Supervised by
Guodong Chen
Offer Shapir

2023
Deans' Undergraduate
Research Fund
Summary Sheet

Drafted by
Peiyang Wu
Baichen Shi

Exploring the Trading Strategy Optimization of Bitcoin, Dogecoin, and Gold in the Cryptocurrency Market

Nowadays, financial activities have become an important part of people's lives. In trading, some people end up making profits but some people lose money. The variation in outcomes can be attributed not only to the choice of assets but also to the trading strategies employed. This paper primarily centers on the development and evaluation of trading strategies. To achieve this goal, we have devised two models: Model I - the Volatile Asset Price Forecasting Model, and Model II - the Strategy Improvement Model.

In Model I, the primary objective is to forecast asset prices with the utmost accuracy to form the foundation for formulating the trading strategy. The model follows a prudent approach of predicting one day ahead in each run. To achieve accurate forecasts, the ARIMA model is applied after testing and manipulating the stationary data. The hyperparameters of the ARIMA model are carefully set to emphasize recent price data while predicting future price movements. The results show a high level of accuracy, with an R-squared value of over 0.98 for all assets.

For the development of the trading strategy, the dynamic programming method is utilized to break down the five-year investment period into day-to-day investment decisions. Each day, the model performs constrained minimization of multivariate scalar functions using the Python package **scipy**. The implementation of the model yields a return of around 770% by the end of the 5-year investment period.

The objective function of the model considers the maximum next-day profit rate while also penalizing fluctuations. Through the analysis, it is evident that risk-preferred investors tend to earn more than risk-averse investors, given the significant fluctuations that cryptocurrencies can experience within a short timeframe.

Sensitivity Analysis is conducted to ensure the stability of the model. The results demonstrate promising ultimate returns with different initial budgets, while an increase in commission fees can reasonably reduce the return. Based on these findings, appropriate advice is provided accordingly.

By testing the trading strategy in early timeframes, it noticable that implementing the strategy when Bitcoin prices are lower can lead to significantly increased ultimate returns. For instance, if the model starts the investment before 2016, investors can earn over 8000% returns compared to their initial investment.

During the evaluation of various trading strategies, it becomes evident that focusing on larger cryptocurrencies like Bitcoin is advisable, as they exhibit promising growth with relatively lower volatility. Conversely, the value of smaller cryptos like Dogecoin can be highly volatile, which may experience dramatic increases but also sharp declines in a short span. It is very obvious when Trading Strategy only considers profit rate without implementing punishment of possible fluctuation, which without over-allocating in Dogecoin and lose a lot when its price drops.

Contents

1 Introduction

1.1 Background

Bitcoin and Dogecoin are both popular cryptocurrencies. Bitcoin is the first cryptocurrency created in 2009. It is designed to be a peer-to-peer electronic payment system that allows pay and receiving money without the need for a bank or other intermediaries. Dogecoin was created in 2013 as a joke based on the popular "Doge" meme. It is also considered a peer-to-peer cryptocurrency, just like Bitcoin.

However, both of them had a significant change since 2021, when a lot of institutions adopts Bitcoin and makes it more mainstream. Especially Elon Musk, when he announced in March 2021, that Tesla had bought $1.5 billion worth of Bitcoin and would accept it for paying for its product. Similarly, in January 2021, Mush send a Twitter about it, which led to a significant increase in it's

2 Assumptions and Justifications

I assume that the data used for our analysis is accurate and reliable.

I assume that the investor samples are independent, the action of one investor will not impact the behavior of other investors.

I assume that investors who will use this strategy are risk-seekers who are willing to take arbitrary risks for the sake of return.

I assume investors only hold cash, gold, Bitcoin, and Dogecoin. No more positions are allowed to be added to an asset while it is held.

For the predictions we give, I assumed that no accidental events that occur during the prediction period. The prediction ignores external incidents that may affect the cryptocurrency price.

3 Notations

The notations used in this paper are listed in Table 1.

Table 1: Notations

Symbol	Definition
x_i	Missing value at i-th day
nan	missing value, NaN in python
σ	3σ limits, used in normal distribution
$Arima(p,q,d)$	p, q, d implies LA, MA, and Difference order respectively
$[c_k, g_k b_k, d_k]$	The proportion of asset
$[c_k^p, g_k^p, b_k^p, d_k^p]$	The change of price of assets
$[\Delta c_k, \Delta g_k, \Delta b_k, \Delta d_k]$	Proportion change of asset results of transaction

4 Establishment and Result of the Model

4.1 Data acquirement and cleaning

To obtain the price of Bitcoin, Dogecoin, and Gold from Yahoo Finance. The data contains the open price, close price, high price, and low price, from 6/18/2018 to 6/18/2023, in a total of 5 years, 1827 days. Importantly, gold is traded only when the market is open, so there are 1260 data for gold. Here, I use the close value to analyze the data.

After obtaining the data, I performed data cleaning and preprocessing.

I gathered information on the open price, close price, high price, and low price from Yahoo Finance, from June 18, 2018, to June 18, 2023, which constitutes a total of 1827 days. However, it's important to note that Gold is only traded when the market is open, so in total there are only 1260 data points for Gold.

After obtaining the data, I then conduct data cleaning and preprocessing to ensure its quality and suitability for analysis. During this process, I observed the Dogecoin price data on June 17, 2023, is missing. Removing this null data directly could potentially disrupt the day-to-day consistency of the dataset.

To address the issue of data inconsistency, I applied a series-based Hermite incorporation method to fill in the missing value for Dogecoin on June 17, 2023. This method ensures the preservation of the overall consistency of the data, allowing for a more robust and accurate analysis of the price fluctuations of Dogecoin, Bitcoin, and Gold over the specified time frame.

4.1.1 Hermite interpolation

To fill the empty data with the empty entry, such that the curve goes smooth and derivable for all interpolated data. Which means:

$$\phi(x_i) = y_i \quad for(i = 1, 2, ..., n)$$

$$\frac{\partial \phi(x_i)}{\partial x_i} = \frac{y_i}{x_i} \quad for(i = 1, 2, ..., n)$$

Assume function f(x) has n+1 different valid points in [a, b], such that $a = x_0 < x_1 < x_2 < ... < x_n = b$ f(x) is defined in [a, b] and satisfies:

$$f(x_i) = y_i$$

$$f'(x_i) = y_i'$$

With a total of 2n+2 conditions, we can uniquely determine a polynomial to the max power of n+1 $H_{2n+1} = H(x)$ satisfies:

$$H(x_j) = y_j, H'(x_j) = m_j$$

With residuals:

$$R(x) = f(x) - H(x) = \frac{f^{2n+2}(\xi)}{(2n+2)!}\omega_{2n+2}(x)$$

Meanwhile, with the grow-up of function degree, we considered the runge phenomenon, which means the polynomial may oscillate wildly near the boundaries of the interval, even if the function being approximated is relatively smooth. So in this problem specifically, we limit the max degree to 3.

4.2 Data Visualization

Since the data is not strictly normally distributed, I can't imply 3σ criterion to check the outsiders. Instead, I use the error value to check the outsiders.

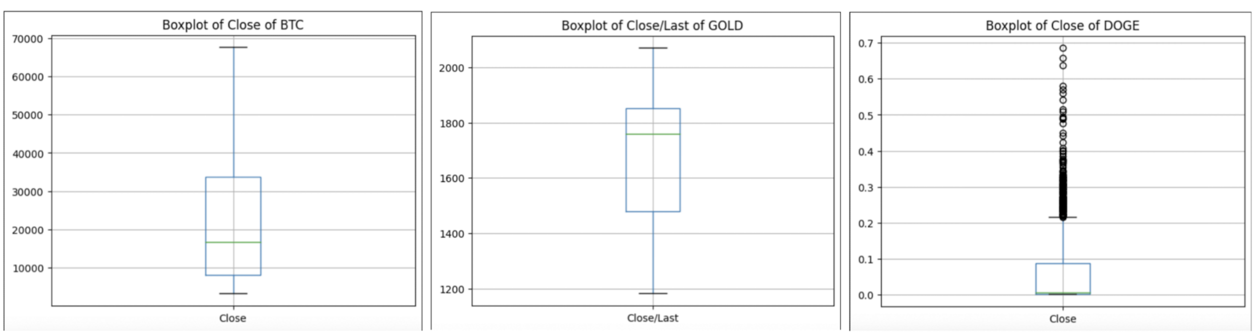

Figure 1: Box plot analysis of Bitcoin, Gold and Dogecoin

By plotting the box graph, it's noticeable that a part of the Dogecoin data of Bitcoin is larger than the upper bound, while most of the data in Bitcoin and Dogecoin is in the range. By counting the outsiders using the program, there are 7 outsiders for Bitcoin, 6 outsiders for Gold, and 185 outsiders for Dogecoin.

Although Dogecoin has many outsiders, this is because the price of Dogecoin is highly volatile and can change rapidly in a short period of time. So when analyzing the price of Dogecoin, I will consider separately analyzing the price after 2021.

4.3 Time Series Analysis

4.3.1 Stationary Test

To ensure the time-series model can be implemented, I first performed a stationary check by Augmented Dickey-Fuller (ADF) Test and Kwiatkowski, Phillips, Schmidt, and Shin (KPSS) Test.

ADF test is conducted with the following assumptions:

- H_0: Series is non-stationary, or series has a unit root.
- H_a: Series is stationary, or series has no unit root.

Conditions to reject H_0

- The Test statistic < Critical Value and p-value < 0.05 – Reject Null Hypothesis(HO), i.e., time series does not have a unit root, meaning it is stationary. It does not have a time-dependent structure.

Whereas the KPSS test is conducted with the following assumptions:

- H_0: Series is trend stationary or series has no unit root.
- H_a: Series is non-stationary, or series has a unit root.

Conditions to Fail to Reject H_0:

- If the Test Statistic < Critical Value and p-value < 0.05 – Fail to Reject Null Hypothesis(HO), i.e., time series does not have a unit root, meaning it is trend stationary.

And there are 4 possible outcomes:

- If the ADF model rejects the null hypothesis, and KPSS does not reject; Stationarity Exist.
- If ADF does not reject the Null Hypothesis while KPSS does reject; Unit Root Exists.
- If ADF and KPSS reject the null hypothesis; Heteroscedasticity is impacting the results.
- If ADF and KPSS do not reject the null hypothesis; Data does not have enough observations.

Using Python, I checked the ACF and KPSS of the three cryptocurrencies:

Results of Dickey-Fuller Test of Bitcoin :	
Test Statistic	-1.507725
p-value	0.529687
#Lags Used	24.000000
Number of Observations Used	1802.000000
Critical Value (1%)	-3.433984
Critical Value (5%)	-2.863145
Critical Value (10%)	-2.567625

Results of KPSS Test of Bitcoin :	
Test Statistic	3.229989
p-value	0.010000
Lags Used	27.000000
Critical Value (10%)	0.347000
Critical Value (5%)	0.463000
Critical Value (2.5%)	0.574000
Critical Value (1%)	0.739000

Figure 2: Bitcoin Stationary Test

Based on a significant level of 0.05, the ADF test cannot reject H_0 while KPSS rejects H_0. Therefore, the series is non-stationary. Similarly, I checked the rest 2 series, as follows:

Results of Dickey-Fuller Test of Dogecoin	
Test Statistic	-2.558259
p-value	0.101935
#Lags Used	25.000000
Number of Observations Used	1801.000000
Critical Value (1%)	-3.433986
Critical Value (5%)	-2.863146
Critical Value (10%)	-2.567625

Results of KPSS Test of Dogecoin :	
Test Statistic	2.337239
p-value	0.010000
Lags Used	27.000000
Critical Value (10%)	0.347000
Critical Value (5%)	0.463000
Critical Value (2.5%)	0.574000
Critical Value (1%)	0.739000

Figure 3: Dogecoin Stationary Test

Results of Dickey-Fuller Test of Gold :	
Test Statistic	-1.307137
p-value	0.625901
#Lags Used	7.000000
Number of Observations Used	1252.000000
Critical Value (1%)	-3.435584
Critical Value (5%)	-2.863851
Critical Value (10%)	-2.568001

Test Statistic	4.352109
p-value	0.010000
Lags Used	21.000000
Critical Value (10%)	0.347000
Critical Value (5%)	0.463000
Critical Value (2.5%)	0.574000
Critical Value (1%)	0.739000

Figure 4: Gold Stationary Test

It's noticeable that all the Cryptocurrencies are non-stationary. So I then check the 1 to 3 order of difference, then noticed that the 1st order difference makes the data smooth. It implies that I can use the ARIMA model to analyze

ADF Test of Gold							
Variable	Difference Order	t	P	AIC	Critical Value		
					1%	5%	10%
GOLD	0	-1.307	0.626	10565.171	-3.436	-2.864	-2.568
	1	-15.597	0.000***	10557.748	-3.436	-2.864	-2.568
	2	-12.411	0.000***	10623.771	-3.436	-2.864	-2.568

ADF Test of Bitcoin							
variable	Difference Order	t	P	AIC	Critical Value		
					1%	5%	10%
BTC	0	-1.508	0.53	30048.993	-3.434	-2.863	-2.568
	1	-8.422	0.000***	30033.637	-3.434	-2.863	-2.568
	2	-15.685	0.000***	30069.507	-3.434	-2.863	-2.568

ADF Test of Dogecoin							
Variable	Difference Order	t	P	AIC	Critical Value		
					1%	5%	10%
DOGE	0	-2.558	0.102	-11022.902	-3.434	-2.863	-2.568
	1	-8.643	0.000***	-11024.996	-3.434	-2.863	-2.568
	2	-14.608	0.000***	-10966.361	-3.434	-2.863	-2.568

Figure 5: Stationary Test Table

4.3.2 Index Selection

For the ARIMA model, I need 3 indexes: p, q, and d, which implies LA, MA, and difference order respective. The stationary test result yields that the first-order difference and second-order difference make the series smooth. Thus d can be 1 or 2.

Then, I need to determine the order of p and q. One way is to observe the ADF and PACF plots. However, to make the index more precise and objective, I first observe the graph and determined a propionate range and then compute the AIC and BIC information criterion.

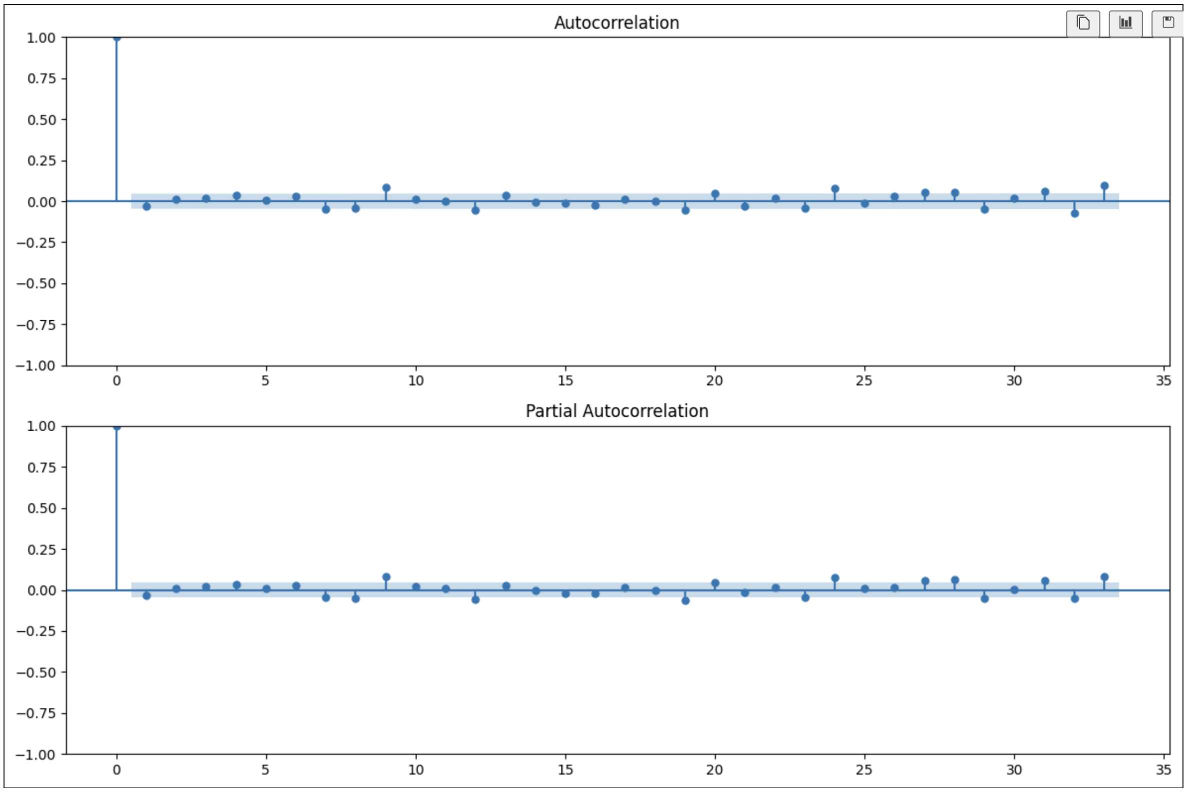

Figure 6: ACF and PACF of Bitcoin

For the ACF and PACF of Bitcoin, I notice there are more data out of confidence interval after 6. Thus, I bound the range of $p \in [1, 6]$ and $q \in [1, 6]$

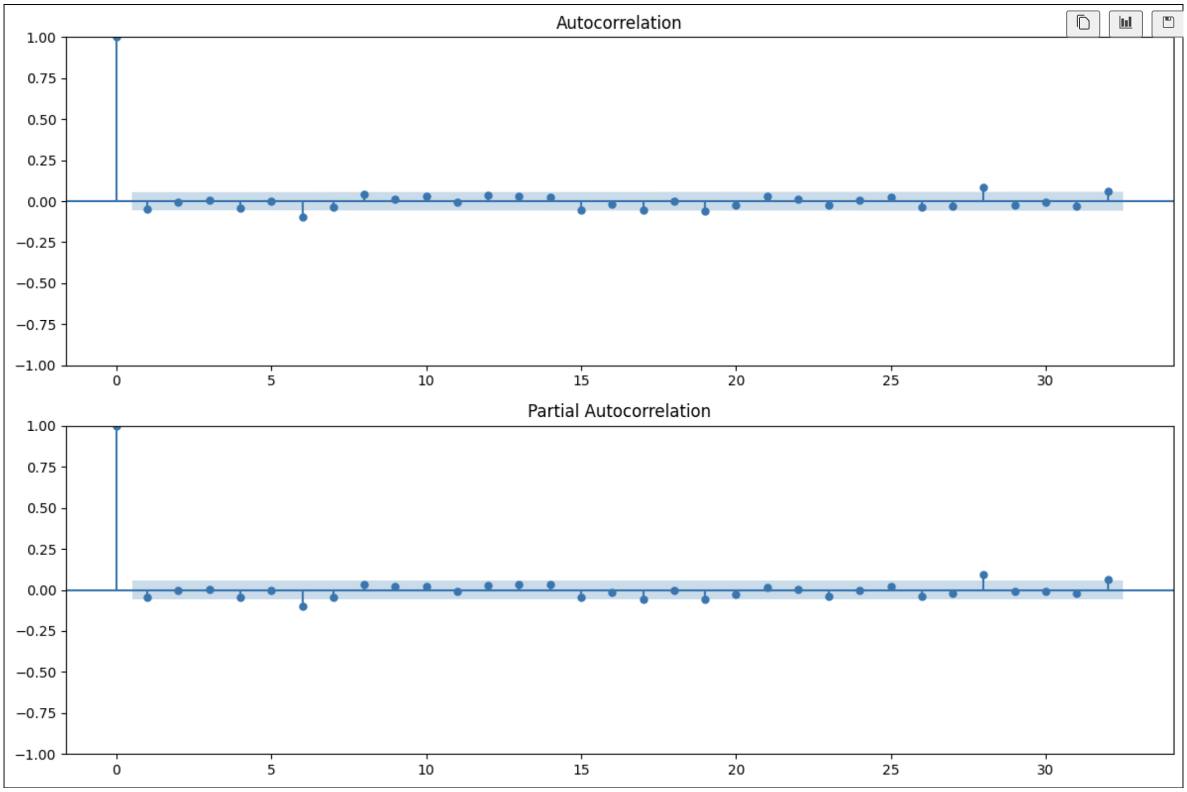

Figure 7: ACF and PACF of GOLD

For the ACF and PACF of Bitcoin, I notice there are more data out of confidence interval after 5. Thus, I bound the range of $p \in [1, 5]$ and $q \in [1, 5]$

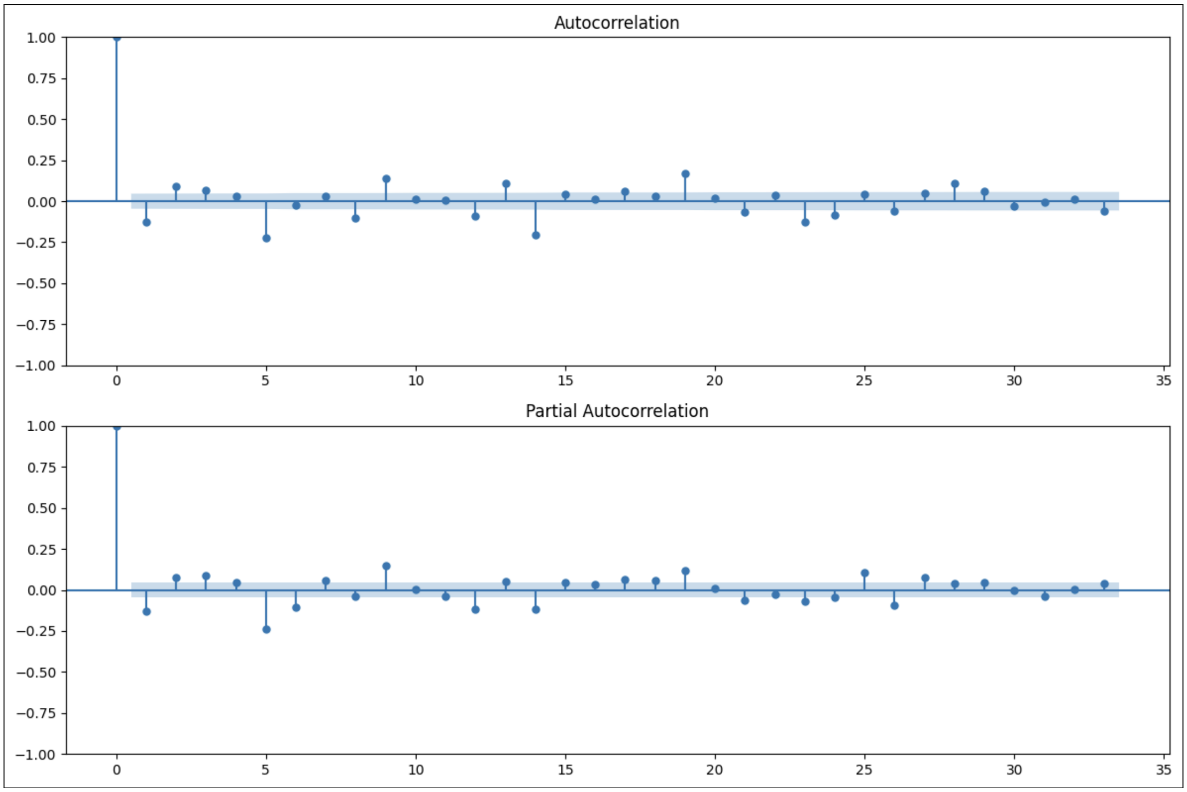

Figure 8: ACF and PACF of GOLD

For the ACF and PACF of Bitcoin, obviously, the smooth performance is bad. The first several difference order failed to stay in the confidence interval. Thus, I take the log before the first difference and improved the result:

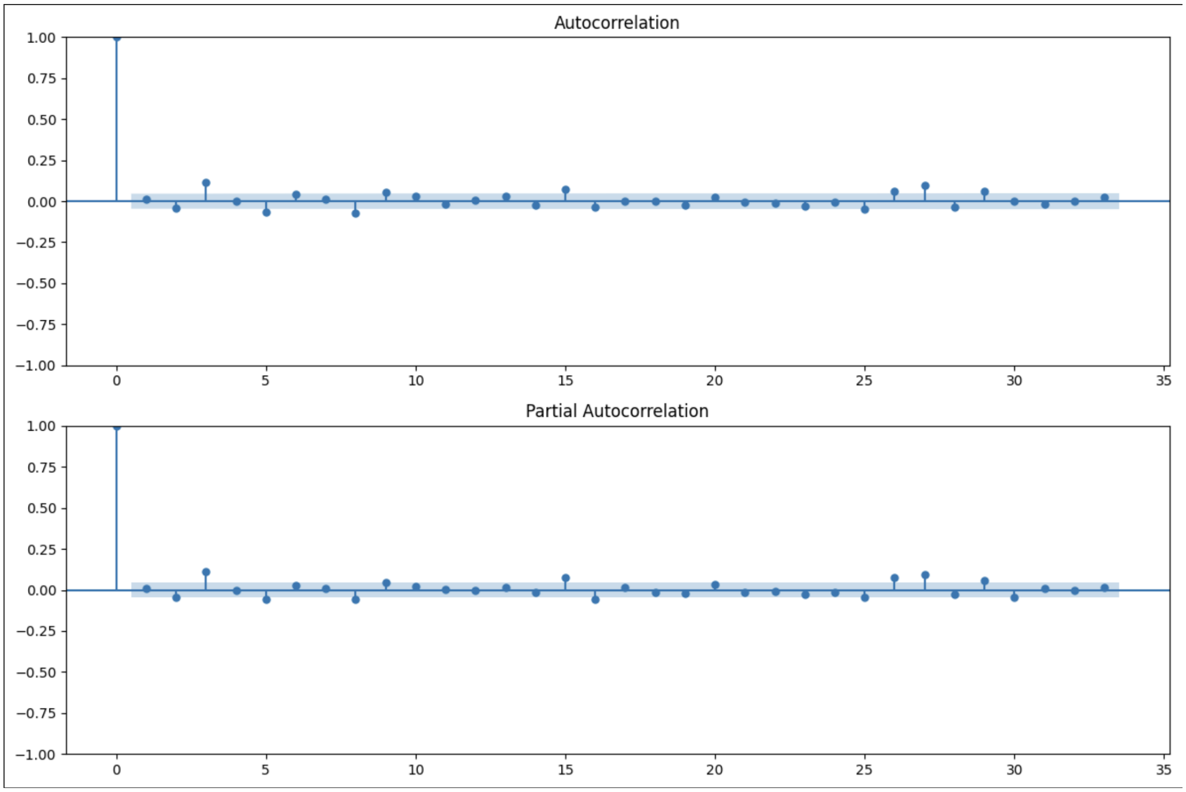

Figure 9: ACF and PACF of GOLD log diff1

This makes the interval [1, 2] stand within the confidence interval. Thus, I choose range $p \in [1, 2]$ and $q \in [1, 2]$

4.3.3 Define Three Performance Functions

To determine which times series model to use, we apply the auto-correlation function (ACF) to the data. The ACF is defined as the following:

$$ACF(k) = Corr(Y_t, Y_t - k)$$

where k is the number of time periods between values being correlated (also call the lag); Y_t is the value of the time series at time t, and $Corr()$ represents the correlation function.

After classification, the auto-regressive integrated moving average (ARIMA) model and the simple exponential smoothing (SES) model are the top choices for the data.

The formula of ARIMA(p, d, q) is given by:

$$\begin{aligned} y(t) = c + \Phi(1)y(t-1) + \Phi(2)y(t-2) + ... + \Phi(p)y(t-p) \\ +\epsilon(t) + \theta(1)\epsilon(t-1) + \theta(2)\epsilon(t-2) + ... + \theta(q)\epsilon(t-q) \end{aligned}$$

where $y(t)$ represents the time series at time t; c is a constant; $\Phi(1), \Phi(2), ..., \Phi(p)$ are parameters of the autoregressive part of the model representing the effect of the previous

p observations on the current value of y. Furthermore, $\epsilon(t)$ is the error term which represents the part of the series that cannot be explained by the autoregressive and moving average parts of the model; $\theta(1), \theta(2), ..., \theta(q)$ are the parameters of the moving average part of the model representing the effect of the previous q error terms on the current value of y. Last but not least, d is the order of differencing which tells the number of times the series has been differenced to make it stationary.

The formula of the SES model is given by:

$$\hat{x}_{t+1} = \alpha x_t + \alpha(1-a)x_{t-1} + \alpha(1-a)^2 x_{t-2} + ... + \alpha(1-a)^{t-1}x_1 + (1-a)^t l_0$$

where α is the smooth factor; t represents the time; and l_0 is equal to $\hat{x}_1$, which is the initial value. For each smoothed data obtained from the past data by weighting and summing, the closer it is to the current period, the greater its weight becomes. As a result, the closer the data is to the current period, the greater its impact becomes on the current period.

The results of both AIC and BIC are consistent, the order of the ARMA model for gold is (1, 1) and the order of the ARMA model for bitcoin and dogecoin is both (1, 1). And the difference parameter d is 1 for Gold and Bitcoin and 2 for Dogecoin.

4.4 Model Prediction Results

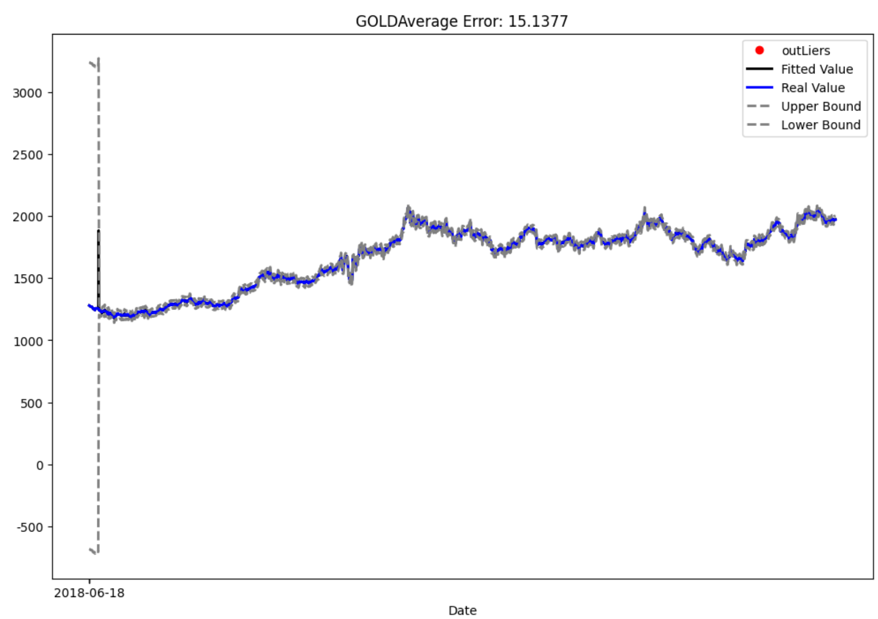

Figure 10: ARIMA of GOLD, $R_2 = 0.995$

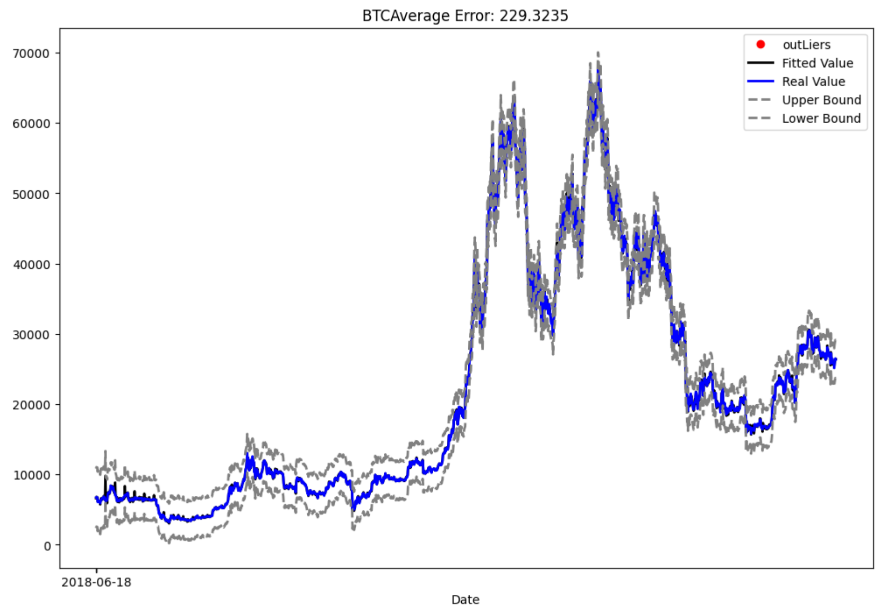

Figure 11: ARIMA of Bitcoin, $R_2 = 0.996$

https://www.overleaf.com/project/64bb4bc4db105360fb0a9d59

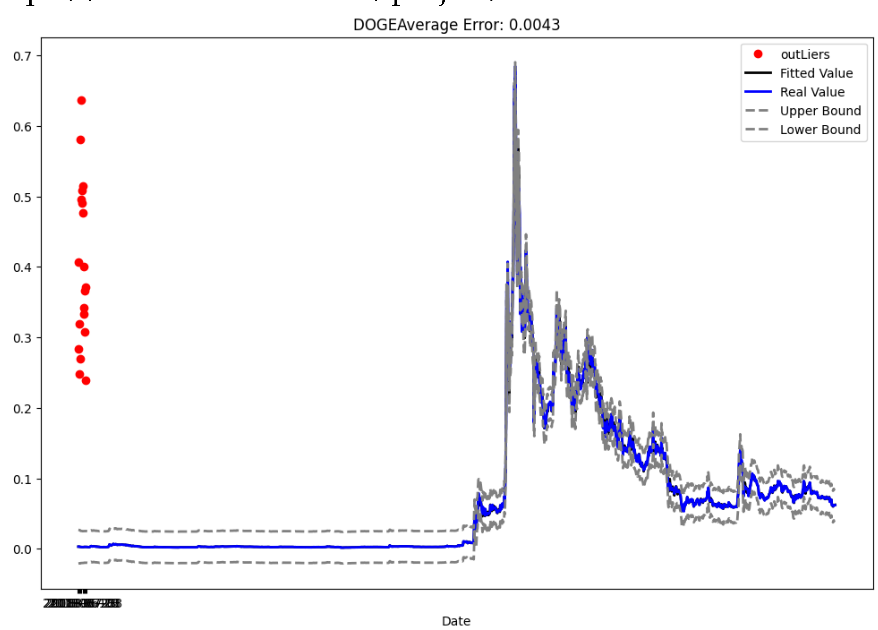

Figure 12: ARIMA of Dogecoin, $R_2 = 0.986$

In evaluating the accuracy of our predictions, we utilize the coefficient of determination (R^2) along with the upper and lower thresholds for the model's prediction results. Notably, the R^2 values obtained for all three models exceed 0.98, indicating an excellent fit of the models to the real data. This suggests that the models can effectively capture and represent the underlying patterns in the data.

Therefore, in the part of investment strategy, we will make investment decisions based on the predict results of the ARIMA model.

5 Trading Strategy analysis

5.1 Idea of Applying dynamic programming

When people invest in the three cryptocurrencies, what they know is the historical prices of the three cryptocurrencies, and based on them to make a trading strategy. The next day they will have to add the price today and update the data they use to predict the price of the following days. It's a dynamic process. Each day's day the trading strategy is given based on the previous data the data is updated day by day, recursively, until the last day.

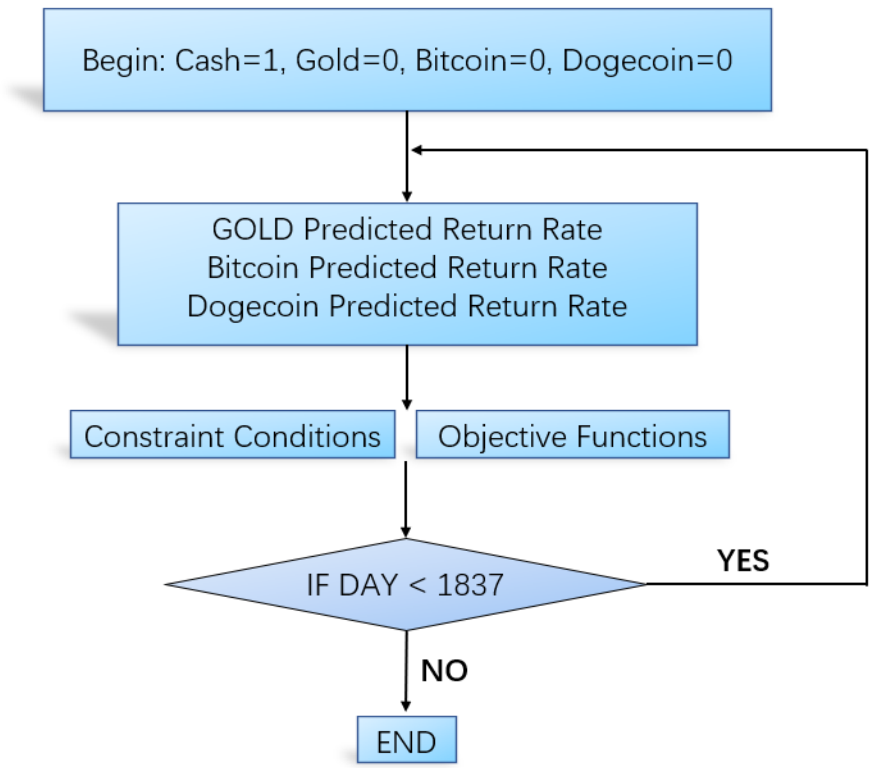

Figure 13: Dynamic Programming Flow Chart

5.2 Modle Assumption

To apply linear programming, I need constraints and a target function to optimize. Assume on k-th day, the proportion of cash, gold, bitcoin, and dogecoin is: $[c_k, g_k, b_k, d_k]$. Their sum is the total asset, so the constraint formula for total assets is:

$$g_k + b_k + c_k + d_k = 1$$

.

On k-th day, assume the price change of cash, gold, bitcoin, and dogecoin is: $[\Delta c_k^p, \Delta g_k^p, \Delta g_k^p, \Delta d_k^p]$. Although I haven't done any transactions, the asset distribution will change. The changed proportion of my asset is: $[(1+\Delta c_k^p)c_k, (1+\Delta g_k^p)g_k, (1+\Delta b_k^p)b_k, (1+\Delta d_k^p)d_k]$. Then, the new proportion will be normalized and distributed to $[c_k, g_k, b_k, d_k]$

When transactions happen including the spending of bitcoin, gold, and dogecoins, assume the transaction cause the change in proportion is $[\Delta c, \Delta g, \Delta b, \Delta d]$. Thus, the proportion of asset, after the transaction at that day is:

$$[(1+\Delta c_k^p)c_k - \Delta g - \Delta b - \Delta d]$$

5.3 Constraint Conditions

In this case, I need to make sure after each day's transaction, the proportion of each property should be larger or equal to zero:

$$\begin{cases} (1+\Delta c_k^p)c_k - \Delta g - \Delta b - \Delta d \geq 0 \\ (1+\Delta g_k^p)g_k - \Delta g_k \geq 0 \\ (1+\Delta b_k^p)c_k - \Delta b_k \geq 0 \\ (1+\Delta d_k^p)d_k - \Delta d_k \geq 0 \end{cases} \tag{1}$$

Above are only the most basic functions of the whole investing process. I also come up with additional formulas to constrain the process.

For risk-averse investors, ensures that the sum of the weights for gold, bitcoin and dogecoin is less than or equal to the weight of US dollars. This ensures that the total portfolio value is not negative:

$$(1+\Delta c_k^p)c_k \leq (1+\Delta g_k^p)g_k + (1+\Delta b_k^p)b_k + (1+\Delta d_k^p)d_k$$

To ensure the profolio is not over-invested in gold, bitcoin or digecoin, the following constrain can be added to ensure the weight of gold, bitcoin, and dogecoin is less or equal to the weight of US dollars:

$$\begin{cases} (1+\Delta c_k^p)c_k \geq (1+\Delta g_k^p)g_k \\ (1+\Delta c_k^p)c_k \geq (1+\Delta b_k^p)b_k \\ (1+\Delta c_k^p)c_k \geq (1+\Delta d_k^p)d_k \end{cases} \tag{2}$$

5.4 Objective Function

Objective function is the formula that the algorithm tries to maximize or minimize, within the given constrains. To evaluate the correctness of decisions, there are multiple ways I come up with.

5.4.1 Sharpe Ratio

To better evaluate the proper gain and potential risks of daily invstment, I apply sharp ratio.

The Sharpe ratio is a measure of how well an investment performs compared to a risk-free asset, such as a treasury bill or bond, after adjusting for its risk. The Sharp Ratio calculated by subtracting the risk-free rate from the expected return of the investment and dividing by the standard deviation of the investment's returns:

$$SR^p = \frac{E(r_p) - r_f}{\sigma_p}$$

A higher Sharpe ratio means a better risk-adjusted return.

5.4.2 Objective Function

When selecting the Sharpe Ratio as the objective function, there is a risk of over-reversing the trading strategy, especially when dealing with cryptocurrencies that tend to exhibit higher fluctuations compared to traditional investments like gold. The Sharpe Ratio may favor a more stable asset like gold, leading to an overly conservative approach where all the funds are allocated to gold.

To fix this issue and also maximize the earning rate for the following days, which is a risk-prefer investing strategy. One approach is to compute the product of the earning rate and the Sharpe Ratio. A higher product signifies a higher expected profit in the subsequent days, without considering the price fluctuations. This risk-prefer strategy aims to optimize returns while ignoring short-term price volatility.

To achieve a balanced consideration of both profit and risk, I consider the standard deviation of predicted values, denoted as σ_p, as a risk evaluation index. σ_p reflects the smoothness of predicted future data, and a higher value indicates higher potential risks. The objective function is then defined as $f_k = a \times Pro - b \times \sigma_p$, seeking higher profits while also considering potential risks.

To determine appropriate values for the weighting coefficients a and b, a basic optimization function is applied. Firstly select an initial pair of $[a, b]$, and implement a loop to iteratively explore different values of $[a, b]$ and evaluate their impact on the return on investment. This iterative process helps in finding a reasonably good solution, although it may not guarantee the global optimum. To address this limitation, I manually selected some $[a, b]$ pairs first, followed by fine-tuning with the optimization method to make small adjustments.

While this optimization function proves effective, there are potential challenges. It may yield satisfactory results but not necessarily the optimal ones. Additionally, there is a risk of getting trapped in local maxima rather than finding the global maximum. Thus, careful consideration and evaluation of the results are crucial to ensuring the reliability and robustness of the chosen $[a, b]$ values.

5.5 Constrained minimization of multivariate scalar functions

To calculate the minimum of the objective function, with given constraints, I applied constrained minimization of multivariate scalar functions, also called scipy package, in Python. This package provides an interface to various optimization algorithms, with setting constraints and bounds.

6 Result and Sensitivity Analysis

6.1 Results Exhibition

For convenience, I assume the starting budget for investors is 1000$. Of course, the starting budget is a key factor of ending profit, I will discuss it later.

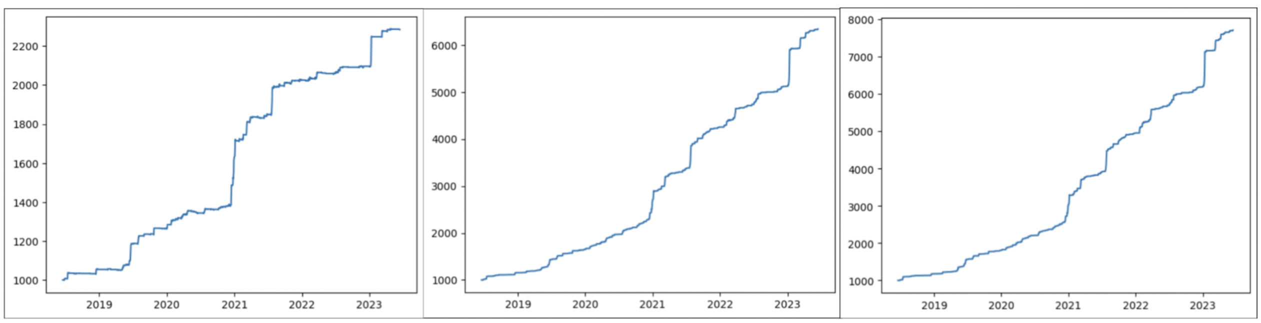

Figure 14: Earning for using different objective functions

By emphasizing the Sharp Ratio, which is a risk-averse strategy, the ultimate return is 2384. By emphasizing the maximum profit, the ultimate earning is 6339. The result by balancing considering both profit and risks ends up with 7702 returns.

6.2 Necessity of Sensitivity Analysis

Sensitivity analysis is a crucial aspect when evaluating a well-programmed model. It ensures that the model's performance remains robust and does not exhibit excessive sensitivity to irrelevant changes in hyperparameters. My purpose of conducting sensitivity analysis is twofold:

Firstly, it allows me to assess the model's stability and guard against overfitting past quotes. By identifying potential issues related to stability, I can ensure the model's reliability and accuracy for future predictions, thus avoiding over-reliance on historical data.

Secondly, sensitivity analysis helps me discover the factors that significantly impact economic efficiency. This is vital in determining the key drivers behind the model's performance and thus making informed decisions for optimizing trading outcomes.

The sensitivity analysis will be carried out from two perspectives: the initial budget and the commission fees. As individual investors' budgets may vary, it is important that the model exhibits consistent and promising results, regardless of the specific initial investment amount.

Additionally, since cryptocurrency trading can be very volatile and risky, it is essential to account for commission fees as unavoidable costs. Factoring in these fees becomes particularly crucial while trading across different types of assets, as transactions with minimal earnings may lead to negative profits.

By conducting a comprehensive sensitivity analysis, I aim to enhance the model's performance, minimize risks, and make well-informed investment decisions in the dynamic cryptocurrency market.

6.3 Sensitivity Test for Different Initial Amount

Actually, investors in the cryptocurrency market hold an average budget much larger than 1000$.

There is no definitive answer to the average amount per investor in the cryptocurrency market. I checked different sources, but they have different methods of estimating the number of investors and the amount they hold. However, based on some statistics, I try to make an approximation.

It can be searched that the total supply of Bitcoin, which is about 18.9 million at the end of 2022. Divided by the number of investors, I get an average of 0.28 Bitcoin per investor. Multiplying this by the average price of Bitcoin at that period, I get an average amount of about $11,200 per investor[8].

This is only a very rough estimate and cannot consider as the distribution of wealth. But it also reminds me of checking how investors' budgets may affect the final return.

By setting the starting budget at 1000, 10000, and 100000, respectively compute the ultimate return, graphs as follows:

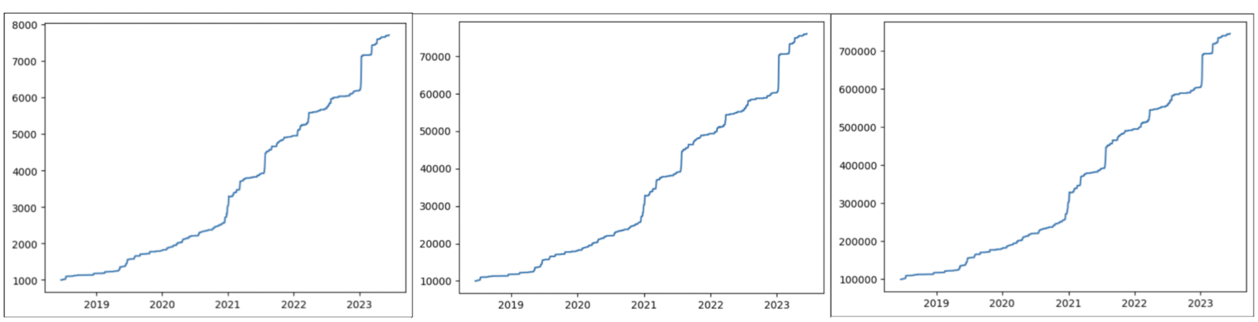

Figure 15: Earning for using different objective functions

The ultimate return is: 7702, 75941, and 745676. this indicates that the model returns a promising result, regardless of the specific initial investment amount.

6.4 Sensitivity Test for Different Commission Fee

Within each transaction, there is a transaction fee as an inevitable cost, knowing the number of fees I will pay when trading crypto beforehand could help develop strategies for maximizing profits. For gold, the transaction fee can vary based on factors such as the type of transaction, the broker involved, and the amount of gold being traded. According to Gold Broker, a globally recognized company for gold and silver investment[9], individuals should expect to pay between 2 and 5 percent over spot when buying gold, and a 1% commission when reselling it.

In contrast, Bitcoin and other cryptocurrencies have varying transaction fees depending on different aspects. An article discussing Bitcoin commission fees highlights that traders may encounter wire fees when moving funds to and from their bank accounts, higher commission fees when exchanging outside their geographical area, and elevated fees when engaging in multiple crypto transactions. Generally, the transaction fee for cryptocurrencies is typically around 1%.

For the subsequent investigation, I will proceed based on the assumption of within 1% transaction fee for cryptocurrencies, and 2% to 5% percent of transaction fees. This information will serve as a fundamental basis for evaluating the costs and potential profits associated with trading both gold and cryptocurrencies

6.4.1 Editing to Models

To ensure the model incorporates commission fees, adjustments to the constraint conditions are necessary. Let $[t_g, t_b, t_d]$ represent the transaction costs for gold, bitcoin, and dogecoin, respectively.

When conducting a transaction with gold on the k-th day, let Δg_k denote the proportion of gold changed. In this scenario, the actual change in cash is not Δg_k, but rather investors need to spend $\frac{\Delta g_k}{1-t_g}$ to make the purchase and receive $\Delta g_k(1-t_g)$ when selling.

Considering that the commission fee is relatively small, for simplification purposes, we approximate the cost of purchasing and selling. For instance, with a commission fee of 1%, I assume that \$100 of gold can be sold for \$99 and purchased for \$101. However, in reality, the actual purchasing amount would be $\frac{100}{99} = 101.0101$. Nonetheless, given the relatively small nature of the commission fee, we can overlook this difference during the programming stage to simplify the model.

The proportion of the cost, after the transaction, is:

$$\begin{cases} ((1+\Delta c_k^p) - (1 \pm t_g)\Delta g_k - (1 \pm t_b)\Delta b_k - (1 \pm t_d)\Delta d_k \\ (1+\Delta g_k^p)g_k + \Delta g_k \\ (1+\Delta b_k^p)b_k + \Delta b_k \\ (1+\Delta d_k^p)d_k + \Delta d_k] \end{cases} \tag{3}$$

The edited version of the constraints function is:

$$\begin{cases} (1+\Delta c_k^p)c_k - (1 \pm \Delta t_g)\Delta g - (1 \pm \Delta t_b)\Delta b - (1 \pm \Delta t_d)\Delta d \geq 0 \\ (1+\Delta g_k^p)g_k - \Delta g_k \geq 0 \\ (1+\Delta b_k^p)c_k - \Delta b_k \geq 0 \\ (1+\Delta d_k^p)d_k - \Delta d_k \geq 0 \end{cases} \tag{4}$$

After implementing the updated constraints and rerunning the optimized algorithm, the following table illustrates the results. The table includes commission fees for gold, bitcoin, and dogecoin, along with the corresponding final returns. It is evident that as commission fees for gold and cryptocurrencies increase, the overall return value decreases when adhering to the model.

commission fee	gold	0	0	1%	1%	2%
	bitcoin	0	1%	1%	1%	1%
	dogecoin	0	0	0	1%	1%
final returns		7702	6750	6526	6070	5718

Figure 16: Earning for using different objective functions

In this table, commission fees for gold, bitcoin, and dogecoin are respectively considered, with the final returns listed accordingly. It can be recognized that with the rising commission fees of gold and cryptocurrencies, the ultimate return value after complying with the model decrease.

Furthermore, by tracking the present value of assets on a daily basis, we obtain a graph that depicts the daily assets' value. Despite the decrease in returns due to commission fees, the trend of asset increment remains relatively consistent.

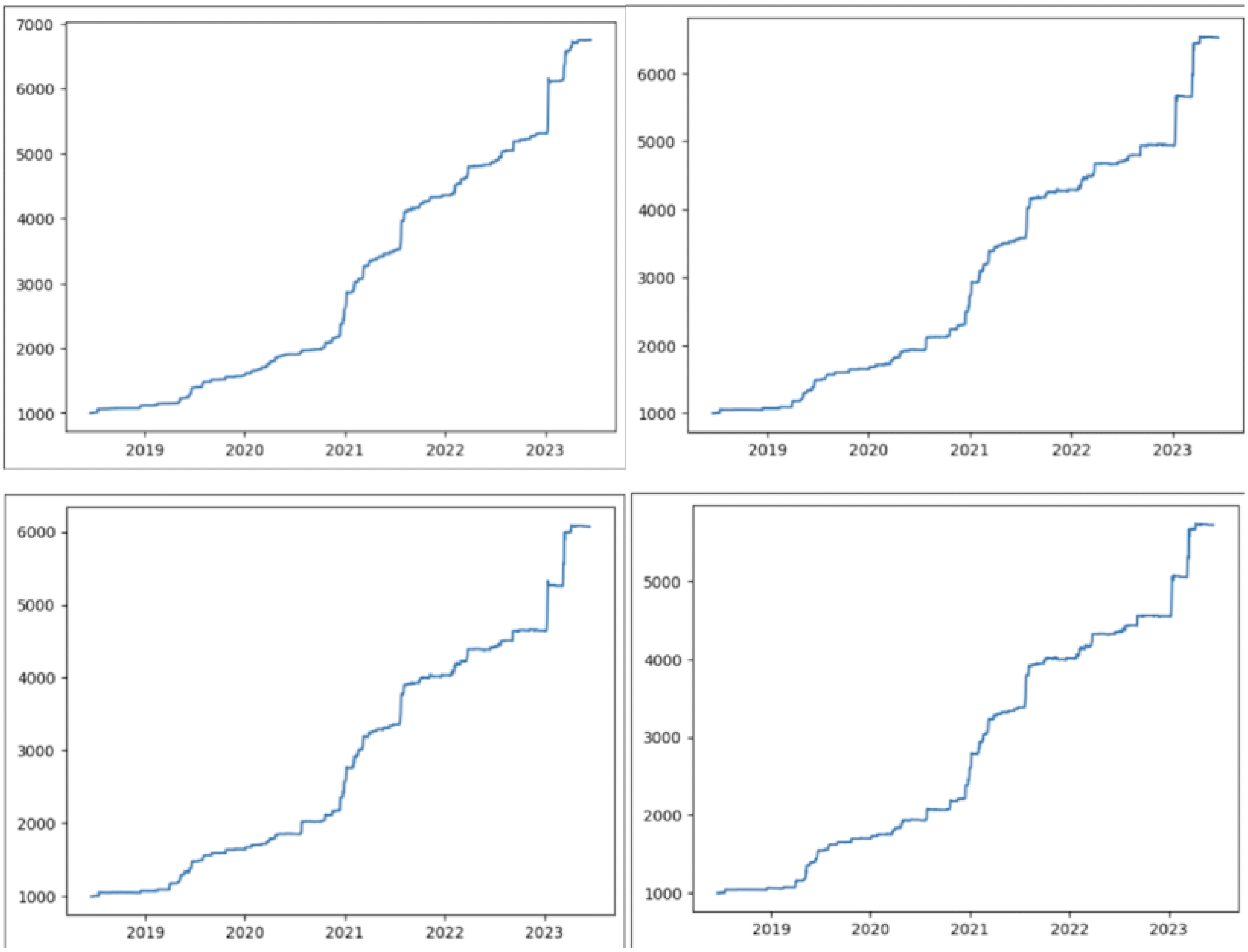

Figure 17: Earning for using different objective functions

During the sensitivity analysis, the model demonstrates satisfactory performance when facing varying transaction fees.

However, it is notable that there is a significant decrease in returns as commission fees rise. Therefore, it is advisable to explore methods to minimize high commission fees. For instance, holding a larger gold value can decrease the commission fee for gold transactions, and conducting cryptocurrency transactions within the same platform or domestically can reduce the commission fee for cryptocurrencies.

6.5 Model Appraisal

During the evaluation, it is evident that the optimization function severely restricts the transaction of dogecoin due to its extremely rapid fluctuations within a short period. The objective function $f_k = a \times Pro - b \times \sigma_p$ penalizes dogecoin heavily for its lack of smoothness.

In comparison, bitcoin exhibits relatively less fluctuation and demonstrates reasonably promising return values. As a result, the model allocates the majority of the investment into bitcoin rather than dogecoin.

It's also notable that, on 6/18/2018, which marks the starting day of applying this model for investment, the estimated profit of bitcoin is 6735, with the highest value being 67566. Despite conducting the best trading option, the estimated maximum return remains at 10 times the initial investment.

However, reflecting on the past two years, when bitcoin's price was only around $500, the model showed a remarkable increase in assets. The ultimate return amounted to 82472, and the total return rate reached an impressive 8247.2%, which is an excellent result. In reality, someone who purchased Bitcoin at that time and sold it when its price reached its acme could have earned over 100 times the initial investment.

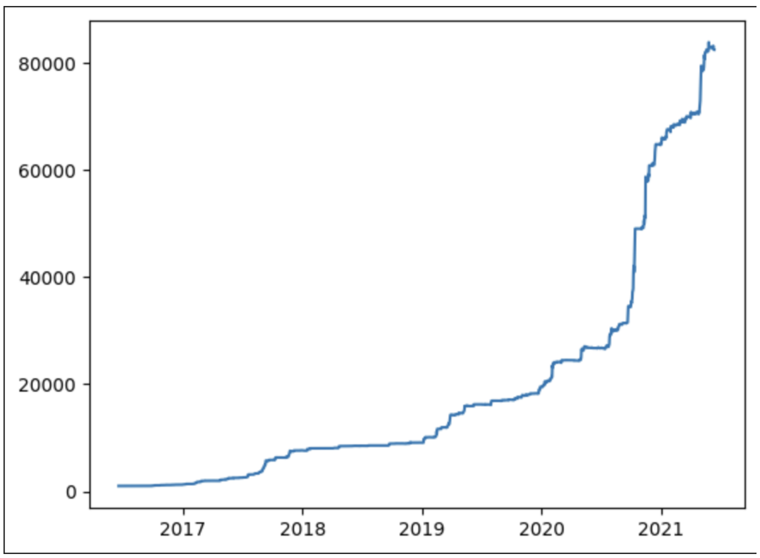

Figure 18: Earning for using different objective functions

Looking at each investment period, the risk-prefer trading strategy consistently outperforms the risk-averse trading strategy, yielding higher returns. The Sharpe Ratio is not an ideal metric to evaluate transactions between cryptocurrencies, which are prone to rapid fluctuations.

In addition, when investing in cryptocurrencies, it is advisable to focus on larger cryptos like Bitcoin. Over the five-year period, bitcoin's value has shown promising growth without significant drops or extreme volatility. Conversely, investing in smaller cryptos like Dogecoin can be highly risky, as their values may experience dramatic increases but also sharp declines in a short span.

7 Evaluations

Financial markets are renowned for their inherent volatility, and time series analysis methods predominantly rely on historical price data to extract potentially valuable insights for forecasting future movements. However, it is essential to recognize that these methods often lack consideration for the underlying causes driving stock price fluctuations. As a result, their analyses are typically more intuitive and limited to short-term predictions.

Regarding the strategy combination approach employed to generate new stocks, it is important to acknowledge its limitations. Due to the adoption of a singular measure and a limited set of indicators, this approach fails to account for the diverse risk-return characteristics inherent in various trading strategies. Moreover, the fitness function utilized in this context solely evaluates performance over time without factoring in cumulative performance.

References

[1] IBM Corp. (2017). IBM SPSS Statistics for Windows. Armonk, NY: IBM Corp. Retrieved from https://hadoop.apache.org

[2] Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., … Oliphant, T. E. (2020). Array programming with NumPy. Nature, 585, 357–362. https://doi.org/10.1038/s41586-020-2649-2

[3] Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in Science & Engineering, 9(3), 90–95.

[4] McKinney, W., & others. (2010). Data structures for statistical computing in Python. In Proceedings of the 9th Python in Science Conference (Vol. 445, pp. 51–56).

[5] Scientific Platform Serving for Statistics Professional 2021. SPSSPRO. (Version 1.0.11)[Online Application Software]. Retrieved from https://www.spsspro.com.

[6] Chen, J. (2021, July 29). Sharpe ratio. Investopedia. https://www.investopedia.com/terms/s/sharperatio.asp

[7] Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., … SciPy 1.0 Contributors. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python

[8] Molla, R. (2021, October 29). Bitcoin market capitalization as of October 29, 2021. Statista. https://www.statista.com/statistics/377382/bitcoin-market-capitalization/

[9] GoldBroker.com. (2023). Rates. Retrieved August 4, 2023, from https://goldbroker.com/rates#xpd-content

[10] Just Start Investing. (2020, December 16). Understanding Bitcoin Fees. https://www.juststartinvesting.com/understanding-bitcoin-fees/