Problem Chosen	2021	Team Control Number
C	MCM/ICM	2222072

Quantitative Trading Strategy Based on Dynamic Programming and LSTM

Summary

With the flourishing of financial markets and the deepening of the use of electronic trading platforms in financial markets, **quantitative trading** has become the focus of research in the field of financial investment. The investment we carried out is to serve as a reference for trading strategies in the financial trading market, helping investors to adopt the right strategies in the investment process.

First, in order to perform predictive analysis of price trends, after data preprocessing, we apply the **Long Short-Term Memory(LSTM)** model to the previous price sequences to predict the price reference value at the next point in time. Based on the predictive analysis, in order to obtain the optimal decision, we can obtain optimal decisions for all time points before by using the idea of **Dynamic Programming(DP)**. As it is difficult to speculate what the optimal decision will be in the future , we try to construct the model by combining our predicted values and the previous optimal decisions.

It is well known that market price prediction is difficult, and the predicted values even have a negative impact in several condition. Since the model we build cannot predict how much impact the predicted value will have, we build the model as **DP-PV**, time-series prediction strong-dependence model, and **LG-DP**, time-series prediction weak-dependence model.

In the DP-PV model, we directly use the predicted values as the assumed future information to make dynamic planning decisions. In the LG-DP model, we combine the previous prediction information and **economic indicators** using the **LightGBM** model, and train them with the historical optimal decisions as the target, aiming to make the current judgment by imitating the actions of the historical "optimal decision makers".

Immediately after, we fuse the decision chains given by the different models through our **model fusion** step. Due to the existence of various types of risks, A **risk control-based solution** for determining whether to adopt the decision is needed. Thus we can achieve a lower degree of risk and a higher rate of return.

Finally, we perform a reasonable **evaluation** of our model to demonstrate the superiority of our model. The main evaluation is in terms of the return before and after model fusion, the reduction of extreme shocks, and the sensitivity to the amount of money spent on transactions. In terms of the return before and after model fusion, we end up making 98372.65$. In terms of extreme short-term price shocks, we observed the number of trades from the optimal decision and made timely adjustments, effectively reducing the risk level of our investment during extreme shocks. In terms of sensitivity to turnover costs, we found that our model had less impact on a certain level of turnover cost variation.

Keywords: **Long Short-Term Memory, Dynamic Programming, LightGBM**

Contents

1 Introduction

1.1 Restatement of the Problem

To maximize the total return of the market traders, a wise investment strategy is essential. Among those strategies, quantitative trading and generalized algorithmic trading are particularly important.

According to previous reports, as of Q2 2019, the size of U.S. quant funds reached close to 9% of total U.S. equity market capitalization, the share of generalized algorithmic trading in the U.S. is about 75.65%. Therefore, further research on quantitative trading strategies is of great importance.

In this task, we need to tackle the challenges below.

- Determine the best trading strategy given the current data.
- Construct a model that gives out the best daily trading strategy and the best outcome, it should use only the price data up to the day.
- Prove that the strategy given by the model is the best strategy
- Confirm of the way of influence between strategies given by model and trading price.

1.2 Our Work

The following is the flow chart of our work. Firstly, we preprocess the data, then gain the prediction price of assets through LSTM model, from which we get the economic index information through analysis. After that, we will obtain different decisions through the processing of LG-DP model and DP-PV model, get the recommended decisions through model fusion, and finally get the final decision combined with the risk evaluation model.

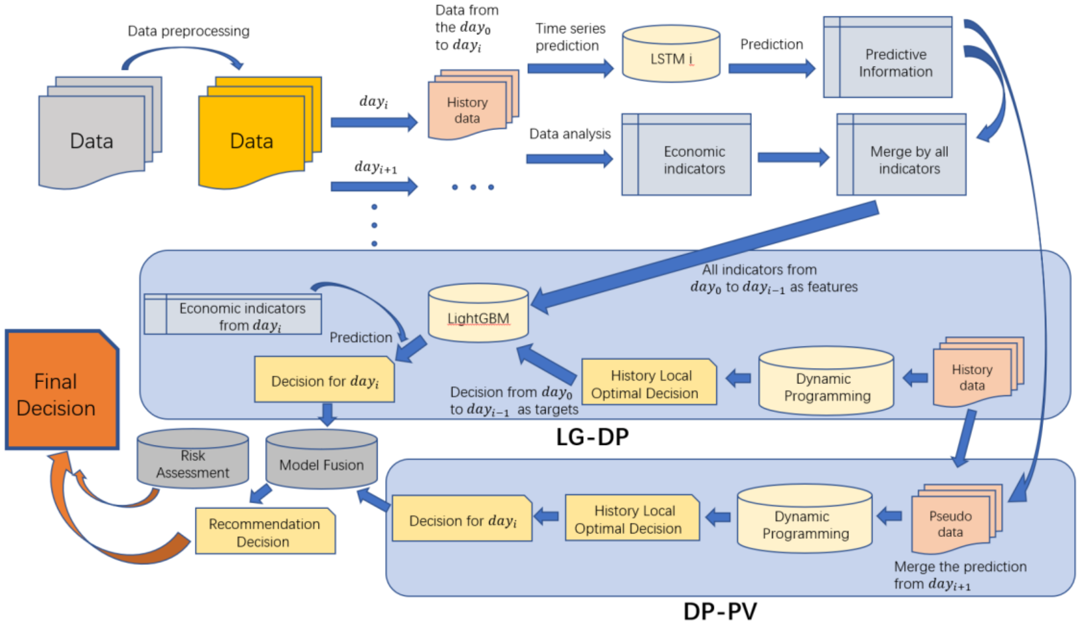


Figure 1:Model Framework

1.3 Assumptions

Before constructing the model, some assumptions need to be illustrated.

Asm. 1. The commission charge are calculated based on the final transaction amount(dollar) of the day, and is deducted only when a transaction is made rather than when the asset is calculated. Also, gold can only be traded when the gold market is open. When it closed, values are calculated based on gold price on the previous trading day.

In current trading rules, the commission and trading rules is as above.

Asm. 2. The price of each currency is fixed each day, trade can be done before the market closes.

This assumption is derived from the description of problem, as only one price for each currency is provided each day.

Asm. 3. External influences on the market in the problem are all attributed to risk.

The market is subject to many external influences during its operation. In current model, we attribute those factors to the parameter of risk.

Asm. 4. The price of gold is generally less volatile than bitcoin.

Based on the data given in the problem, the above conclusion can be established. As a naturally scarce substance, the price of gold is more stable than bitcoin.

Asm. 5. Daily returns should only change with previous currency prices and decisions, and the previous decisions cannot be changed.

This conclusion can be derived from requirements of the problem.

Asm. 6. The price of currency is corelated with time.

This conclusion can be derived from the data given in the problem.

1.4 Nomenclatrue

- **MA**

Moving Average (MA) is a calculation used to analyze data points by creating a series of averages of different subsets of the full data set. We set the MA of m days as MA_m, the price on day_i as $price_i$, the sum of days as n, MA can be calculated as $MA_m = \frac{\sum_{n-m}^{n} price_i}{m}$.[1]

- **BIAS**

Bias Ratio (BIAS) is an indicator used in finance to analyze the returns of investment portfolios, and in performing due diligence.We set BIAS of n days as $BIAS_n$, it can be calculated as $BIAS_n = \frac{price_i - MA_m}{MA_m} * 100\%$.[2]

- **MACD**

Moving Average Convergence Divergency(MACD) is a trend-following mome ntum indicator. In current models, we use MA as a substitute for EMA, as we ca nnot know the future market interval. We use 9,12 and 26 days as the time perio d for MA. We set $diff = MA_{12} - MA_{26}$, $dea = \frac{\sum_{n-8}^{n} diff}{9}$, then MACD can be calculated as $MACD = 2 * (diff - dea)$.[3]

- **PSY**

Psychological Line (PSY) indicator is the ratio of the number of rising periods over the total number of periods. It can be calculated as

$$\mathrm{PSL} = \frac{\text{UP Movements in the last Periods}_\mathrm{n}}{\text{Periods}_\mathrm{n}} * 100$$

- **RSI**

Relative Strength Index (RSI) is a momentum indicator used in technical analysis that measures the magnitude of recent price changes to evaluate overbought or oversold conditions in the price of a stock or other asset.

We set A as the sum of n day's difference between today and yesterday when it is a positive value, B as the sum of n day's difference multiply by -1 when it is a negative value .RSL of n day is RSI_n , which can be calculated as $RSI_n = \frac{A}{A+B} * 100$.

- **UB**

Upper Bound (UB) generally identifies the upper limit of the normal range of stock price fluctuations. When σ represents the standard deviation, $roll$ represents the average of price in the last 30 days, the UB can be calculated as $UB = roll + \sigma * 2$.

- **LB**

Lower Bound(LB) generally identifies the lower limit of the normal range of stock price fluctuations. When σ represents the standard deviation, $roll$ represents the average of price in the last 30 days, the UB can be calculated as $LB = roll\ -\ \sigma * 2$.

2 Price Trend Prediction: based on LSTM model

2.1 Data Preprocess

Before entering data into the model, we conduct our data set from the given data to get better use of it.

- **Data Padding**

  From the dataset given in the question, we can find missing values in the original dataset. The missing data centers on gold price data. To ensure the smoothness and accuracy of the data, the appropriate data needs to be filled.

  In consideration of the range of gold price data, we operated Data Padding Process by *cubic spline interpolation*, which is a mathematical method commonly used to construct new points within the boundaries of a set of known points.

- **Data Standardization**

  In the process of model construction, the presence of odd sample data may cause undesirable effects. To avoid this problem, we operated Data Standardization. Moreover, this operation can also speed up the model convergence.

  In current model, we use Z-score standardization method to map the data into a zero-mean sample space. When μ represents the average value and σ represents the standard deviation, the Z-Score can be calculated using the following formula:

$$Z\text{-}score = (x - \mu)/\sigma$$

2.2 LSTM Model

2.2.1 Model Background

To construct a model which gives out the best daily trading strategy, firstly, we need to propose a model of future return trends that can be built from past data. **Neural Networks**(NN) are essential in this predictive process. We can conclude from **Asm. 6.** that the predictive model we choose should be highly correlated with time.

In that case, we adopt **Long Short-Term Memory** (LSTM), an expansion of Recurrent Nerual Network. Compared with ordinary neural networks, like

- ARIMA, a rather tranditional model that fits to time series data either to better understand the data or to predict future points in the series (forecasting).
- Gray Model, a model that predicts with small amount of input data.

Compared to these models, LSTM has a great advantage with long sequence data, as it controls status of transmission through gates, Giving higher weight to long term memory. It is well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series.

2.2.2 Model Overview

Nomenclature used in this section is listed below, followed by further explanation of it.

Symbol	Description
f_t	Forget gate in LSTM.
i_t	Input gate in LSTM.
o_t	Output gate in LSTM.
$\widehat{c_t}$	New memory cell of $step_t$.
c_t	Final memory cell of $step_t$.
h_t	Hidden layer of $step_t$.
W_f, W_i, W_o	Weight matrix for forget gate, input gate, output gate
b_f, b_i, b_o	Bias for forget gate, input gate, output gate
W_c, W_{out}	Weight matrix for memory cell, fully connected network
b_c, b_{out}	Bias for memory cell, fully connected network

2.2.3 Model Construction

In previous **Asm. 6.**,we conclude that the chosen model would be able to solve the problem of predicting time-series related variables. For this problem, time series are correlated with the price of gold and bitcoin to some extent.

In current model, to obtain sequential predictions for the two currencies, we adopt LSTM to train a time-series model for gold and bitcoin respectively. Suppose we are now at step n and our current data $x_t(0 \le t \le n-1)$ are all derived from historical data, the data to be predicted is x_n. To construct the model LSTM $(x_0, x_1, \dots, x_{n-1}) =$

x_n, we compute the $t_{th}(0 \leq t \leq n-1)$ step of the hidden layer for the LSTM as follows[4]:

$$f_t = \delta\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \qquad i_t = \delta(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$o_t = \delta(W_o \cdot [h_{t-1}, x_t] + b_o) \qquad \hat{c}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

$$c_t = f_t * c_{t-1} + i_t * \hat{c}_t \qquad h_t = o_t * tanh(c_t)$$

After computing n hidden layers, we concatenate all the hidden layers and feed them into a fully connected network to get our prediction $x_n = W_{out}[h_0, h_1, \dots, h_{n-1}] + b_{out}$. While getting the prediction result, we can shift this window of length n one place to the right, which could predict the x_{n+1}. In our experiments, we predict the next 15 data in succession each time as our prediction information.

In the process of model constrution, we use datasets from closely related time period of the predicted time for model training, in order to make predictions for short-term and medium-term trends, while preventing a larger use of the data set from leading to inaccurate results for tests performed later. Taking into account the accuracy of the model and the above reasons, we choose the data set of 50 days before the prediction time point as the training set for prediction.

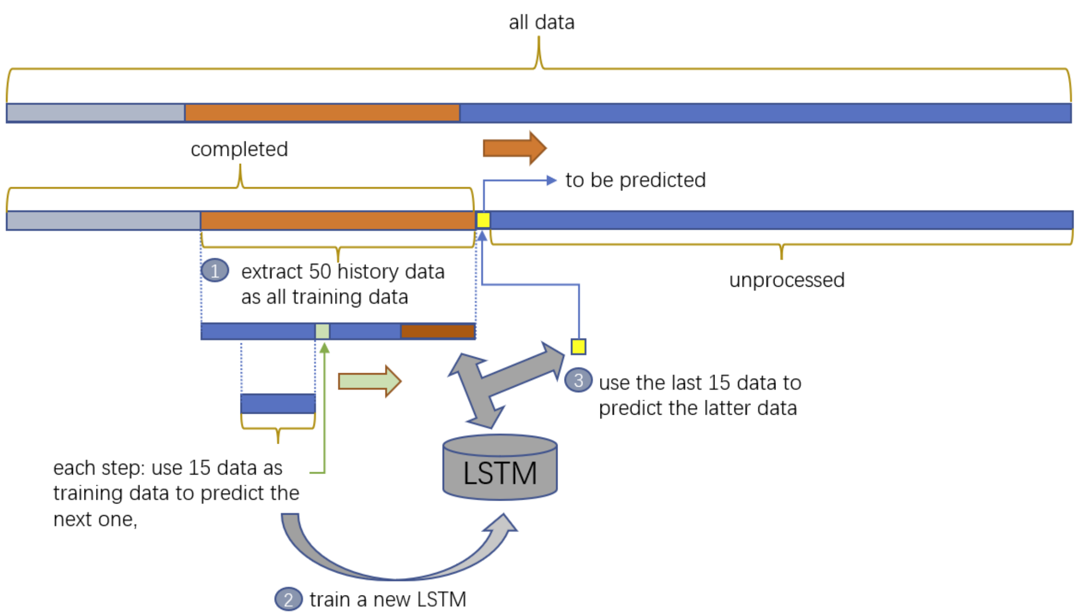


Figure 2: Time-series Prediction Model

2.2 Model Evaluation

After the training is completed, the predicted results for bitcoin and gold are shown in the figures 3 respectively,. There we will use Mean Squared Error(MSE) and R^2 to show the fitting degree of the model. In the predicted results for Bitcoin, the MSE metric reaches 1039086.5751, the R^2 metric reaches 0.9947. In the predicted results for Gold, the MSE metric reaches 291.3900, the R^2 metric reaches 0.9953. This

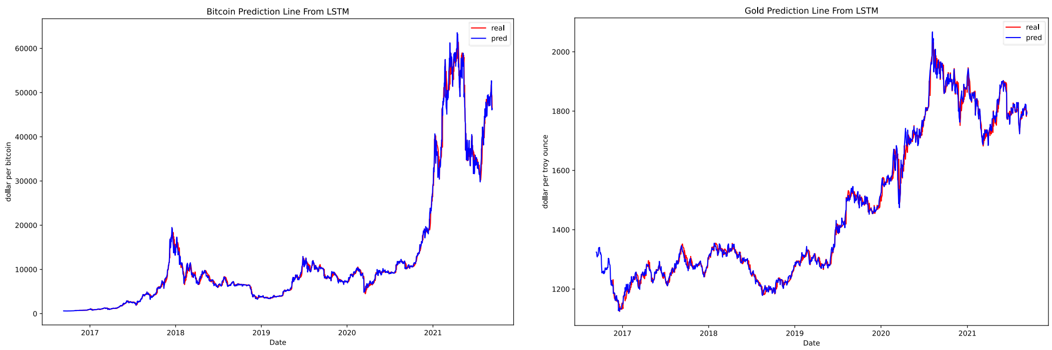


Figure 3: Predicted Results for Bitcoin and Gold

figure and metrics only show information about our model's predicted one-day data

3 Decision model based on local optimal prediction

Based on the existing data set with the price trend forecasting model constructed above, in this phase, a model for making decisions about what to invest in on a daily basis needs to be built. The main structure of the model is as follows.

3.1 Optimal Decision Chain: Dynamic Programming

In the process of model construction, in order to evaluate the benefits of various decisions, we need to obtain the correct decision results and benefits and evaluate them against the current decisions. This operation is performed on the basis of the known daily price data.

The overall optimal strategy can be derived using the **Dynamic Programming(DP)**. This algorithm refers to simplifying a complicated problem by breaking it down into simpler sub-problems in a recursive manner. With this algorithm, we can obtain the optimal value of the historical decisions in the current state and the current optimal decision chain leading to it.

3.1.1 Algorithm Derivation

To ensure the correctness of this algorithm, the algorithm has the following principles。

- **All-choice best principle**: If we know the optimal investment choice, investing all of our money in one product on the same day is bound to yield more than diversifying our money into multiple products.

  Prove the following conclusions by using the method of counter evidence: Suppose there are two types of investments, product *a* and product *b*, respectively. if this principle does not hold, assume that under a certain optimal investment strategy, there must exist j, $i(0 < j < i)$ that allocates a quantity of assets α to product *a* and a quantity of assets β to product *b* on day *j*. On day_i, the final values of these two assets become α' and β' after different investment path. The growth rates of the assets under the two investment paths are as $f(\alpha)=\frac{\alpha'-\alpha}{\alpha}$, $f(\beta) = \frac{\beta'-\beta}{\beta}$.

  - When f(α)=f(β), On the day_j, the total assets will be the same as all of them are invested in any of the product.
  - When f(α)>f(β), the total assets on day_i will be higher if you choose to invest the funds invested in product *b* in product *a* on day_j, which contradicts the assumption that the decision on day_j is the optimal investment strategy. Same when f(α)<f(β).

  To sum up, it can be seen that the all-choice optimal strategy holds

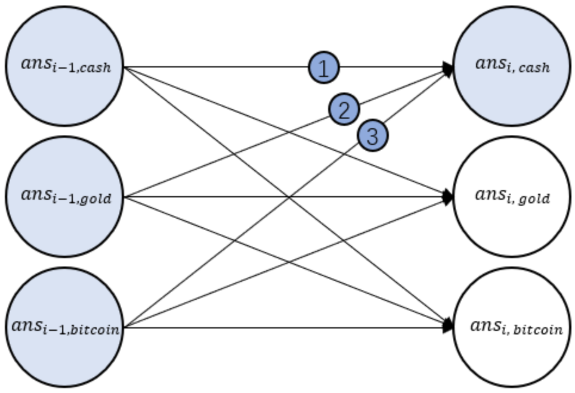
- **Sub-structure Optimization Principle**: As we know the products held under the optimal investment strategy on day_{i-1}, we can calculate the products and the number of products held under the optimal investment strategy on that day from the prices on day_i.

  For the purpose of proof below, we define several parameters that we will use.

Symbol	Description
$currency_i$	The price of the corresponding currency on day_i.
$ans_{i,\text{currency}}$	The maximum amount of the currency we can earn on day_i.
$\alpha_{currency}$	The commission charge of the corresponding currency

Then, for example, when converting to gold, The state transfer equation for dynamic programming is as follows.

According to the Sub-structure Optimization Principle, There are only three possibilities for the total portfolio on day_{i-1} : $ans_{i-1,\text{cash}}$, $ans_{i-1,\text{gold}}$, $ans_{i-1,\text{bitcoin}}$.



① **All the cash is retained**: The final return should be all the cash retained, same as day_{i-1}.

② **All of them are transferred from gold**: The gold is converted to cash according to $gold_i$ and the commission charge.

③ **All of them are transferred from bitcoin**: The bitcoin is converted to cash according to $bitcoin_i$ and the commission charge.

the maximum of the three currency is chosen as the maximum value on day_i.

Since the optimal value of day_i comes from only one choice of day_{i-1}, the decision of the day_i satisfies the functional relationship with the decision of day_{i-1}, the optimal decision chain which composed of decisions of each day can be constructed accordingly. It can be denoted as

$$Benefit_i = \max(ans_{i-1,cash},\ ans_{i-1,gold} * gold_i,\ ans_{i-1,bitcoin} * bitcoin_i)$$

$$ans_{\mathrm{i,cash}} = max(ans_{\mathrm{i\text{-}1,gold}}, ans_{i-1,\mathrm{gold}} * gold_i * (1-\alpha_{\mathrm{gold}}), ans_{i-1,\mathrm{bitcoin}} * \mathrm{bitcoin}_i * (1-\alpha_{\mathrm{bitcoin}}),$$

$$ans_{i,gold} = max\left(ans_{i-1,gold}, \frac{ans_{i-1,cash}}{\frac{gold_i}{1+\alpha_{gold}}}, ans_{i-1,bitcoin} * bitcoin_i * \frac{1-\alpha_{bitcoin}}{\frac{gold_i}{1+\alpha_{gold}}} \right)$$

$$ans_{i,\mathrm{bitcoin}} = max\left(ans_{i-1,\mathrm{bitcoin}}, \frac{ans_{i-1,\mathrm{bitcoin}}}{\frac{\mathrm{bitcoin}_i}{1+\alpha_{\mathrm{bitcoin}}}}, ans_{i-1,bitcoin} * bitcoin_i * \frac{1-\alpha_{bitcoin}}{\frac{gold_i}{1+\alpha_{gold}}} \right)$$

Using the above method, a historical decision chain based on the maximum profit obtained on that day as a result can be calculated on any day i using the data before that day. Although we cannot modify our historical decision by observing the current optimal decision chain, we make reference to the current decision by deducing and learning its decision-making means

In particular, we calculated the theoretical optimal effect of the investment based on five years of data with dynamic programming: an initial $1,000 could end up with $337,021,69703.18.

3.2 Decision Model based on Price Trend Prediction

Based on the previous optimal decision chain model, with the Price Trend Prediction model described in the previous section, it is possible to build a series of **decision models** that accept the output of the LSTM model.

Based on the Price Trend Prediction results, with a series of improvements, we built several types of decision models as follows.

3.2.1 DP-PV model: Strongly Dependency Model

In this model, we receive the predicted value output from the LSTM model and use it as the settlement price for the day, thus generating the optimal decision chain for the initial model building. Nomeclature used in this section are listed below.

Symbol	Description
D_i	All data up to day_i
$History_i$	All previous data by day_i, $History_i = \{v_{gold}(p) \mid 0 \leq p < i\}$
$Pred_i$	The data of day_i output by the LSTM model, $Pred_i = \{pred_{gold}(i), pred_{bitcoin}(i)\}$

In this modeling process, the input $Pred_i$ generated by the LSTM model is combined with the $History_i$, which consists of previous data, as today's data D_i. This data is fed into the *Optimal Decision Chain* model to obtain the optimal decision($DP - PV_i(D_i)$) with taking LSTM's prediction as today's data, and its corresponding best decision chain(choice(DP-PV_i)).

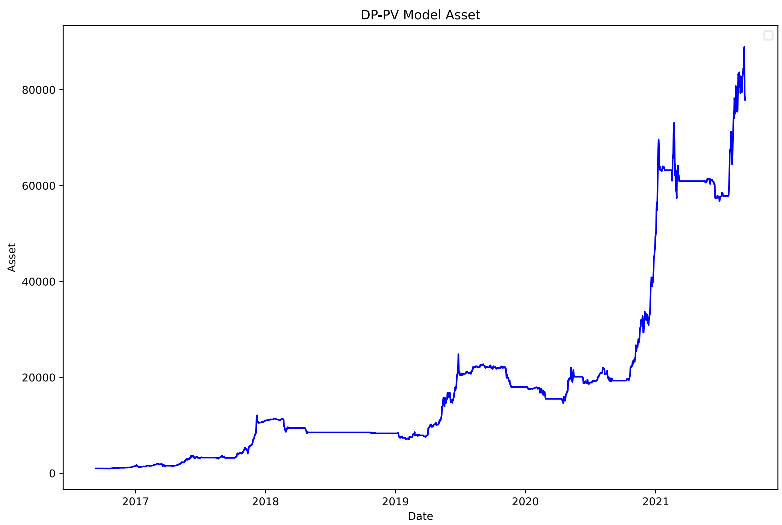


Figure 4:DP-PV Model Asset

3.2.2 LG-DP model: Weakly Dependency Model

In the previous DP-PV model, we established the link between the predicted values and the current decision. However, since the previously trained LSTM model may have inaccurate predictions, in order to reduce the impact of this type of problem on the final results, we build a second model which uses the predicted values with various financial indicators for the generation of the optimal decision chain.

Some of the economic indicators used in the model are as follows:

Symbol	Description	Time Period(if exists)
MA	Average value of a last time period	6,12,24,38,57,127
$BIAS$	Indicator to analyze the returns of investment portfolios	6,12,24,38,57,127
$MACD$	A trend-following momentum indicator	
PSY	The ratio of the number of rising periods over the total number of periods	
RSI	An indicator that measures the magnitude of recent price changes	6,12,24
UB	identifies the upper limit of the normal range of stock price fluctuations	6,12,24,38,57,128
LB	identifies the lower limit of the normal range of stock price fluctuations.	6,12,24,38,57,128
Pred	Next day's earnings increase compared to today	
avg5	The increase of price today compared to last 5 days	5
avg15	The increase of price today compared to last 15 days	15
more15	Forecast for the number of days within 15 days will have higher prices than today	15
Less15	Forecast for the number of days within 15 days will have lower prices than today	15

In this model, we calculate the above metrics separately for the historical data and the predicted data output from the LSTM model.

Based on the previous information, we can collate the changes in indicators at each previous time point with the optimal decision chain for that time point. We can learn from the *All-choice best principle* int the generation of the optimal decision chain that the optimal decision problem can be considered as a triple-class classification problem (cash/gold/bitcoin)at each time point, thus we can try to simulate the classification of the optimal decision chain at the current time node through a machine learning model.

Symbol	Description
$K\text{-}Gold_i$	The all indicators of day_i for gold
$K\text{-}Bit_i$	The all indicators of day_i for bitcoin
T_i	The set of optimal decisions of day_i ,$T_i = \{T_p \mid T_p \in \{0,1,2\}, 0 \le p < i\}$
K_i	The all indicators for gold and bitcoin up to day_i, $K_i = \{K\text{-}Gold_j, K\text{-}Bit_i \mid 0 \le j < i\}$
$\hat{T}_j^t$	The predicted value for day_j of the model in round t
$f_t^p(\cdot)$	The value of the p_{th} order derivative of the decision tree function
Ω	Function Remainder
Obj(t)	The objective function of the t_{th} round
Constant	Constant value

Suppose currently at day_i, we get a model that satisfies $LG\text{-}DP(K_p) = T_p$ $(0 \le p < i)$, then we could feed today's all indicator information K_i into the model to predict the decision T_i.

Given our large number of indicators, we chose **LightGBM**, a model that can quickly handle a large number of features, to construct the current phase of our model. It is a gradient boosting tree that expresses the residuals approximately by the Taylor expansion of the loss function, mainly using the information of the first-order derivatives and second-order derivatives, and making the model more accurate and efficient by the level-wise iteration, sampling method, sorting method, and cache optimization. The brief objective function of LightGBM is as follows:

$$obj(t) \approx \sum_{j=1}^{n} \left[l\left(T_j, \hat{T}_j^{t-1}\right) + g_j f_t(K_j) + \frac{1}{2} h_j {f^2}_t(T_j) \right] + \Omega\left(f(t)\right) + Constant$$

, where $g(\cdot)$ and $h(\cdot)$ denote the corresponding first-order derivative coefficients and second-order derivative coefficients, respectively.

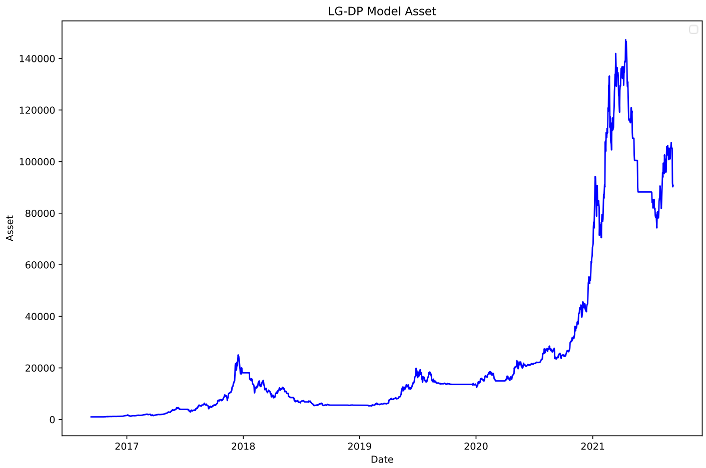


Figure 5: LG-DP Model Asset

3.2.3 Model Fusion

In our above model construction process, we constructed two different models: the *DP-PV model* and the *LG-DP model*. Similarly, in the actual model application, we may use a wider variety of models. In order to use different kinds of models and aiming to ensure that the final result maximizes revenue and minimizes risk, a multi-model composite decision framework is necessary.

We use the following strategies to fuse the models. When the decisions given by the two models are consistent, we follow their choice; When the model decisions are different, we consider the maximum growth range($G_{DP\text{-}PV}$&$G_{LG\text{-}DP}$) and fall range($F_{DP\text{-}PV}$&$F_{LG\text{-}DP}$) of the two models in recent 100 days. We will prefer the model with more increase and less decrease. Therefore, we can get the following formula:

$$\delta = \theta * e^{-\frac{G_{LG-DP}}{G_{DP-PV}}} + (1-\theta) * e^{-\frac{F_{DP-PV}}{F_{LG-DP}}} \qquad (\theta = 0.62)$$

When δ is greater than δm=0.3, we prefer to believe in DP-PV model; On the contrary, we prefer to believe in LG-DP model.

When making choices, if the strategy of the previous day is different from that of today, we will transfer $\mu 1 = 0.65$ of the assets; If the strategy is the same, we will transfer $\mu 2 = 0.2$ of the assets, so as to make the decision more stable and smooth and reduce the impact caused by sharp fluctuations in value.

3.3 Risk-Based Evaluation for Decision Making

Based on decision models above, we can obtain the decision information for day_i. However, the decision given by the current model relies on the local optimal solution obtained by *dynamic programming*, and the local optimal solution does not necessarily represent the global optimal choice.

In order to obtain a better global decision, we also have to decide whether and how to adopt the prediction results of the model according to the *Risk-Based Evaluation* strategy.

3.3.1 Model Risk Evaluation: Recommender Strategy

During the model decision process, there are cases when the market fluctuates frequently, leading to frequent changes in the local optimal solution. In that case, the decisions given by the model may switch frequently, resulting in lower final returns. To solve this problem, a risk evaluation of the decisions made by the model is required.

We set the prediction decision chain used on day_i as {choice($model_i$)}, which is the chain of decisions from day_0 to day_i based on the local optimal solution on day_i, abbreviated as C_i.

We can consider $model_i$ as a stock recommender whose decision chain can generate the most returns on day_i. As the daily prediction model changes, the stock recommenders will vary from day to day and the previous decisions they made will change, What we can be sure of is that if the stock recommenders make the same decisions over consecutive days (e.g., the optimal decision recommends choosing Bitcoin for 15 consecutive days), then we are more likely to trust a recommender who has made the correct decision in recent consecutive days.

This leads us to derive the model risk assessment scheme below:

We only make the same choice when the current prediction decision chain {choice($model_i$)} makes the same choice in consecutive D days.

3.3.2 Market Risk Evaluation

A solution based only on model risk evaluation causes a certain delay in judging the market, mainly in the onset of an up trend and the delay in the onset of a down trend. Therefore we need to optimize our decision scheme for these two delays.

This delay is mainly related to the value of our D.

- When D is small, although the delay generated is also small, it also raises the risk of being misled by decisions that locally correct, while globally wrong. Also, in the case of high short-term price volatility, it is easy to generate high commission charge due to frequent trading.
- When D is large, it will directly avoid short-term dramatic price fluctuations. Its bias is towards very conservative long-term investments, while insensitive to long-term declining bear markets.

This leads to the introduction of a market risk evaluation strategy, takes into account factors such as sharp short-term price fluctuations, frequent trading, and trends in prices, and makes dynamic adjustments to D and other changes in decision making.

When analyzing price fluctuations, the model tends to avoid drastic price declines when calculating the local optimal decision chain, however, this will also greatly increase the number of trades in the model in the short term, so we introduce the model's recent trade count *Trans*. This indicator characterizes to some extent the drastic price fluctuations in the recent time period N_{day}, and when the model's short-term trade count reaches a certain threshold, we consider that during this period When the number of short-term trades of the model reaches a certain threshold, the price has generated repeated fluctuations in this period of time.

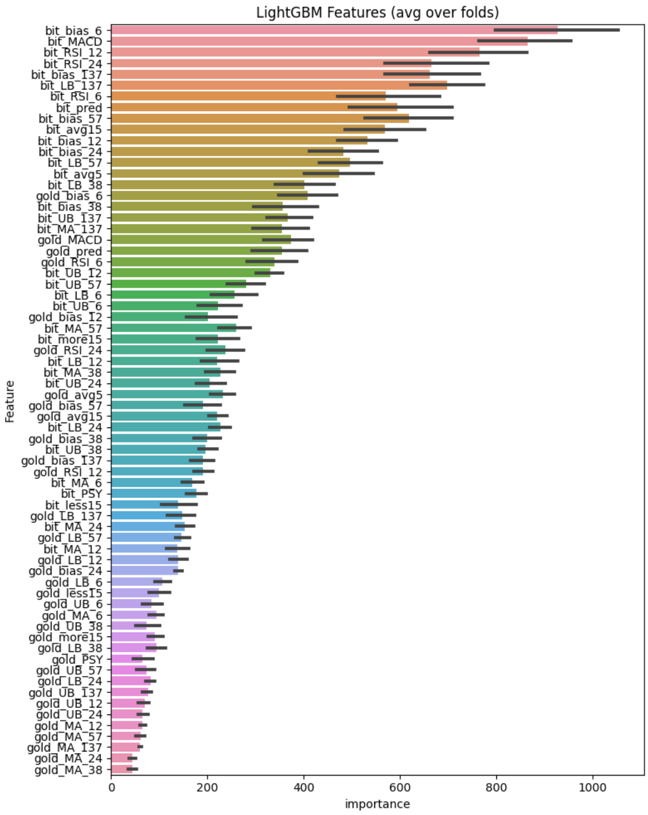


Figure 6:LightGBM Feature Importance

The average continuous holding time $T = \frac{N_{day}}{Trans}$ of the model decreases as the number of short-term trades increases, i.e., the reasonable number of trades over time changes under different price change trends. Therefore, we need to adjust the size of D dynamically, so that our decision can capture the changes of the model more sensitively when the price changes drastically. We can get D=min$\{D_{max}, \frac{N_{day}}{Trans}\}$.$D_{max}$ is the max value we set for a period of time.

For a more accurate market risk evaluation, we use a number of financial indicators to make predictions about price trends. In the above figure 6, the MACD indicator and the 6-day deviation ($bias_6$) indicator have the best indication in the joint decision of gold and bitcoin, so we use these two types of indicators to guide the market risk evaluation.

In conclusion, we can get the market risk evaluation strategies:

If the MACD indicator and the $bias_6$ *indicator are both greater than 0 and the prediction for tomorrow is also increasing, we will buy the relevant product.*

If one of the MACD indicators or the $bias_6$ *indicator is less than 0 and the prediction for tomorrow is also decreasing, we sell the relevant product.*

The market risk evaluation allows us to correct, to some extent, for the delay in our observation of the locally optimal decision model. Decisions based on risk-based evaluation composite model are as follow.

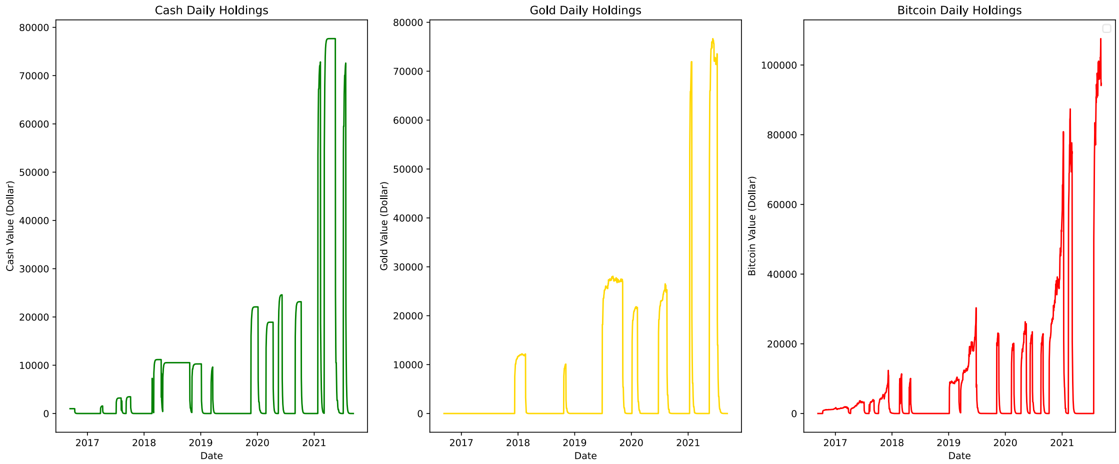


Figure 7: Assets' Daily Holdings

4 Model Evaluation and Verification

In this part, we will compare and evaluate DP-PV, LG-DP and the Multi-Model, mainly from three aspects: revenue progress under model fusion, reduction of tip fluctuation and sensitivity to changes in transaction costs.

4.1 Revenue Progress under Model Fusion

The assets on different dates under the three models are evaluated in this part and the portfolio of the day is used to convert the assets into US dollars for evaluation.

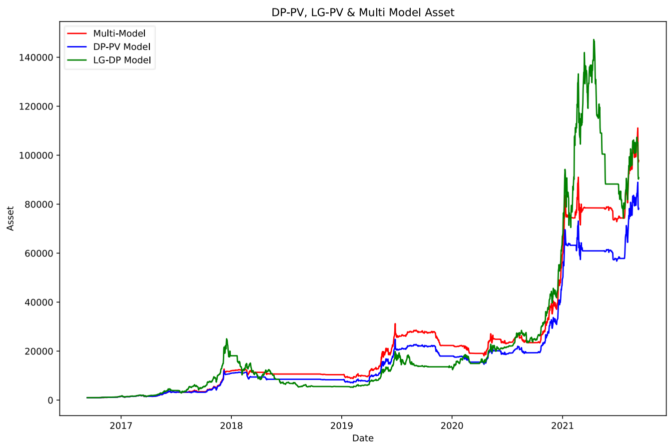


Figure 8: DP-PV, LG-DP & Multi Model Asset

It can be seen that the assets of LG-DP model change violently. Although there is a high value at the peak when bitcoin's price improves greatly, it then falls rapidly in value fluctuation. Finally, the assets rank second, at 90783.05; The assets change of DP-PV Model is relatively gentle without sharp fluctuation, while, in the other hand, it missed many growth opportunities, resulting in the final result of 78300.67, ranking third; The Multi Model combines the advantages of the two. Its overall trend is relatively flat. In case of large fluctuations, it can make reasonable investment and is capable of stopping losses in time. At the same time, it can also grasp the opportunity to enlarge its assets. The final asset is 98372.65, ranking first. It is evident that DP-PV and LG-DP have their own characteristics, and the fusion model successfully selects the advantages and gets better prediction results.

4.2 Reduction of Tip Fluctuation

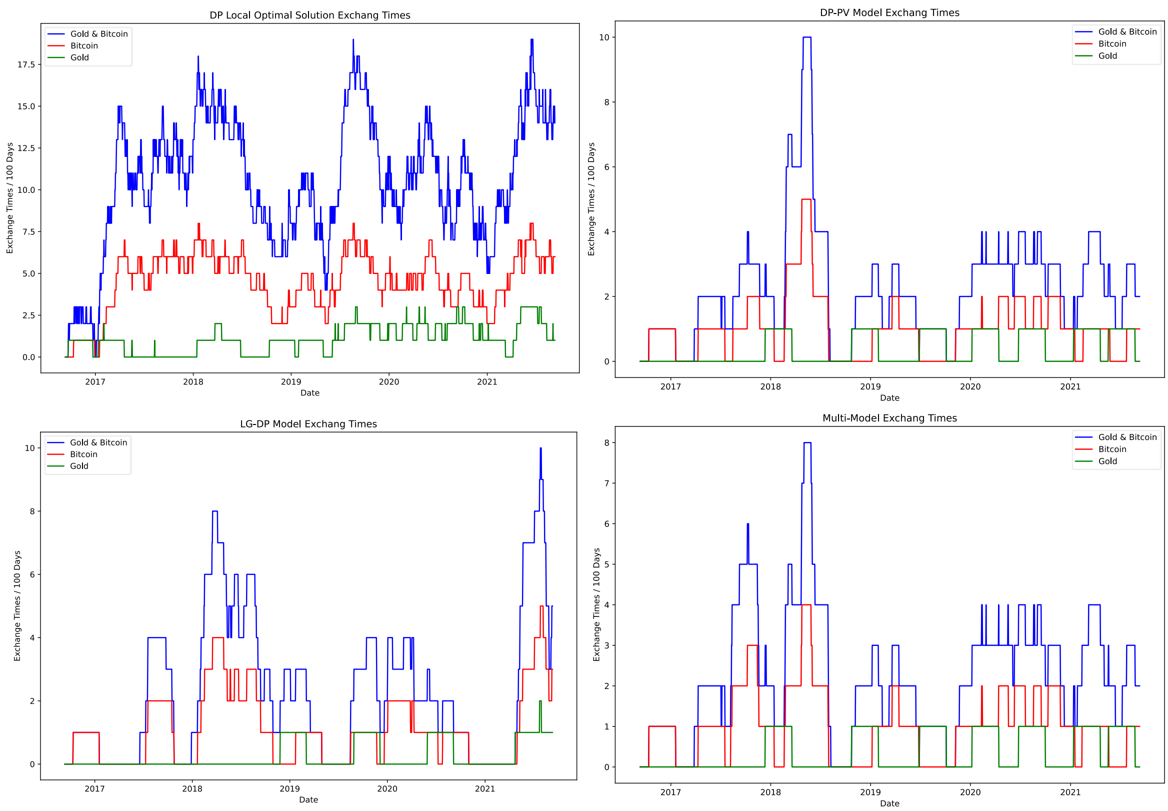


Figure 9:DP Decisions and Models' Exchange Times

Tip fluctuation indicates the change of trading times in a short period. In Dynamic Programming model, the higher its value is, the more volatility it will be. Generally speaking, it is difficult for a normal model to accurately judge the rise and fall, let alone make good investments . From the decision fluctuations of the three models over the past 100 days, we can see that the local optimal decision made by DP has more tip fluctuation situation because it is greatly influenced by fluctuations, and the number of changes within 100 days is high and drastic, not stable enough. the DP-PV model has less overall fluctuations, only a higher peak, relatively flat and stable. the LG-DP model fluctuates more frequently in a short period of time. However, when compared to the DP prediction, the overall is more smooth because it uses decision trees for decision making.

4.3 Sensitivity to changes in transaction cost

We adjust the transaction cost ratios to check the impact of the transaction changes on the model to see whether our model can maintain smooth returns under different trading environments. To ensure equality between the two products, we adjust the transaction cost ratios for both currency at the same intervals and do not make the transaction amounts too large. At the same time, we re-inferred our model with the same parameters for each adjusted transaction cost.

Specifically, we tested the model a total of 36 times from 0.0 to 0.025 for both products, adjusting each time by an increase of 0.005.

The graph above is a heat map of the prices after we adjust the data. The brighter the color of the picture, the more money is made at that volume.The least amount of money was made when the bitcoin trade cost 1.5% and the gold trade cost 2.5%, with a total gain of 61,258.13. The highest amount of money was made when the bitcoin trade cost 1.5% and the gold trade cost 0%, with a total gain of 246,139.79. What we can see from the heat map is that in the case of the bitcoin trade cost 2.5%, or when the gold trade cost is 0%, the overall gain is higher. However, the fluctuations are smoother in the other ranges.

This shows that our model is less influential in the range of 0.5% to 2.5% for gold transactions and 0% to 2% for bitcoin transactions. However, it is greatly influenced when the gold transaction amount is small or the bitcoin transaction amount is large.

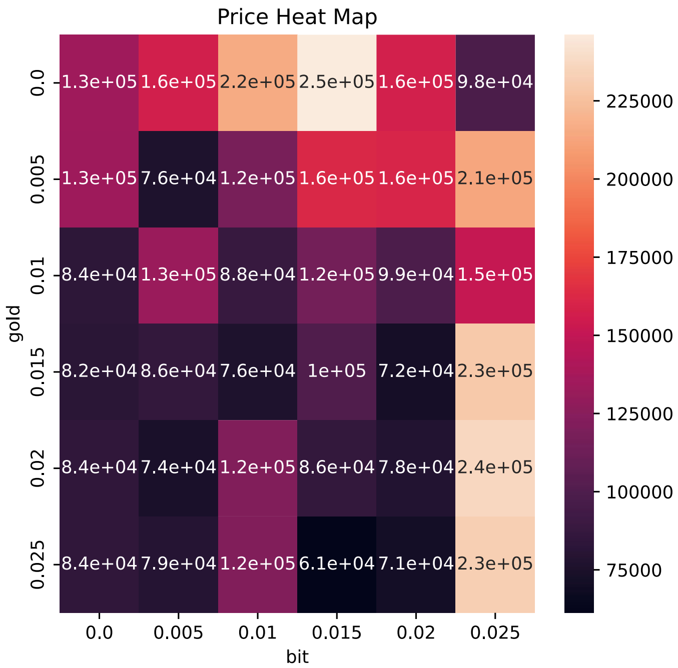


Figure 10: Heat Map of Price

Look deeper into the reasons for this, we believe that since our main model revolves around the local optimal solution generated by dynamic programming, it is difficult for our decision-making solutions to draw on when the local optimal solution generates a lot of difficult to understand operations, such as selling the day before a drop and buying the day before a rise, and when trading is very frequent.

When gold is traded in small amounts, it can be a good hedge funding platform for bitcoin, so it's easy to draw on for our decision-making options on whether to hold gold for hedging. And at times when bitcoin transactions cost more, it is also easy to absorb for our decision scenario as the local optimal solution wants to harvest more money and may store it in bitcoin Over a longer period of time. Contrary to that, at times when bitcoin transactions are small and gold transactions are large, the trading strategies of the local optimal decision are very frequent, which makes our risk evaluation model consider it extremely risky and often does not draw on the operations in it.

MEMORANDUM

To: Traders

From: Team 2203695

Subject: Prioritizing Research on Strategy of Quantitative Trading

Date: February 22, 2022

Dear Traders, we are proud to present to you the results of our quantitative strategy research. We have developed a mathematical model that can make optimal investment decisions to a certain extent.

Strategise of the Model

The model we developed combines the inferred historical optimal decisions using DP with the price predictions obtained by the LSTM，combines them with a risk evaluation strategy for investment decisions. To start with, LSTM model and DP were combined as follows.

- **DP-PV model**: Time-series prediction strong dependency model

  This model mainly takes the approach of giving the predicted values to DP directly for inference. Simply speaking, this model is like a very powerful decision maker who makes a decision after predicting the future information. This model will mainly be influenced by the accuracy of the future forecast information.

- **LG-DP model**: Time-series forecasting weakly dependent model

  This model mainly takes the way of learning from previous economic indicators, forecasting information and optimal decision information to simulate the current decision. Simply speaking, this model is like a person learning from the most powerful decision maker at present, and imitating his means to make a decision. This model is mainly influenced by some economic indicators and the accuracy of the model.

Then, we combine the two models' decisions by judging economic indicators like their upside and short-term maximum downside indicators to make the best possible decision. As it’s hard for us to achieve accurate forecasts and accurate model judgments, also the local nature of the optimal decision will produce a certain delay at this point in time for a truly correct decision in the future, we cannot have a blind trust in the model's decision. Therefore, from model metrics, economic metrics to asset balance, we introduce risk evaluation to adjust our decisions on-the-fly to minimize risk and maximize return.

Evaluation of the Model

To give our model more credibility, we evaluated our model in three main aspects and proved its superiority.

- **Return** Before and After Model Fusion
  Comparing each single model before model fusion with the model after model fusion, the experiment proves that the model is superior after model fusion.
- **Reduction** of Extreme Fluctuations
  Extreme Fluctuations indicate the change in the number of trades in a short period of time, and the higher the value in a DP model, the more volatile a commodity is in the short term, and it is difficult for a normal model to accurately determine the ups and downs in this investment. Therefore, we control for the reduction of extreme shocks to enable our model to hedge the risk in times of high price volatility.
- **Sensitivity** to the amount spent on transactions
  As can be seen from the experiments, our model is relatively unaffected by the small range of variation in trading volume, and the final return is almost controlled by a variation of about 1.5 times in the small range of variation.

Results of the Model

In the end, our model went from \$1,000, to making \$98,372.65, a nearly 100x increase in 5 years. At the same time， we were able to basically hedge instantly during each period of large changes in our model in Bitcoin, and buy instantly when it started to have big rallies again. We were also able to maintain a largely solid upside through asset allocation during the time of immediate hedging. To conclude, our model has been very profitable in general.

All of our strategies, models, and results are described above, thank you for reading. Hope our models are practical for you.

Sincerely,
Team #2222072 of 2022 MCM

References

[1] What Is a Moving Average (MA)? https://www.investopedia.com/terms/m/movingaverage.asp

[2] Wikipedia. Bias ratio https://en.wikipedia.org/wiki/Bias_ratio

[3] What Is Moving Average Convergence Divergence (MACD)? https://www.investopedia.com/terms/m/macd.asp

[4] Natural Language Processing with Deep Learning http://web.stanford.edu/class/cs224n/