is divided into two steps: first, prediction is performed using the ARIMA algorithm, and then the Stochastic Oscillator and turtle trading principle are used to determine whether and how much to change the current asset structure.

First, we conduct ADF tests on the prices of gold and bitcoin. When using ADF test, we can find that gold and bitcoin data hardly have stationarity locally. Therefore, we use the difference method to obtain the d-order difference of the original local data, so that the data can show the characteristics of stationarity continuing its data. Therefore, combining autoregression model (AR) and moving average model (MA), we finally choose to use ARIMA model to achieve the prediction.

Second, because bitcoin is more volatile than gold, it can yield higher returns if invested properly. Therefore, in the pursuit of higher yield, we adopt the strategy of investing mainly in bitcoin and supplemented by gold, and give priority to price prediction and trend tracking of bitcoin to obtain buy and sell orders. When the holding of bitcoin is small, we judge whether to buy and sell gold through price prediction and trend tracking of gold.



Model

Modeling should also be viewed from two perspectives.

As for ARIMA model, it involves d order difference of original local data,p continuous time interval data is selected for regression (autoregression) and offset with the errors in q continuous time interval (moving average), so three key orders d, p and q are involved. In the process of selecting order (p, d, q), we choose AIC (Akaike information criterion) and BIC (Bayes information criterion) as standards:

$$\mathrm{AIC} = 2k - 2\log(L)$$

$$\mathrm{BIC} = k\log(n) - 2\log(L)$$

L is the maximum likelihood function, replaced by $\hat{\delta} = \frac{RSS}{n-k-1}$.

In order to ensure that the time series can continue the previous trend after difference, white noise test (model residual test) is performed on it, and then ARIMA prediction is performed.

The next step is to use the Stochastic Oscillator to determine whether there is a need to buy or sell. The Raw Stochastic Value (RSV) of Bitcoin in the cycle shall be determined first to calculate the KDJ index of bitcoin, and the cycle shall be set to 5 days in consideration of the large fluctuation of Bitcoin. RSV of 5 days = $\frac{C5-L5}{H5-L5} \times 100$, and $C5$ in the formula is the price of the fifth day. L5 is 5-day low and H5 is 5-day high. Then calculate K and D: K of the current day $\frac{2}{3} \times K$ of the previous day $+\frac{1}{3}\times$ RSV of the current day, D of the current day = $\frac{2}{3} \times D$ of the previous day $+\frac{1}{3} \times K$ of the current day, if there is no K and D of the previous day, it is replaced by 50, $J = 3K$ of the current day $-2D$ of the current day. Due to the low volatility of gold, we set the cycle for gold at 15 days, otherwise the same as bitcoin.

Overbought and oversold signal: when the value of K line is higher than 90, D is higher than 90, and J is higher than 90 for three consecutive days, overbought appears and Bitcoin has a high probability of callback in a short term. When K and D are lower than 40 and J are lower than 40 for three consecutive days, oversold appears, and Bitcoin is weakened in the short term with a high probability of rebound. For gold, we set 80 as overbought point and 20 as oversold point.

When overbought or oversold appear, we can use turtle trading principle to determine the amount we sell or buy.

Results

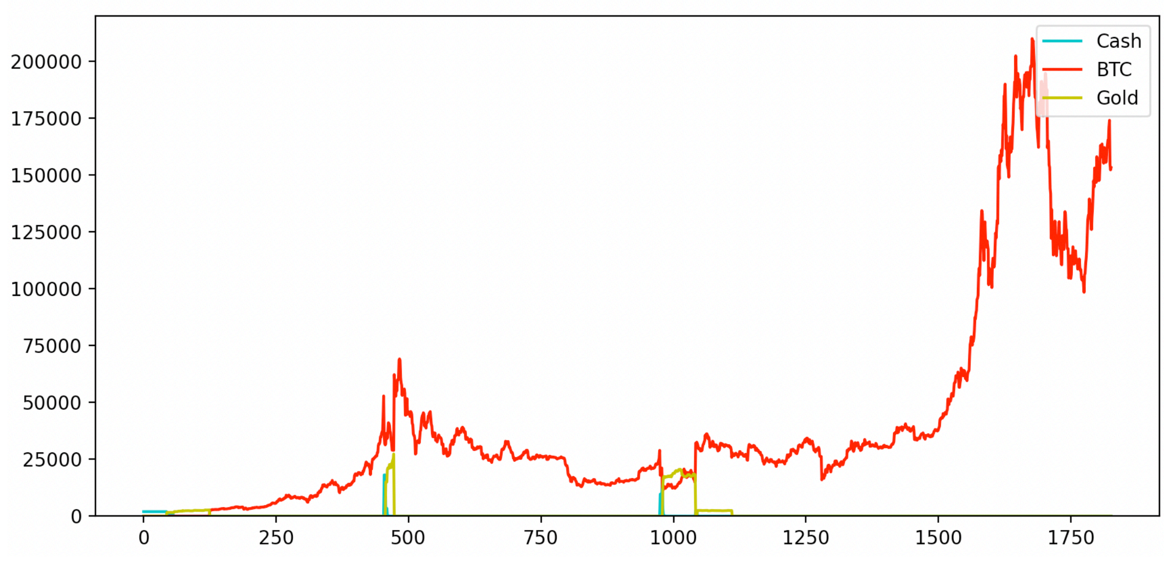


Figure 11: Complete Results

Complete transaction result increases to 153,295.97 with an APR of approximately 173.59%.