Problem Chosen	2022 MCM/ICM Summary Sheet	Team Control Number
C		2220297

Trading Strategy Based on SVR Prediction and Risky Portfolio Decision

Summary

Investing is a means of asset growth. Many people tend to invest their assets for greater benefits. There are many types of investments, which can be classified by whether they are risky or not. Investing risky investment probably results in a loss. However, "risk and opportunity go hand in hand" Traders frequently buy and sell volatile assets for profit. In the process of trading, optimal trading strategies can maximize return. Therefore, choosing the optimal investment portfolio becomes the key to getting the most benefit in trading.

In order to complete the task given by the trader, we established the following model to provide traders with the optimal portfolio.

For model 1, we established a price predictive model suitable for Bitcoin and gold, which is based on **Support Vector Regression**. The model can accurately predict the price of the day based only on data up to the trading day. The most suitable kernel function, Gaussian kernel, is finally determined in several attempts that can make the model achieve the highest accuracy. In addition, the model can not only provide the forecast price of the day, but also provide relevant risk indicators (**NMSE**, Normalized mean square error) for decision-making.

For model 2, we establish a decision model based on the **Markowitz Portfolio Selection Model** to formulate the trading strategy for each trading day. The model helps the trader decide whether to buy gold or bitcoin on the trading day and gives the optimal trading strategy after weighing expected returns and risks. This model can not only avoid investment risk effectively, but also give the most investment choice to obtain the maximum benefit.

From September 11, 2016 to September 11, 2021, we used these models to develop optimal trading strategies and invested in bitcoin and gold with an initial capital of $1,000. In the end, we made a total profit of **$79,155.5** dollars. Various risk indexes and stability analysis can prove that our models have excellent performance and can provide optimal strategies for the trader. Through sensitivity analysis, we finally find that transaction costs are negatively correlated with final profits.

Our model has a number of significant advantages. Especially in the aspect of prediction, SVR is the core model, which makes the model have advanced mathematical algorithm and better accuracy than neural network and many models. Besides, our model is also very scalable. Because the models used are well adapted to more complex situations. Markowitz's model is applicable to a variety of investment methods, while SVR also has considerable stability for volatile assets with different risks. When the amount of volatile assets' types increases, this model can still be used normally after some improvement. Of course, there are some possible improvements in our model. In order to adapt to the requirements of the competition, we made some assumptions on the model, which made the model too idealistic in some aspects. In order to better adapt to the actual environment, we should make certain strengthening and give up some assumptions.All in all, the model we established can successfully complete this task and show excellent performance in it.

Key words: Trading strategy, SVR, Markowitz Portfolio Selection Model, sensitivity analysis.

Contents

1 1 Introduction

1.1 Background

"Value investing doesn't guarantee us a profit, but it gives us the only chance of real success."

---- Warren Buffett

People increase their assets through various investments. Buying volatile assets is a way of investment with both risks and benefits. Traders frequently buy and sell volatile assets to maximize profits. Each transaction incurs a fee. A good trading strategy can help traders avoid some risks in trading and bring considerable profits. Therefore, in view of the transaction of volatile assets, making the right trading strategy becomes the key to purchase volatile assets.

Gold and bitcoin are two typical volatile assets. Traders can make money by buying and selling these two assets at the right time. The difference is that the two prices change by different amounts, and gold can only be traded on set trading days, while bitcoin can be traded every day. Traders need to time the two trades well, buy when the price is low, sell when the price is high, and profit from the difference between the two prices. Therefore, a reasonable mathematical model should be established to predict the price trend of gold and bitcoin, and the best trading strategy can be obtained -- traders can sell their own gold and bitcoin at the peak value, and timely buy at the trough value, so as to maximize profits. In addition, trading strategies need to take into account certain risks brought by price forecasts and make different decisions for different risk levels. Reduce or stop buying and selling when risk becomes too high.

1.2 Restatement of the Problem

At the request of traders, we developed a model. The model determines trading strategies based only on past daily prices. From September 11, 2016 to September 11, 2021, the model will offer the trader a portfolio consisting of cash, gold and bitcoin on each trading day, starting at $1,000. Each transaction (purchase or sale) costs α% of the amount traded as commission. And eventually, maximize their total return.

To achieve our goal, our specific work is as follows:

- Develop a model to predict the price of the day based only on the past stream of daily prices to date
- Establish a model that provides the best trading strategy to determine each day whether the market trader should buy, hold, or sell their portfolio.
- Finally, figure out how much the initial ＄1000 investment will be worth at the end of the five-year trading period using the model and strategy?
- Prove that the model provides the optimal strategy.
- Determine the sensitivity of the strategy to transaction costs.
- Figure out the strength and the possible improvement of the model.

1.3 Our Work

Our work is shown in Figure 1-1, mainly including the data processing, two models' building and sensitivity analysis. Before modeling, we preprocessed the data provided. Since gold is only traded on trading days, data cleaning is required to fill in the missing price data on weekends and statutory holidays to ensure the normality of the prediction model. The model predicts the price trend of the current day by using the data for a period of time before the trading day, and gives the predicted price and the corresponding risk coefficient (to characterize the confidence of the predicted value and the stability of the prediction). Then the decision model makes corresponding trading strategies to the predicted values, and gives the best investment portfolio [C, G, B]. In a sensitivity analysis, we will explore and determine the impact of transaction costs on this forecast-decision model. And through a series of evidences, it is proved that the model gives the best investment plan.

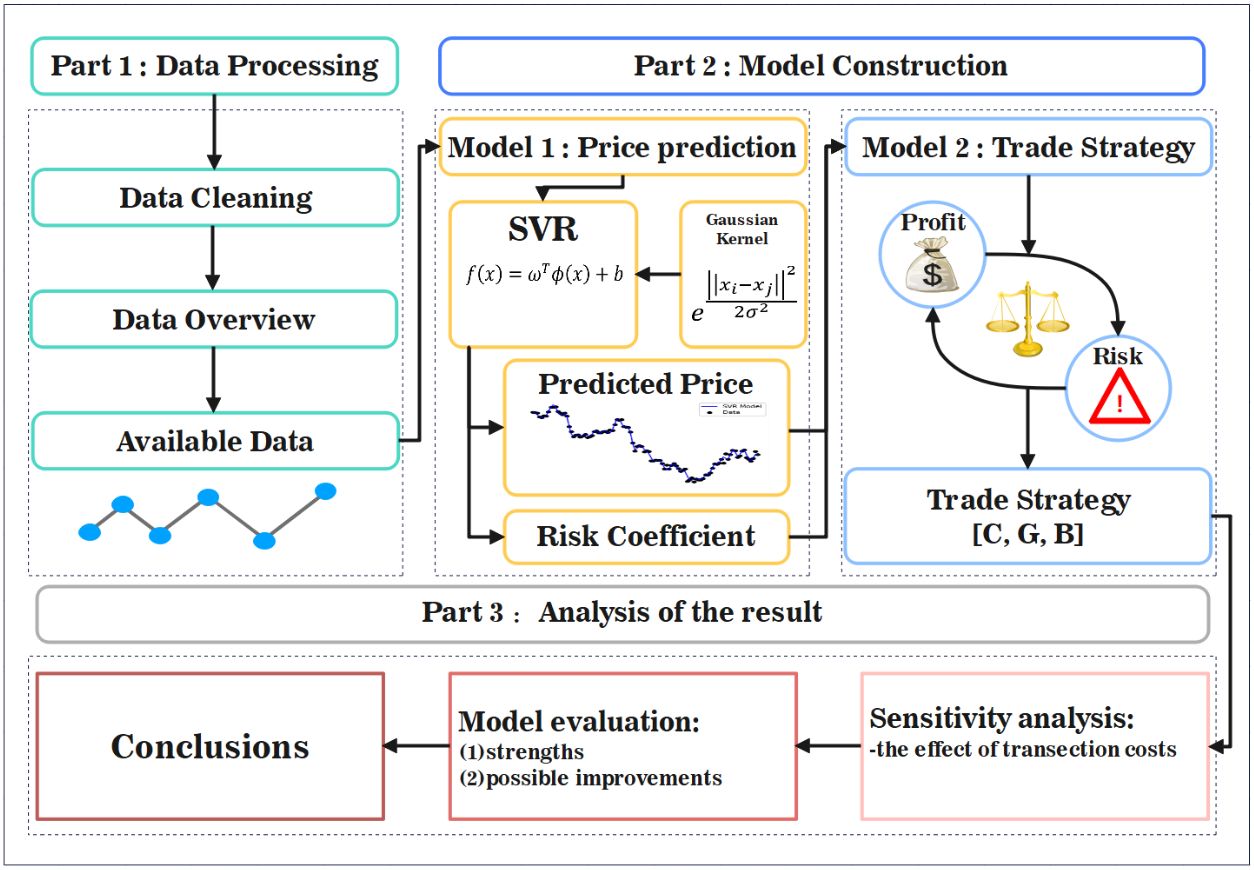


Figure 1-1 Our Workflow

2 Assumptions

To simplify the problem, we make the following assumptions. Later, we may relax some of these assumptions to optimize our model making it more applicable in the complex reality trade.

- Buying and selling are limited to cash, gold and Bitcoin and do not involve any other form of investment. The trading strategy is simplified to buy, hold, or sell assets in the [C, G, B] portfolio. The initial conditions are [1000,0,0]
- There is no cost to holding an asset, and the amount does not change without buying or selling
- The commission for each transaction (purchase or sale) costs α% of the amount traded. Assume α_{gold} = 1% and $\alpha_{bitcoin}$ = 2%, which do not change over the entire transaction period,

- The prices of gold and bitcoin are independent. One's change has no effect on another's.
- For an initial period of time, just hold cash and do not trade. The trading strategy is given after collecting a certain amount of data for prediction. And at the end of the transaction period, all gold and bitcoin will be sold and the trader will hold cash only.
- The minimum scale of transaction time is day. If sold on the same day, the portion bought will not gain.
- To simplify the model, we assume that gold and Bitcoin can be bought in any amount, rather than integer multiples of the unit price.
- At the end all the money turns into cash
- Sell high and buy low when you can. Specifically, to buy when the previous two days have fallen and the third day is expected to fall (and vice versa)
- NMSE was used to quantify the error of the second day forecast.

3 Notions

Table 3-1 Notations used in this literature

Symbol	Definition
C(t)	Cash (in U.S. dollars)
G(t)	Gold (in troy ounces)
B(t)	Bitcoins
α_{gold}	The proportion of transaction costs in the amount traded of gold
$\alpha_{bitcoin}$	The proportion of transaction costs in the amount traded bitcoin
P_G	The predicted price of gold
P_B	The predicted price of bitcoin
R_G	The risk coefficient of gold
R_B	The risk coefficient of bitcoin
R_{G0}	The threshold of R_G
R_{B0}	The threshold of R_B
A	the Attenuation coefficient of gold
B	the Attenuation coefficient of bitcoin
r_{Gold}	Gold expected rate of return
$r_{Bitcoin}$	Bitcoin expected rate of return
r_P	Total rate of return
σ	Variance
E(r)	Expected return
k	Amount of days used for prediction

4 Data Processing

4.1 Data Pre-processing

There are two sets of data presented in this topic. Price data for gold and bitcoin from September 11, 2016 to September 11, 2021, respectively. Unlike Bitcoin, gold can only be traded during the trading day. Therefore, there is no price data for gold on non-trading days. It should be data cleaned first. Considering that the price of gold and bitcoin on the same day needs to be compared when making decisions, it is not advisable to directly delete the data on non-trading days of gold and rearrange its timeline. Therefore, we choose to fill in the missing price data on non-trading days in order to predict the trend of the day.

4.1.1 Data Cleaning

What needs to be preprocessed is the price data of gold. We fill in the data of non-trading days with the price data of the previous day to make the price curve appear horizontal (the increase is 0). As a result, in the decision model, it is judged that buying or selling gold on non-trading days will not bring profit, so gold will not be traded. Therefore, the simplification of judgment is realized, so that the model can make decisions normally.

4.1.2 Data Overview

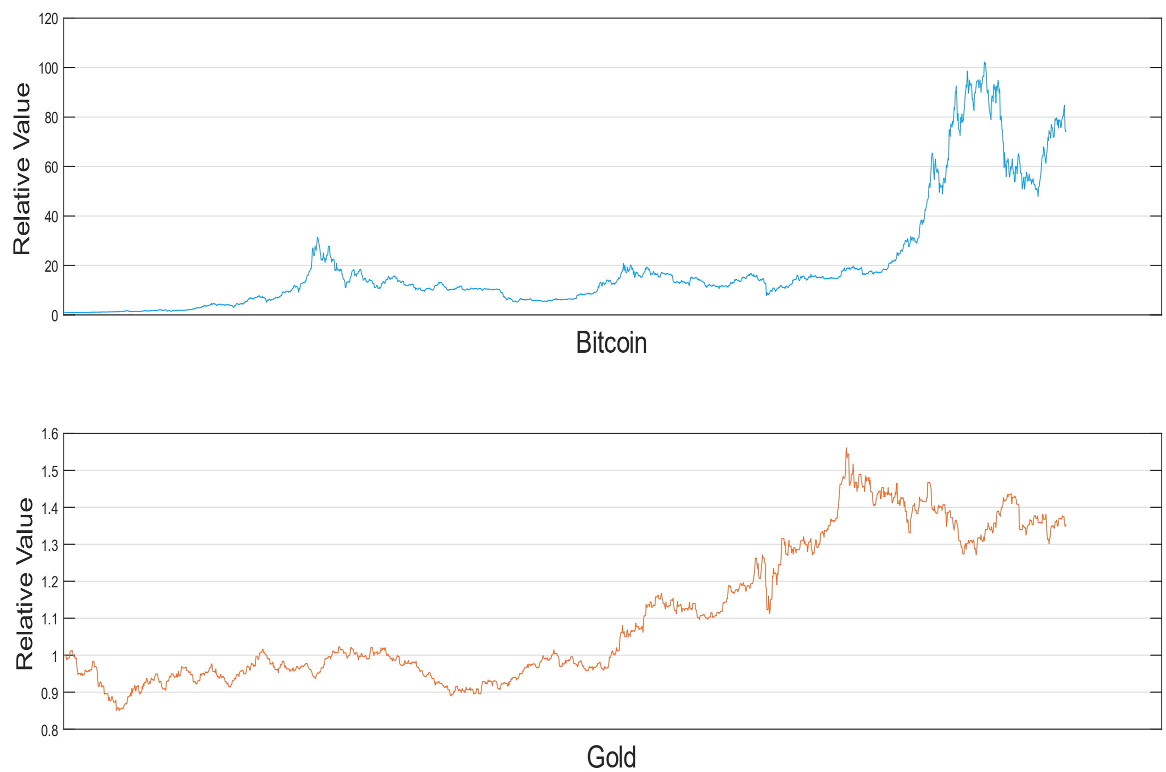


Figure 4-1Gold and Bitcoin price gains relative to the first day（ September 11, 2016）

By observing and comparing the rises of gold and bitcoin over the same time period as shown in Figure 4-1,(The independent variable on the horizontal axis of the two graphs is time and the same time axis is used, and the vertical axis is the increase of their respective prices relative to the first day.) it is obvious that bitcoin's change is much larger than that of gold. It is clear that investing in Bitcoin is riskier and more likely to yield greater returns, while investing in gold is less risky and less rewarding. Therefore, if you want to maximize your returns, you should invest more in Bitcoin to obtain high returns on the premise of adequate estimates and as little risk as possible. The change in gold is relatively gentle at first, and cash can be used to buy gold when Bitcoin is risky or unsuitable for investment.

5 Model Construction

We mainly built two models to accomplish the tasks that traders gave us. One is a prediction model that predicts the price trend of the day and gives the corresponding risk coefficient based on past price data. The model needs to give the price forecast and trend of the day to further determine the expected return of the investment, and at the same time give several risk indicators. In order to estimate the risk of investment; the other is a decision-making model that formulates a daily trading strategy based on the risk coefficient and investment return indicators and gives several evaluation indicators. The model needs to estimate, calculate and select whether to purchase, the type of purchase, the amount of purchase, the timing of purchase and the risk of purchase. In this way, traders are provided with the optimal trading strategy on each trading day, so that at the end of the trading period, traders can obtain the maximum profit and income.

5.1 A Price-predicting Model Based on Support Vector Regression

5.1.1 Application of SVR in Predictive Model

This prediction model uses **SVR** (Support Vector Regression). The starting point of SVM is later than that of neural network, and there is less research on SVM in financial time series forecasting. However, after previous analysis, SVM can be used for the analysis of financial time series forecasting, and the forecasting effect is excellent. Trafalis and Ince use SVM regression to predict stock outcomes, and the results show that SVM predicts better than MLP and ARIMA. Tay and Cao used support vector machines to predict species financial time series data. The results show that SVM is better than BP neural network in terms of Root Mean Squared Error (NMSE), Mean Square Error (MAE), and trend accuracy (DS). The prediction performance of the network is better. As the application of support vector machine in financial time series forecasting becomes more and more popular, many SVM models emerge, such as WSVM (Wavelet Support Vector Machine), fuzzy approximate support vector machine, etc. The prediction accuracy of support vector machine continues to improve.

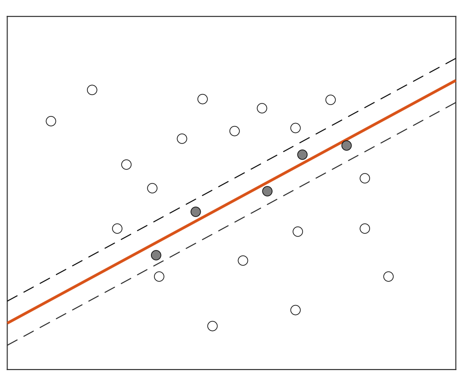


Figure 5-1 SVM classification diagram

SVM is a model proposed by Russian statistician and mathematician Vampuni. Initially, this model is used to select the classification surface of the sample points, so that the classification surface can accurately classify the sample points, and the interval between the two parts is as large as possible.

The classifier for this model is

$$\omega x_i + b = 0$$

Where ω is the coefficient matrix, representing the direction of the classification plane, and b is the threshold, representing the up and down orientation in the coordinate axis of the classification plane.

The model was originally used to solve binary classification planning in mathematics. With continuous research and improvement, the algorithm gradually expanded to the field of regression.

According to the purpose, SVM can be divided into SVC (Support Vector Classifier) and SVR (Support Vector Regression Machine) According to the purpose of the topic, we chose the SVR model. The model is good at handling nonlinear regression problems. The main principle is: when the model encounters data that cannot be linearly classified, the Kernel function is used to project it into a high-dimensional space for linear classification while avoiding the disaster of dimensionality.

Let $\phi(x)$ denote the feature vector after mapping x. Therefore, the model corresponding to dividing the hyperplane (classification plane) in the feature space can be expressed as:

$$f(x) = \omega^T \phi(x) + b$$

where ω and b are the model parameters as defined in the previous formula.

The process of linear regression is transformed into a minimum value problem, as follows:

$$min_{\omega,b} \frac{1}{2} \left||\omega|\right|^2$$
$$s.t. \ \ y_i(\omega T \phi(x_i) + b) \geq 1 \quad i = 1,2,\cdots,m$$

The solution of the above formula will involve calculating $\phi(x_i)^T \phi\left(x_j\right)$, because the dimension of the feature space may be very high, or even infinite. So we use the kernel function:

$$\kappa\left(x_i, x_j\right) = \phi(x_i)^T \phi\left(x_j\right)$$

So it can be solved:

$$f(x) = \omega^T \phi(x) + b$$

$$= \sum_{i=1}^{m} \alpha_i y_i \kappa(x, x_i) + b$$

According to the principle of SVR, we hope that the samples are linearly separable in the high-dimensional feature space, so the quality of the feature space directly affects the performance of SVR. "The choice of kernel function" has become the biggest variable of SVR. Selecting the appropriate Kernel function can make the prediction accurate. After a lot of literature searches and attempting, we determined that the most suitable kernel function is the radial basis kernel function (Gaussian kernel)

5.1.2 Modeling Process

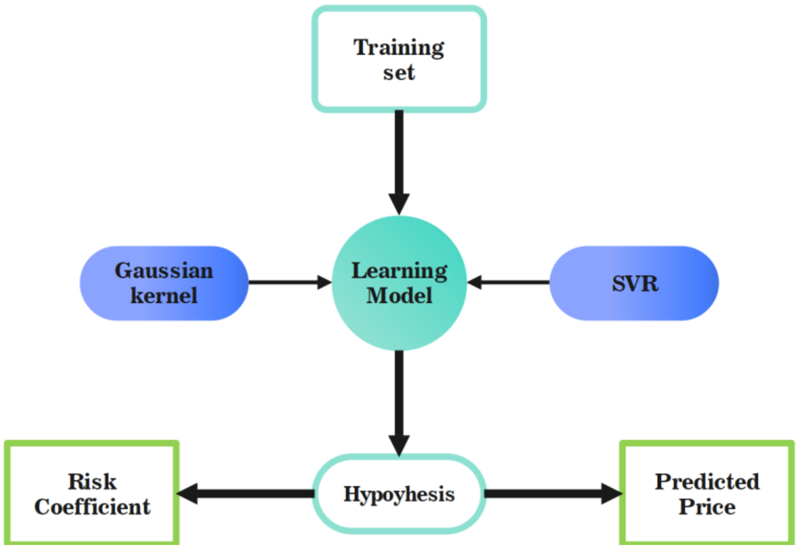


Figure 5-2 Process of Predicting Prices

The specific process of the model is shown in Figure 5-2. First, import the cleaned data, and select the price data up to 3 days before the trading day (considering that the data that is too far is not helpful for prediction) to train the retrieved SVR model. . Then, use the model to predict the predicted value of the price on the trading day and give (NMSE) as a risk indicator.

5.1.3 Determination of the best predictive model

As we have mentioned above, the accuracy of the model prediction mainly depends on the

name	equation	parameter
Linear kernel function	$\kappa\left(x_i, x_j\right) = x_i^T x_j$	
Polynomial kernel function	$\kappa\left(x_i, x_j\right) = \left(x_i^T, x_j\right)^d$	$d \geq 1$,powers of polynomials
Gaussian kernel function	$\kappa\left(x_i, x_j\right) = e^{-\frac{\left\|x_i - x_j\right\|^2}{2\sigma^2}}$	$\sigma > 0$,the width of Gaussian kernel
Sigmoid function	$\kappa(x_i, x_j) = tanh(\beta x_i^T x_j + \theta)$	$tanh$, Hyperbolic tangent function; $\beta > 0,\ \theta < 0$

selection of the kernel function. The following is the process by which we determine the Gaussian kernel as the optimal kernel function. Following are the steps to choose a kernel function:

Table 5-1 Available kernel funtions

Step 1: Use the basic knowledge of commonly used kernel functions to pre-select kernel functions. Available kernel functions are shown in Table 5-1.

- Linear kernel function, mainly used for linearly separable cases
- The polynomial kernel function ,is suitable for orthonormalized data and belongs to the global kernel function, allowing data points that are far apart to affect the value of the kernel function. The polynomial kernel function has many parameters. When the polynomial order d is relatively high, the learning complexity will be too high, and the phenomenon of "overfitting" will easily occur. The element value of the kernel matrix will tend to be infinite or infinitely small, and the calculation is complicated. The degree will be too large to calculate.
- The Gaussian radial basis function（RBF） is a kernel function with strong locality. This kernel function is the most widely used one. It has better performance regardless of large samples or small samples, and it has fewer parameters than the polynomial kernel function. , so in most cases, Gaussian kernel function is preferred. The radial basis kernel function is a local kernel function. When the data point is farther from the center point, the value will become smaller. The Gaussian radial basis kernel has good anti-interference ability to the noise existing in the data. Due to its strong locality, its parameters determine the function range, which weakens with the increase of the parameter σ. Moreover, there is another point of view that many situations in nature conform to the normal distribution, so the laws of nature determine that the Gaussian kernel function has a wider range of applications. Simply put, Gauss radial basis function is a kernel function with strong locality, and its extrapolation ability weakens with the increase of parameters. The kernel function in polynomial form has good global properties and poor locality.
- The Sigmoid function， When used as the kernel function, the support vector machine implements a multi-layer perceptron neural network. Moreover, the theoretical basis of the support vector machine determines that it finally obtains the global optimal value instead of the local minimum value, and also ensures its good generalization ability for unknown samples without over-learning phenomenon.

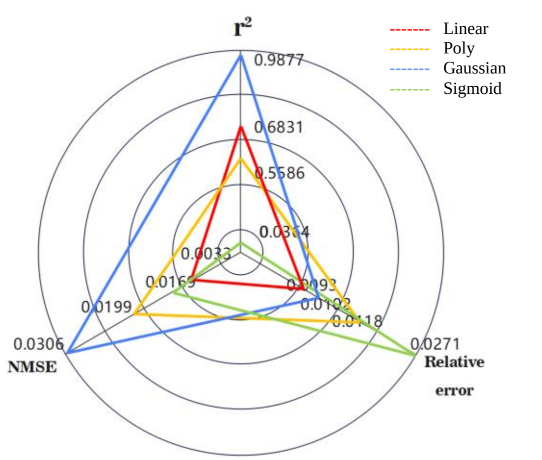


Figure 5-4 Comparison of three indicators

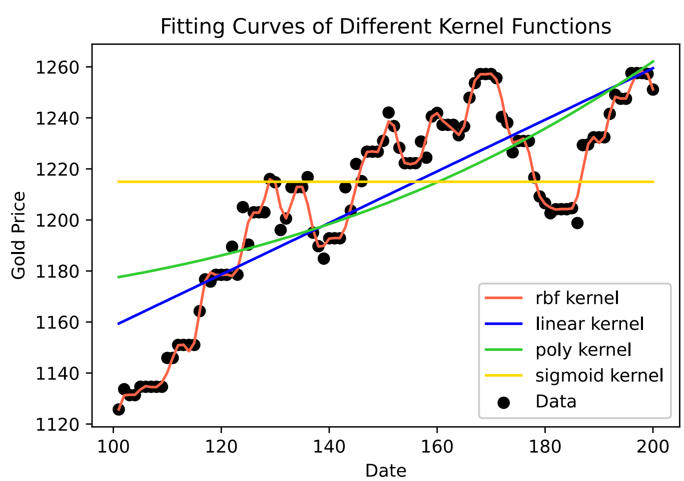


Figure 5-3 Comparison of fitting results

According to our knowledge and experience, we have reason to think that the Gaussian kernel function is the most suitable kernel function, but we still write the program according to the following steps to compare. We will write a program in the following steps to compare the fitting results of the above kernel functions

Step 2: Adopt the Cross-Validation method, that is, when selecting the kernel function, try different kernel functions respectively, and the kernel function with the best fit is the best kernel function.

Step 3: Comparison of indicators (three indicators are made into corresponding histograms and placed on the left side of the above figure), as shown in Figure 5-4

We have selected four commonly used kernel functions, as shown in Figure 5-3, obviously the Gaussian kernel function can be almost perfectly fitted, which is in line with our assumption in the first step, while other kernel functions are far from each other.

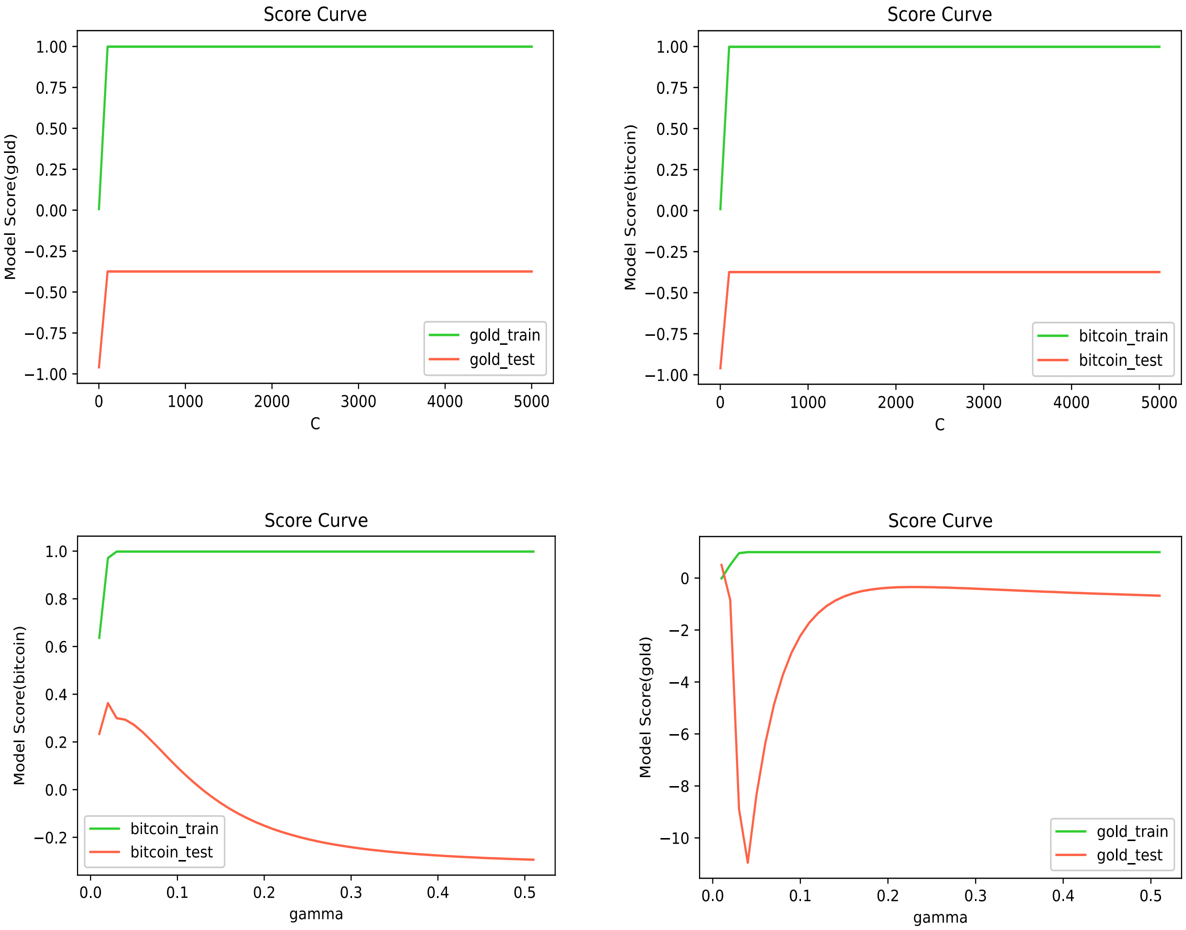


Figure 5-5 Comparison of C and gamma

Now we are going to find the best super-parameter.

- C is the penalty coefficient. The higher C is, the easier the model is to overfit. The smaller C is, the easier the model is to be under-fitting.

- If gamma is set too large, it can easily lead to overfitting. On the contrary, the smaller gamma is, the easier the model is to be under-fitting.

When we use the Gaussian kernel function (RBF) of SVM, if we find low accuracy on training data, we can try to increase C and gamma. But not too large in order to prevent overfitting. So we try to find the best parameters based on the exhaustive method.

As can be seen from the figure, the model fit quite well when C reaches X and the score is pretty the same thereafter. Therefore, we make C 1e3.

For gamma, by taking both the score curves of gold and bitcoin for gamma changes into account, we believe that gamma=0.1 is the best.

The above is the process from the establishment of the best prediction model to the adjustment to the best prediction state. Next, we will use this model to predict the prices of gold and Bitcoin and give the corresponding two indicators (RMSE, NMSE) as Criteria for assessing risk. In the subsequent decision model, the output of the prediction model will be used as the core variable to formulate the optimal trading strategy.

5.2 A Decision Model of Trading Strategy

5.2.1 Model introduction

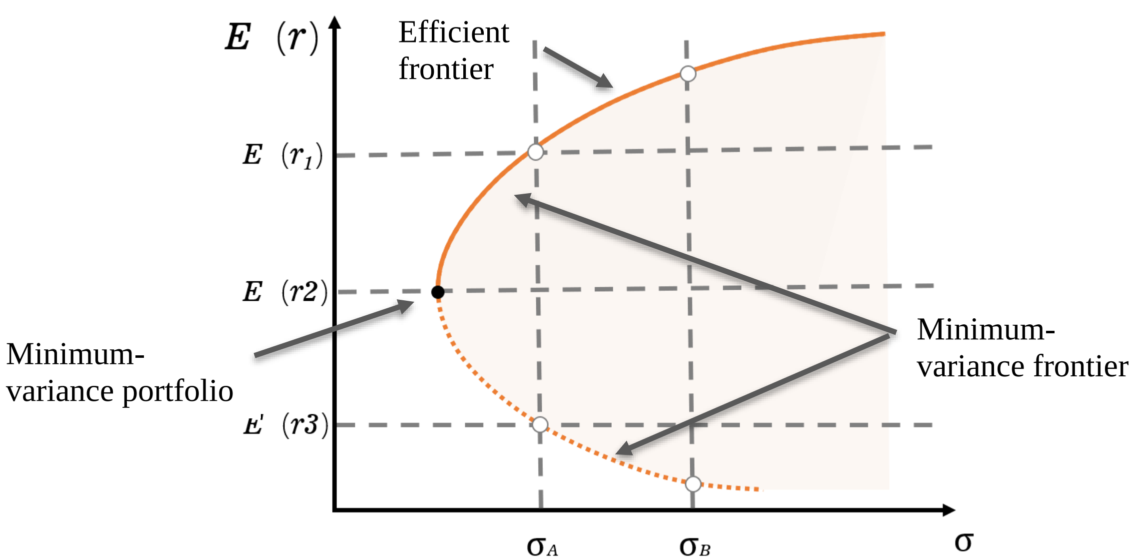


Figure 5-6 the Minimum-variance Frontier of risky assets

The main basis of the decision model is the Markowitz portfolio selection model. Through this model, we realize the trade-off between risk and return of trading strategy.

When allocating multiple assets, we need to consider the expected return of each portfolio and its risks. There are two risky assets (Bitcoin and Gold) in this mission. According to the Markowitz model: first, we need to confirm the risk-return trade-off of the feasible set; then, by calculating the weight of each asset that maximizes the slope of the capital allocation line, we can confirm the optimal risky investment portfolio, and finally confirm the most suitable investment portfolio (including no risky and risky assets). There are also constraints that the actual model needs to take into account. For example, the customer's assets cannot be deficient, the customer has a small expectation of the rate of return, and so on.

First, we need to determine the risk-return opportunities faced by investors through the minimum-variance frontier of venture capital as shown in Figure 5-6. In this step, we need to calculate: the expected return, standard deviation of each venture investment, and the covariance (between gold and bitcoin) between investment types. The expected interest rate can be calculated from the forecast price, and the variance can be obtained from the NMSE and RMSE given by the forecast model.

$$r_P = w_{Gold} r_{Gold} + w_{Bitcoin} r_{Bitcoin}$$

$$E(r_P) = w_{Gold} E(r_{Gold}) + w_{Bitcoin} E(r_{Bitcoin})$$

$$\sigma_P{}^2 = w_{Gold}^2 \sigma_{Gold}^2 + w_{Bitcoin}^2 \sigma_{Bitcoin}^2 + 2 w_{Gold} w_{Bitcoin} Cov(r_{Gold}, r_{Bitcoin})$$

Among them, r and w represent the expected interest rate and investment weight of the corresponding object, and r_P represents the total interest rate, which is obtained by the weighted summation of the interest rate of each venture capital according to the investment weight w.

The minimum variance portfolio is obtained by calculation. If the following portfolios are below the minimum variance portfolio, the investment method needs to be changed. Using the minimum-variance frontier, we find that there is a portfolio with the same standard deviation but a higher expected return on the frontier. Therefore, we divide the minimum-variance frontier based on this point, and define the part above this point as efficient frontier of risky assets

Second, there are some practical issues that we need to consider before choosing the optimal portfolio to invest in. Customers may be subject to certain constraints. In order to simplify the model, we only propose two qualifications:

- In the process of investment, it is necessary to exclude the situation where the asset position is negative;
- The trader requires each investment to meet the minimum expected return. According to how much gold and Bitcoin change, we assume that the minimum yield required for gold is 5% and that for Bitcoin is 10%.

The above conditions are sufficient to identify a feasible and effective investment area with lower risk in the $E(r_P) - \sigma$ diagram. Due to the five-year trading cycle, we have relaxed the standard of the optimal portfolio. Portfolios at the tangency point between the capital allocation line and the efficient frontier are no longer taken. Instead, invest in this suitable area. When the expected return of the selected portfolio is less than the minimum variance portfolio or deviates too far from the efficient frontier, the risk is considered too high and the transaction is abandoned.

5.2.2 Modeling process

With the trade-offs between risk and benefit provided by the Markowitz model, we further refine the decision-making process as shown in Figure 5-7. The figure mainly includes multiple decision objects including investment types, whether to invest or not, and responses after receiving corresponding signals. The main reactions are:

- Stop trading (end, which means neither buying nor selling, and it executes when the risk is high);

- Buy or sell specified currency (sale or purchase, used for profit)

We decide whether to buy or sell by predicted price given by the model. When the risk coefficient is within the allowable range, we consider buying when we judge that the price will rise. If not, we sell it.

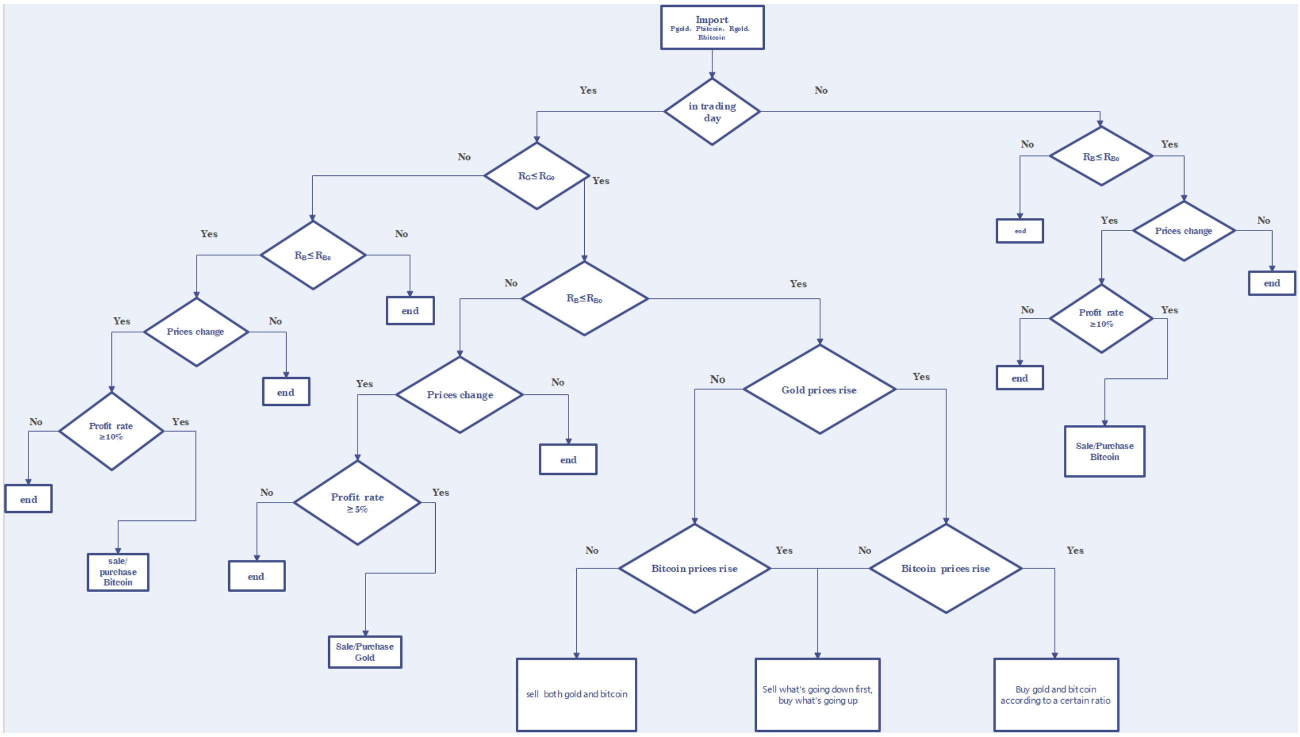


Figure 5-7 Decision Tree

There are two cases to buy currency: one is to buy only one assets, then only need to invest according to the risk and the amount of cash available;

The other is to buy two currencies at the same time. When buying two currencies can make a profit, it needs to invest in proportion according to the forecast interest rate of both currencies.

In addition, when both buy and sell operations need to be performed at the same time, the selling is prioritized and the buying is carried out to ensure that the amount available for buying is larger.

5.2.3 Determination of optimal model& results

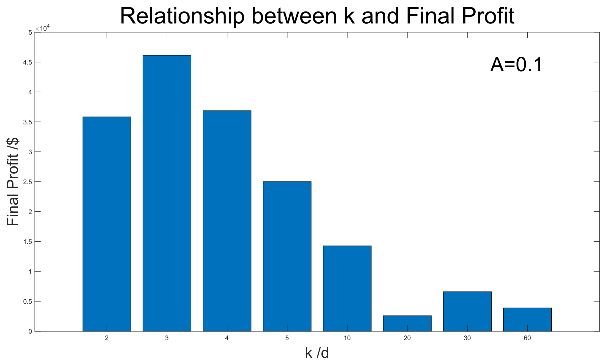
Determination of R_0 (the threshold of R, is used to judge whether the portfolio is far from the efficient frontier):

For day n, R_0's value is the average value of the NMSE of the previous n days. In the case of k less than 10, trading is not carried out for the first 10 days and starts on the eleventh day.

Before the analysis begins, we must make it clear that each person wants to take different risks, so the selection of parameters such as attenuation coefficient at the risk threshold will also be very different. Here we give the result of maximum benefit under certain risk

Confirm k's value:

Under the conditions of the original problem, we change the value of K and A for numerical calculation (where B=1). Finally, the following results can be obtained:



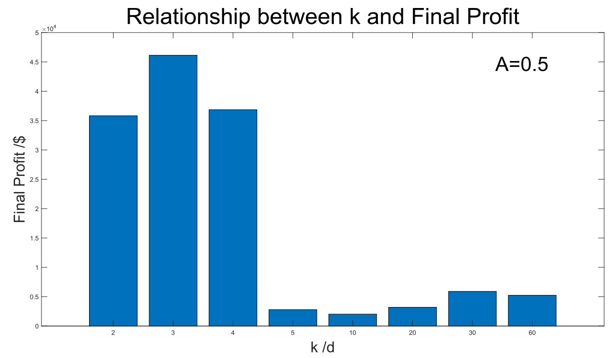


Figure 5-8 Relationship between k and Final Profit

Now we confirm the optimal shielding coefficient:

First, change the value of B and ensure that A=0.1 to calculate the final return, plot as follows:

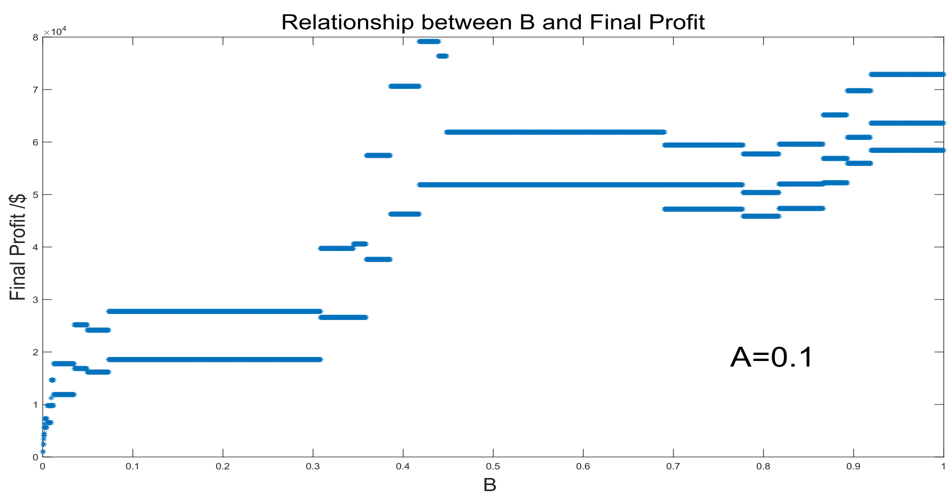


Figure 5-9 Relationship between B and Final Profit

The maximum value is obtained at a point slightly above 0.4. Note: For the two parallel lines in the figure, in fact, the Final Profit oscillates when B grows, and the points are taken densely so that the scatter plot looks like two parallel lines.

Similarly, take B=0.42 and change the value of A to calculate the final income. The plot is as follows:

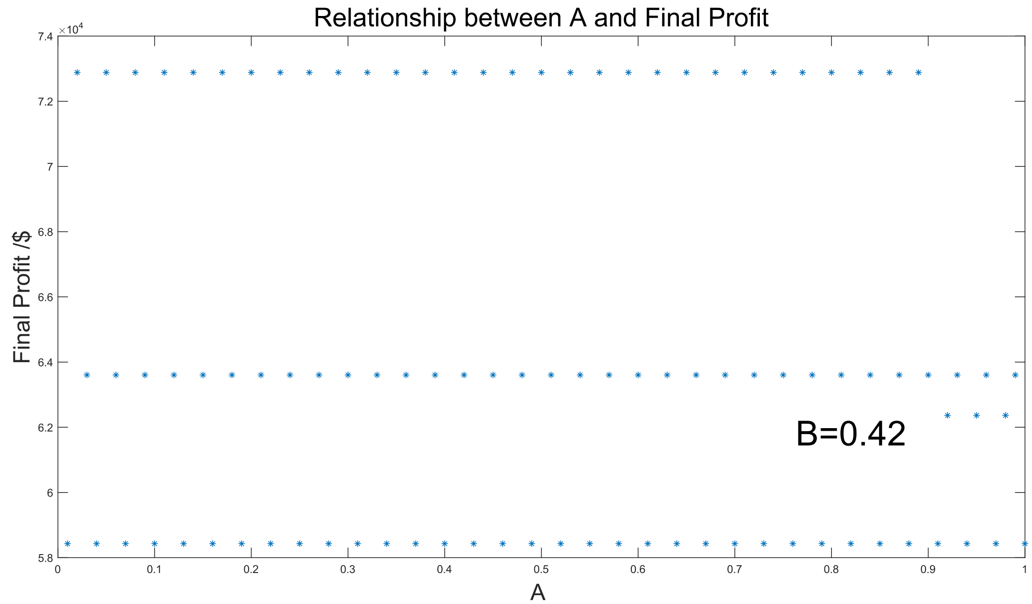


Figure 5-10 Relationship between A and Final Profit

As we can see, Final Profit oscillates on three lines when A changes. Let's take A=0.09 as the maximum.

To sum up, we finally take A=0.09, B=0.42, k=3. With bitcoin and gold expected to return 10% and 5% respectively, the final return is

$79,155.5

6 Sensitivity Analysis

Take α_B =2% and change α_G to see what happens to Final Profits. The diagram is as follows:

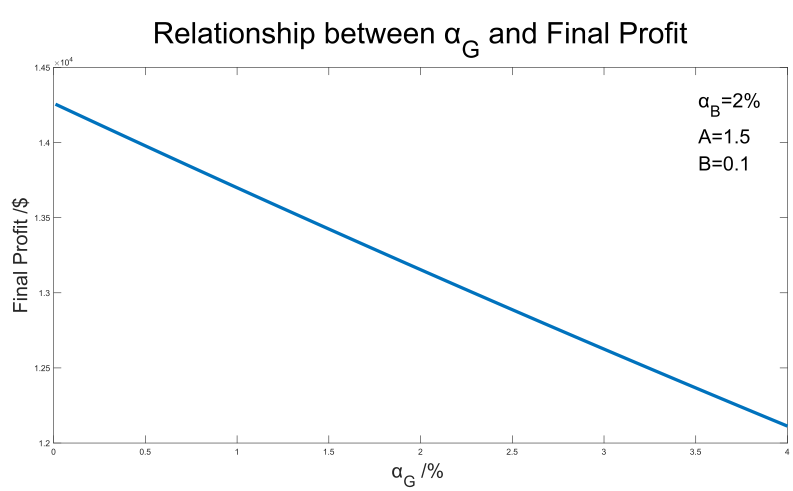


Figure 6-1 Relationship between α_G and Final Profit

Where A=1.5, B=0.1. As you can see, the value of Final Profits goes down as the commission goes up. This is consistent with common economic wisdom -- that higher commissions discourage trading and make trading less profitable.

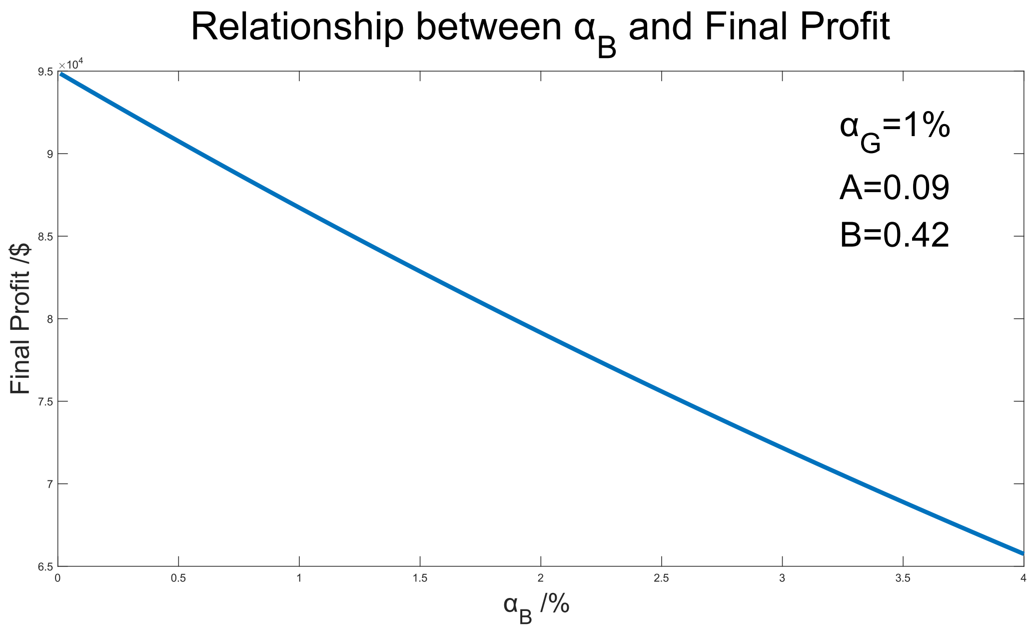


Figure 6-2 Relationship between α_B and Final Profit

In the same way, take α_G=1% and change α_B to see how the change to α_B affects Final Profits:

Where A=0.09, B=0.42. Also, as the commission goes up, the value of Final Profits goes down. That is, higher commissions discourage trading and make trading less profitable.

7 Strengths & Possible Improvements

7.1 Strengths

The model we have chosen is very tailored to the needs of the trader and has considerable sensitivity to adapt to different conditions.

- The Support Vector Regression algorithm used in the predictive model can accurately predict the price and its trend, especially after the optimal kernel function (Gaussian kernel) is determined. Even in the face of gold and bitcoin, whose price changes are very different, the model can still be very accurate in predict the price trend of gold and bitcoin and the corresponding risk coefficient.
- The decision-making model uses the portfolio selection model proposed by Markowitz, the Nobel Laureate in Economics, which can effectively weigh the expected returns and risks, and make correct decisions based on the prediction of the trend of the two volatile assets. the optimal trading strategy. This model can effectively help traders get the most profit during the trading period.

Our model is highly developable. From the perspective of the prediction model, even if the two curves are very different, the SVR can still guarantee a considerable accuracy; from the perspective of the decision model, Markowitz's model is suitable for the selection of a variety of investment portfolios. Further increase, the superiority of the model will be further reflected.

7.2 Possible Improvements

The decision-making part of the model does not use the complete Markowitz portfolio selection model, which is limited to the absence of risk-free investment objects such as bank savings and government bonds. portfolio. But we have also locked our investments in areas with a total of higher returns and minimal risk. With some margin for error, our portfolio can be considered optimal.

We put forward a series of assumptions, one is to simplify the model to ensure that the task is completed within the specified time, and the other is to ensure that the model can run and iterate normally in a long transaction period (about 1,800 days). However, the actual situation is often more complicated, not only to consider the impact of various economic and non-economic factors on the price trend, but also to adjust the model to adapt to different market environments and customer needs. Therefore, if the model is to be further developed, some assumptions need to be relaxed to enhance the adaptability of the model to the actual situation.

To sum up, although our model can perfectly complete this task, it still needs to be further improved when it is put into practical use.

8 Conclusions

We mainly set up two models to complete the formulation of the best trading strategy

For Task 1: We have completed the model. Using the model we built, our initial $1000 will grow to $79,155.5 on September 10, 2021

For Task 2: First of all, our model determines the optimal value of each parameter in both prediction and decision. The final model is used to complete the five-year trading decision. Secondly, our model not only considers how to maximize the benefits, but also fully considers the risks of investment. Therefore, trading strategies can maximize profits while minimizing risks.

For Task 3: In Section 6, we have completed the sensitivity analysis and determined that transaction costs are negatively correlated with final profits.

9 A Memorandum to the Trader

To: Mr./Mrs. xxxx (the trader)
From: Team # 2220297
Subject:
Date: 2/21/2022

In order to complete the task you give us, we have established a complete model system to provide you with the best trading strategy. Investing according to this model can help you avoid many risks and make the best investment plan while forecasting.

On each trading day, our trading strategies will help you select the right volatile assets to invest. When the risk of a volatile asset is too high, we will choose to give up trading to avoid risks. For tradable objects, we decide whether to buy or sell based on the predicted price, thus making a profit. Our model is very flexible. Firstly, it can meet the needs of different traders to avoid risks to different degrees, and the corresponding optimal trading strategy can be obtained by adjusting the risk threshold. Secondly, in sensitivity analysis, we explored that transaction costs are negatively correlated with final profits.

Using our model to invest in gold and bitcoin, from September 11, 2016 to September 11, 2021, our initial $1000 grew to $79,155.5

Our model consists of two main parts. One is the price-predicting model. The other is the decision model responsible for developing trading strategies. The price-predicting model uses SVR with Gaussian kernel, which can accurately predict the price of the day based only on past data and provide corresponding risk indicators. The decision model uses the Markowitz Portfolio Selection Model to weigh the price forecast and risk indicators given by the former to formulate the optimal trading strategy.

The main advantages of our model lie in three aspects:

- Accuracy and stability of prediction
- Scientific decision-making
- Universality and expansibility of the model

Our model parameters have been adjusted to the best state through repeated attempts. Each parameter can be based on theory and a large number of experimental data. It is known that each volatility asset has a different nature. In terms of gold and bitcoin, it is obvious that the change of Bitcoin is faster and the range of change is larger, while the change of gold is gentler. However, both of it can be predicted accurately by our models. It shows that our prediction model has high stability and accuracy. As the theoretical basis of our decision model, Markowitz Portfolio Selection Model can make the decision scientific and optimal. In addition, both SVR and Markowitz's model can show greater advantages when the investment object increases. Further refinement of the model can be used to predict the optimal portfolio for a wider range of volatile assets. For example, the improved model could be used to predict the stock market, and it could examine multiple stocks at once and develop an optimal portfolio.

References

[1.] Ashutosh, S., et al., *Bitcoin Price Alert and Prediction System using various Models.* IOP Conference Series: Materials Science and Engineering, 2021. **1131**(1).

[2.] Choi, Y.-G., J. Lim, and S. Choi, *High-dimensional Markowitz portfolio optimization problem: empirical comparison of covariance matrix estimators.* Journal of Statistical Computation and Simulation, 2019. **89**(7).

[3.] D, M. and L. Z, *Prediction of gold price with ARIMA and SVM.* Journal of Physics: Conference Series, 2021. **1767**(1).

[4.] F, R.M., et al., *Comparison Four Kernels of SVR to Predict Consumer Price Index.* Journal of Physics: Conference Series, 2021. **1737**(1).

[5.] Jian, C., et al., *Structural analysis and forecast of gold price returns.* Journal of Management Science and Engineering, 2021(prepublish).

[6.] Naccarato, A., A. Pierini, and G. Ferraro, *Markowitz portfolio optimization through pairs trading cointegrated strategy in long-term investment.* Annals of Operations Research, 2019. **299**(prepublish).

[7.] Zhenni, J., et al., *The Industrial Asymmetry of the Stock Price Prediction with Investor Sentiment: based on the comparison of predictive effects with SVR.* Journal of Forecasting, 2020. **39**(7).

Appendices

Our Code:

#python code

```
def final_predict(n,k):
    g_days=[]
    g_USD=[]
    b_days=[]
    b_value=[]
    if n<=k:
        for i in range(n):
            g_days.append(i+1)
            g_USD.append(full_g_USD[i])
    else:
        for i in np.arange(n-k,n,1):
            g_days.append(i+1)
            g_USD.append(full_g_USD[i])
    g_days = np.reshape(g_days, (len(g_days), 1))

    if n<=k:
        for i in range(n):
            b_days.append(i+1)
            b_value.append(full_b_value[i])
    else:
        for i in np.arange(n-k,n,1):
            b_days.append(i+1)
            b_value.append(full_b_value[i])
    b_days = np.reshape(b_days, (len(b_days), 1))

    svr_rbf_g = SVR(kernel = 'rbf', C = 1e3, gamma = 0.1)
    svr_rbf_b = SVR(kernel = 'rbf', C = 1e3, gamma = 0.1)

    g_actual=[]
    if n>k:
        for i in np.arange(0,k,1):
            g_actual.append(g_USD[i][0])
        g_actual=np.array(g_actual)
    else:
        for i in np.arange(0,n,1):
            g_actual.append(g_USD[i][0])
        g_actual=np.array(g_actual)
```

```
    b_actual=[]
    if n>k:
        for i in np.arange(0,k,1):
            b_actual.append(b_value[i][0])
        b_actual=np.array(b_actual)
    else:
         for i in np.arange(0,n,1):
             b_actual.append(b_value[i][0])

    svr_rbf_g.fit(g_days, g_actual)
    svr_rbf_b.fit(b_days, b_actual)
    g_predict=svr_rbf_g.predict(g_days)
    b_predict=svr_rbf_b.predict(b_days)

    g_predict_price = svr_rbf_g.predict(np.array(n+1).reshape(-1,1))[0]
    b_predict_price = svr_rbf_b.predict(np.array(n+1).reshape(-1,1))[0]

    g_RMSE = np.sqrt(np.mean(np.square(g_actual - g_predict)))
    g_NMSE = g_RMSE/(np.sqrt(np.mean(np.square(g_actual))))
    b_RMSE = np.sqrt(np.mean(np.square(b_actual - b_predict)))
    b_NMSE = b_RMSE/(np.sqrt(np.mean(np.square(b_actual))))

return [g_predict_price, b_predict_price, g_NMSE, b_NMSE]


svr_rbf_g = SVR(kernel = 'rbf', C = 1e3, gamma = 0.1)
svr_linear_g = SVR(kernel = 'linear', C = 1e3)
svr_poly_g = SVR(kernel = 'poly', C = 1e3, degree = 3)
svr_sigmoid_g = SVR(kernel = 'sigmoid', C = 1e3)
g_actual=[]
for i in np.arange(0,k,1):
    g_actual.append(g_USD[i][0])
g_actual=np.array(g_actual)

svr_rbf_g.fit(g_days, g_actual)
svr_linear_g.fit(g_days, g_actual)
svr_poly_g.fit(g_days, g_actual)
svr_sigmoid_g.fit(g_days, g_actual)
```

```
g_predict_rbf=svr_rbf_g.predict(g_days)
g_predict_linear=svr_linear_g.predict(g_days)
g_predict_poly=svr_poly_g.predict(g_days)
g_predict_sigmoid=svr_sigmoid_g.predict(g_days)

plt.scatter(g_days, g_actual, color = 'black', label = 'Data')
plt.plot(g_days, g_predict_rbf, color = 'tomato', label = 'rbf kernel')
plt.plot(g_days, g_predict_linear, color = 'blue', label = 'linear kernel')
plt.plot(g_days, g_predict_poly, color = 'limegreen', label = 'poly kernel')
plt.plot(g_days, g_predict_sigmoid, color = 'gold', label = 'sigmoid kernel')
plt.xlabel('Date')
plt.ylabel("Gold Price")
plt.title("Fitting Curves of Different Kernel Functions")
plt.legend()
plt.savefig('choose_kernel_function.svg',format='svg')
plt.show()

C_learn_train_g=[]
C_learn_train_b=[]
C_learn_test_g=[]
C_learn_test_b=[]
C_range=np.linspace(1, 5001, 51)
for C in C_range:
    svr_rbf_g = SVR(kernel = 'rbf', C = C, gamma = 0.2)
    svr_rbf_b = SVR(kernel = 'rbf', C = C, gamma = 0.2)
    svr_rbf_g.fit(g_days, g_actual)
    svr_rbf_b.fit(b_days, b_actual)
    C_learn_train_g.append(svr_rbf_g.score(g_days, g_actual))
    C_learn_train_b.append(svr_rbf_b.score(b_days, b_actual))
    C_learn_test_g.append(svr_rbf_g.score(X_test, gold_test))
    C_learn_test_b.append(svr_rbf_b.score(X_test, bitcoin_test))

plt.plot(C_range, C_learn_train_g, color = 'limegreen',label='gold_train')
plt.plot(C_range, C_learn_test_g, color = 'tomato', label='gold_test')
plt.xlabel('C')
plt.ylabel("Model Score(gold)")
plt.title("Score Curve")
plt.legend()
plt.savefig('C Learning Rate(gold).svg',format='svg')
plt.show()
```

%matlab code

```
for t=3:1814
    c1=sum(c3(1:(t+7)))./(t+7);
    d1=sum(d3(1:(t+7)))./(t+7);
    Y(t)=(1+a(t))*Y(t);
    Z(t)=(1+b(t))*Z(t);
    if a(t)==0%
        Y1(t)=0;
        if d(t)<=d1
            if b(t-2)>0&&b(t-1)>0&&b1(t)>0
                Z1(t)=X(t);
            elseif b(t-2)<0&&b(t-1)<0&&b1(t)<0
                k1=nonzeros(Z1);
                if numel(k1)==0
                    m1=0;
                else
                    m1=k1(end);
                end
                if Z(t)>1.1*abs(0.98*m1)
                    Z1(t)=-Z(t);
                else
                    Z1(t)=0;
                end
            else
                Z1(t)=0;
            end
        else
            Z1(t)=0;
        end

       else
       ……

    X(t+1)=X(t)-0.005*abs(Y1(t))-0.01*abs(Z1(t))-0.995*Y1(t)-0.99*Z1(t);
    Y(t+1)=Y(t)+(0.995-0.005*sign(Y1(t)))*Y1(t);
    Z(t+1)=Z(t)+(0.99-0.01*sign(Z1(t)))*Z1(t);
end
```