Gold Price Prediction

LO Wai Leuk, TSANG Yu, YAO Zhi Xing, YUE Yu Yang

***Abstract*—We present the time series of daily gold price from Year 2004 to Year 2024, explore the serial correlation of the time series, investigate different models to fit the time series and make predictions, and finally discuss the results and conclude.**

I. Introduction

Gold is widely considered a valuable asset and is used to hedge against inflations. Over the past several decades, there is a trend of inflation and hence the gold price has been rising steadily as well. On the other hand, gold is a volatile commodity in the short term, whose price is easily affected by gold supply, interest rate, etc., making the price difficult to predict. In [1], a demand-and-supply framework was proposed to study the relation between inflation and gold price, and a linear model is applied to predict the price. In [2], several models were tested for their predictive power of the gold price, including the ARIMA model, OLS model and hidden Markov model, and it was found that ARIMA and OLS do not perform well while hidden Markov model could capture the market mood to some extent.

In this project, the gold price dataset from Year 2004 to 2024 is explored for its features, and several models, both linear and non-linear, are investigated for their predicting power of the price. The remaining parts of the report is organized as follows: in Section II, the dataset is discussed in details, where standard features such as auto-correlation and residuals are calculated and visualized. A solution overview is also provided, where each model is briefly introduced. In Section III, various models are introduced for their basic idea, architecture and mathematical formulation. Following that an evaluation is conducted in Section IV via a metric called the mean absolute percentage error (MAPE). Finally the findings and lessons learned are summarized in Section V.

II. Dataset and Solution Overview

The daily gold price dataset is downloaded from Yahoo Finance and contains the date, adjusted close price, close price, daily high, daily low, open price and trade volume of each day, ranging from 1 Nov 2004 to 30 Oct 2024. As can be seen from the Figure 1, the daily price is not stationary, which can further be verified by the unit-root test as shown in Figure 2.

The log return of the daily price is stationary and exhibits no auto-correlation, as shown in Figure 1. However, the log return is not white noise and has ARCH effect, which is discussed in more details in Section III-A, and it motivates the use of a GARCH model as a baseline for this project.

An iGARCH model is selected as the baseline, and several other models are compared against it. One of the models tested is the hidden Markov model, a non-linear model with hidden

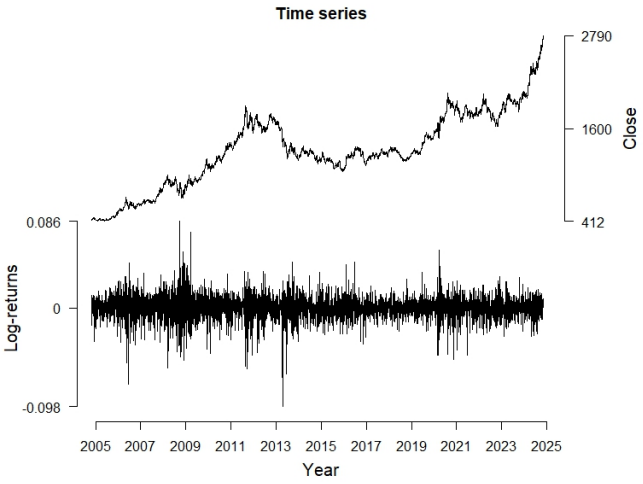

Fig. 1: Gold Price and Log Return

```
> unitrootTest(data_train$price,lags=1,type=c("c"))

Title:
 Augmented Dickey-Fuller Test

Test Results:
  PARAMETER:
    Lag Order: 1
  STATISTIC:
    DF: 0.306
  P VALUE:
    t: 0.9787
    n: 0.9709
```

Fig. 2: Augmented Dicky-Fuller Test for the Price

variables controlling the regime shift. Teammates please supplement your model overview.

III. Methodology and Results

In this section, different models are introduced and their mathematical background are discussed. They include the iGARCH model, the hidden Markov model, and Teammates please supplement your model names.

A. iGARCH Model

The log return of the price is modeled by the following general formula:

$$r_t = \mu_t + \sigma_t \epsilon_t, \tag{1}$$

where

$$\sigma_t^2 = Var(r_t|F_{t-1}) = Var(a_t|F_{t-1}) \tag{2}$$

```
> at=log_return - mean(log_return)
> Box.test(at^2,10,type="Ljung")

        Box-Ljung test

data:  at^2
X-squared = 532.71, df = 10, p-value < 2.2e-16
```

Fig. 3: Residual squared of the log return

is the conditional standard deviation of the log return based on past information F_{t-1}.

In contrast with models like ARMA model, where $\sigma_t^2 = \sigma_a^2$ is assumed to be constant, GARCH model admits a time-dependent σ_t^2. The difference between the two is the so-called ARCH effect, and can be tested via the autocorrelation function (ACF) of the residual squared:

$$\xi_t = (r_t - \mu_t)^2 \tag{3}$$

where $\mu_t = E(r_t|F_{t-1})$ is the conditional mean of the log return based on past information. The null hypothesis states that the ACF ρ_k of ξ_t are all zero, i.e.

$$H_0 : \rho_1 = \cdots = \rho_m = 0.$$

The Ljung-Box test of this null hypothesis can be done on the sample ACF:

$$Q(m) = n(n-1)\sum_{k=1}^{m} \frac{\hat{\rho}_k^2}{n-k} \sim \chi^2(m)$$

As can be seen in Figure 3, the test rejects the null hypothesis and there is correlation among ξ_t, which implies that the standard variation of the log return is not constant. This helps to explain why a naive ARIMA model in [2] cannot predict the price well and in particular the forecast interval does not cover the real price most of the time.

The baseline model adopted is an iGARCH model with mean, i.e.

$$\begin{aligned} r_t &= \phi_0 + a_t, \\ a_t &= \sigma_t \epsilon_t, \\ \sigma_t^2 &= \alpha_0 + (1-\beta_1)a_{t-1}^2 + \beta_1 \sigma_{t-1}^2. \end{aligned}$$

The model fitted over the whole dataset is given by:

$$\begin{aligned} r_t &= 0.000283 + a_t, \\ a_t &= \sigma_t \epsilon_t, \\ \sigma_t^2 &= 0.043670 a_{t-1}^2 + 0.956330 \sigma_{t-1}^2. \end{aligned}$$

For model checking, the null hypothesis of the squared standardized residuals being white noise is checked by the Ljung-Box test, as shown in Figure 4. The p-value is larger than 0.05 and null hypothesis cannot be rejected, meaning that the model is correct.

For forecasting, the conditional mean ϕ_0 is used as the forecasting value. To obtain an accurate forecast, rolling prediction is adopted for the baseline model and any subsequent model. This makes sense as the price is increasing over time, and the

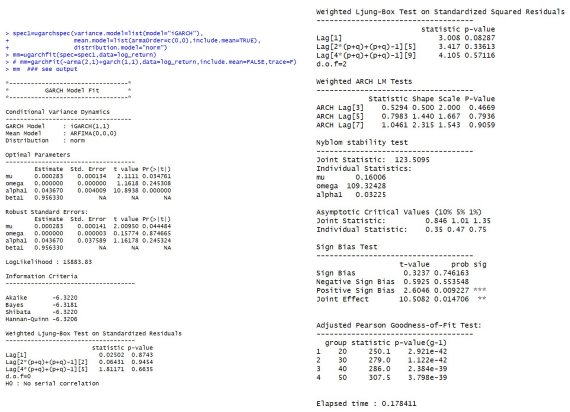

Fig. 4: Summary of iGARCH Model

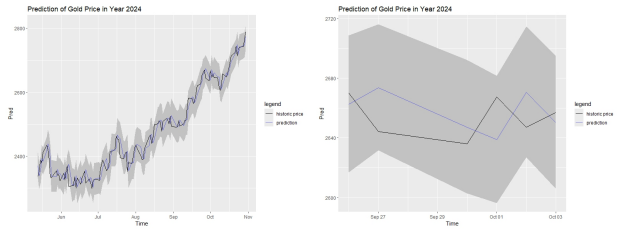

Fig. 5: Prediction of iGARCH Model

forecast should be updated accordingly as well. Regarding the forecast interval, since the noises a_{n+l} are not white noises (not i.i.d. normal) when $l > 1$, it is difficult to estimate the forecast interval for more than one step. Hence, one-step prediction is adopted for the baseline model.

The evaluation and comparison with other models is conducted in Section IV, while here the forecasting results are presented as in Figure 5. As can be seen, the forecasting value follows the real price in general, and the forecast intervals cover the real price. However, the forecast is always one step behind the real price, indicating that the model is simply mimicking the price after the fact.

B. Hidden Markov Model

The basic idea of a hidden Markov model (HMM) states that the time series observed is generated by some hidden states that represent different regimes the time series is in. In the case of the gold price, the hidden states captures the market mood at different times, and the price is generated according to some probability distribution corresponding to the hidden states.

Formally, a HMM consists of an observed time series $\vec{x} = \{x_1, x_2, \ldots, x_T\}$ drawn from a (finite) set of possible outputs $V = \{v_1, v_2, \ldots, v_{|V|}\}$, i.e. $x_t \in V, t = 1, 2, \ldots, T$, an unobserved time series $\vec{z} = \{z_1, z_2, \ldots, z_T\}$ drawn from a (finite) set of possible states $S = \{s_1, s_2, \ldots, s_{|S|}\}$, i.e. $z_t \in S, t = 1, \ldots, T$ [3]. The hidden states z_t follows the two Markov assumptions, namely the limited horizon assumption,

$$P(z_t|z_{t-1}, z_{t-2}, \ldots, z_1) = P(z_t|z_{t-1}), \tag{4}$$

i.e. probability of the next state depends only on the current state, and the stationary process assumption,

$$P(z_t|z_{t-1}) = P(z_2|z_1), \tag{5}$$

i.e. the conditional distribution does not change over time.

Due to the two Markov assumptions concerning the hidden states, the transition probability between different states can be characterized by a transition matrix,

$$A_{ij} = P(z_t = s_j|z_{t-1} = s_i).$$

To model the observed time series $\vec{x}$ as a function of the unobserved time series $\vec{z}$, an additional output independence assumption is made,

$$B_{jk} = P(x_t = v_k|z_t = s_j) = P(x_t = v_k|x_1, \ldots, x_T, z_1, \ldots, z_T),$$

where the matrix B is the matrix generating $\vec{x}$ from $\vec{z}$.

The probability of an observed series $\vec{x}$ can be calculated by

$$\begin{aligned} P(\vec{x}; A, B) &= \sum_{\vec{z}} P(\vec{x}, \vec{z}; A, B) \\ &= \sum_{\vec{z}} P(\vec{x}|\vec{z}; A, B) P(\vec{z}; A, B) \\ &= \sum_{\vec{z}} \prod_{t=1}^{T} P(x_t|z_t; B) \prod_{t=1}^{T} P(z_t|z_{t-1}; A) \\ &= \sum_{\vec{z}} \prod_{t=1}^{T} B_{z_t, x_t} A_{z_{t-1}, z_t}, \end{aligned}$$

where from the first line to the second line, the output independence assumption and the limited horizon assumption are applied, and from the second line to the third line, the stationary process assumption is used. In principle, the sum is over all possible configurations of $\vec{z}$, which requires $O(|S|^T)$ time, but a faster algorithm called *Forward Procedure* based on dynamic programming is used in practice. The description of the algorithm can be found in Appendix A.

Given an observed series $\vec{x}$, the most likely series of hidden states $\vec{z}$ can be calculated by

$$\begin{aligned} \arg\max_{\vec{z}} P(\vec{z}|\vec{x}; A, B) &= \arg\max_{\vec{z}} \frac{P(\vec{x}, \vec{z}; A, B)}{\sum_{\vec{z}'} P(\vec{x}, \vec{z}'; A, B)} \\ &= \arg\max_{\vec{z}} P(\vec{x}, \vec{z}; A, B), \end{aligned}$$

where from the first line to the second line, the Bayes rule is applied, and from the second line to the third line, an observation is made that the denominator does not depend on $\vec{z}$. Similar to the calculation of an observed series (by changing the $\sum_{\vec{z}}$ to $\arg\max_z$, an algorithm called the *Viterbi Algorithm* based on dynamic programming can be applied to solve for the $\vec{z}$.

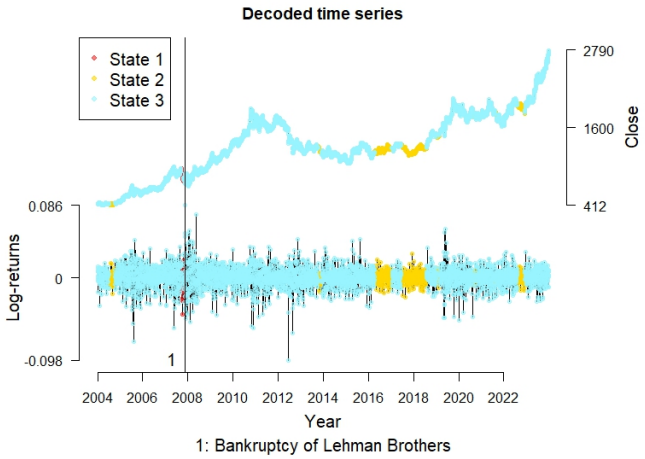

Fig. 6: State Decoding of HMM Model

Parameters estimation of A and B is another important question to ask of HMM, and they are in principle estimated by:

$$\begin{aligned} A, B = \arg\max_{A,B} & \sum_{\vec{z}} Q(\vec{z}) \log \frac{P(\vec{x}, \vec{z}; A, B)}{Q(\vec{z})} \\ s.t. \quad & \sum_{j=1}^{|S|} A_{ij} = 1, i = 1, \ldots, |S|; A_{ij} > 0, i, j = 1, \ldots, |S| \\ & \sum_{k=1}^{|V|} B_{jk} = 1, j = 1, \ldots, |S|; B_{jk} > 0, j = 1..|S|, k = 1..|V|. \end{aligned}$$

In practice, the estimation is done via the so-called *Forward-Backward Expectation–Maximization (EM) Algorithm*to avoid enumerating all possible $O(|S|^T)$ configurations of $\vec{z}$.

In R, a package called *fHMM* [4] is used for parameter estimation based on the EM algorithm and subsequent predictions based on the forward procedure. As discussed, the HMM admits an interpretation of the hidden states as representing the market mood, and this can be visualized by plotting the price time series with the corresponding hidden states marked at different time points, as shown in Figure 6 and Figure 7. State 3 occupies most of the history of the gold price, and represents an "increasing state", while very rarely does the price enters State 1, the "decreasing state", and occasionally the price is somewhere in between.

The state-dependent probability distribution functions that comprise the generating matrix B can also be visualized in Figure 8. Confirming the intuition mentioned above, State 3 and State 1 have a positive mean and negative mean, respectively, representing a generally positive / negative return. Both states have a relatively large standard deviation, representing volatile market conditions. State 2 with a mean close to zero and narrow shape represents a market with lower return but also lower risk.

For model checking, the residual is plotted in Figure 9 and it resembles a normal distribution. The correlation of the residual

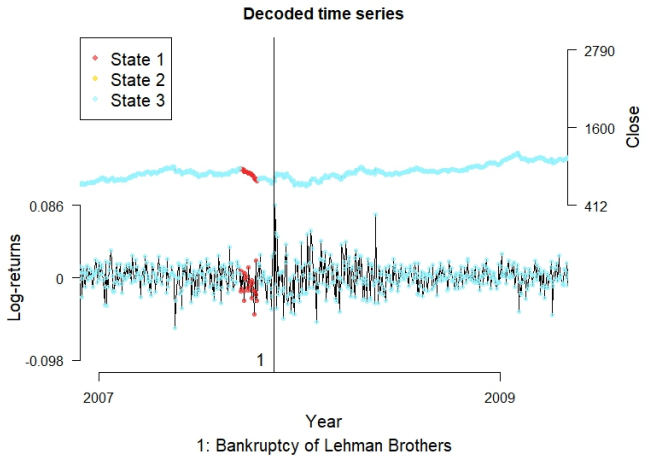

Fig. 7: Zoom-in of State Decoding of HMM Model

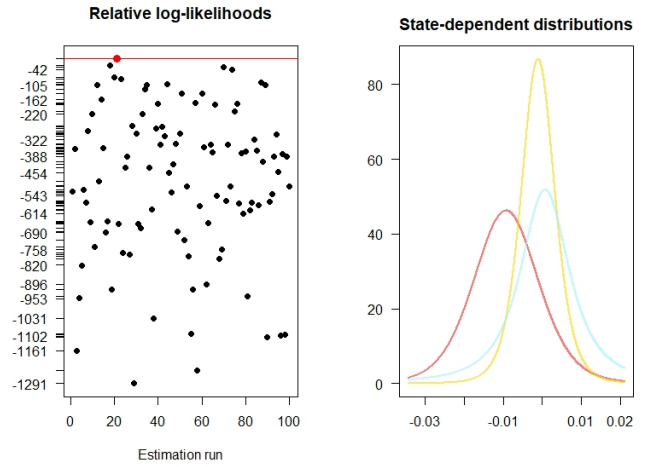

Fig. 8: State-dependent Distribution Functions

squared may further be examined to confirm this, and the Ljung-Box test gives a p-value of 0.2247, which is larger than 0.05, and thus the null hypothesis that the model is correct is not rejected.

Similar to the baseline model, a rolling prediction is made and shown in Figure 10. The prediction generally follows the market trend, but occasionally overshoots, indicating an unstable prediction of HMM.

IV. Evaluation

In this section, an evaluation is made for the model presented based on the Mean Absolute Percentage Error (MAPE), defined as

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|p_i - a_i|}{|a_i|} \times 100\%$$

The MAPE for each model is presented in Table I. The iGARCH model with 1-step prediction receives a comparatively good score of 0.74% and serves as a robust baseline.

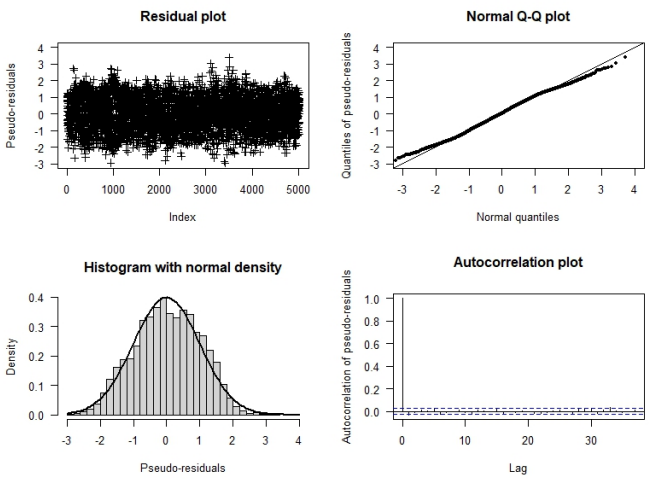

Fig. 9: Residual of the HMM Model

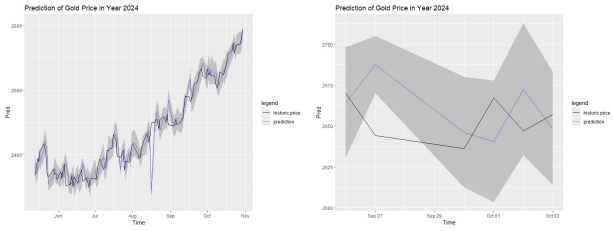

Fig. 10: Prediction of the HMM Model

The HMM with 4-step prediction performs relatively well considering that the forecasts are made over more steps. On the other hand, the HMM with 1-step prediction gives a larger error, perhaps due to overfitting (the overshoot seen in Figure 10). Teammates please supplement your model evaluations.

Model	MAPE
iGARCH (1-step)	0.74%
HMM (1-step)	0.87%
HMM (4-step)	0.78%

TABLE I: MAPE of Different Models

V. Conclusion

In this project, the gold price dataset ranging from Year 2004 to Year 2024 is studied. The features of the dataset is explored and it is found that the price is a non-stationary dataset that exhibits ARCH effect. As a baseline, iGARCH model is deployed for fitting and forecasting, and various other models including the HMM, Teammates please supplement your model names are investigated for their respective formulation and forecasting power. It is found that although the models perform relatively well according to the metric MAPE, the prediction mostly lags behind the real price, and it just represents the challenge of price prediction in general.

Algorithm 1: Forward Procedure

1. Base case: $\alpha_i(0) = A_{0i}, i = 1..|S|$;
2. Recursion:
   $\alpha_j(t) = \sum_{i=1}^{|S|} \alpha_i(t-1)A_{ij}B_{jx_t}, j = 1..|S|, t = 1..T$;
3. $P(\vec{x}; A, B) = \sum_{i=1}^{|S|} \alpha_i(T)$.

APPENDIX A
FORWARD PROCEDURE

APPENDIX B
FORWARD-BACKWARD ALGORITHM FOR HMM PARAMETER LEARNING

Algorithm 2: Forward-Backward algorithm for HMM parameter learning

1. Initialization: Set A and B as random valid probability matrices
where $A_{i0} = 0$ and $B_{0k} = 0$ for $i = 1..|S|$ and $k = 1..|V|$.
2. Repeat until convergence $\{$
(E-Step) Run the Forward and Backward algorithms to compute α_i and
β_i for $i = 1..|S|$. Then set:

$$\gamma_t(i,j) = \alpha_i(t)A_{ij}B_{jx_t}\beta_j(t+1)$$

(M-Step) Re-estimate the maximum likelihood parameters as:

$$A_{ij} = \frac{\sum_{t=1}^{T} \gamma_t(i,j)}{\sum_{j=1}^{|S|}\sum_{t-1}^{T} \gamma_t(i,j)}$$

$$B_{jk} = \frac{\sum_{j=1}^{|S|}\sum_{t-1}^{T} 1(x_t = v_k)\gamma_t(i,j)}{\sum_{j=1}^{|S|}\sum_{t-1}^{T} \gamma_t(i,j)}$$

$\}$

APPENDIX C
CODE OF IGARCH MODEL AND HMM MODEL

See https://github.com/waileuklo/Hidden-Markov-Model.

REFERENCES

[1] E. J. Levin, A. Montagnoli, and R. Wright, "Short-run and long-run determinants of the price of gold," 2006.

[2] L. Shen, K. Shen, C. Yi, and Y. Chen, "Regression and hidden markov models for gold price prediction," in *2020 IEEE International Conference on Big Data (Big Data)*, 2020, pp. 5451–5456.

[3] W. Zucchini, I. MacDonald, and R. Langrock, *Hidden Markov Models for Time Series: An Introduction Using R, Second Edition*, ser. Chapman & Hall/CRC Monographs on Statistics and Applied Probability. CRC Press, 2017. [Online]. Available: https://books.google.com.hk/books?id=8AdEDwAAQBAJ

[4] L. Oelschläger, T. Adam, and R. Michels, "fhmm: Hidden markov models for financial time series in r," *Journal of Statistical Software*, vol. 109, no. 9, p. 1–25, 2024. [Online]. Available: https://www.jstatsoft.org/index.php/jss/article/view/v109i09