Time Series Analysis of Stock Prices of HK Transportation Company

Group 2

May 3, 2024

Abstract

This report presents a time series analysis of the stock prices of three Hong Kong transportation companies: Kwoon Chung, MTR, and Transport International. The analysis encompasses the processes of data acquisition and cleaning, differencing and stationarity testing, white noise testing, ARCH effect examination, ARMA model identification and fitting, significance testing of parameters, residual checks, and fitting and forecasting with the ARMA-EGARCH model. The results of this analysis will aid in understanding the volatility characteristics of the stock prices and in predicting future price trends.

Additionally, the report integrates a Vector Autoregressive Moving Average (VARMA) model to further explore the dynamic interactions between the stock prices of MTR and Transport International. This inclusion is crucial to capture the interdependencies and feedback mechanisms that may not be apparent through univariate models. The VARMA model, specifically, allows us to quantify the influence that each company's past stock prices have on the other's current and future prices, thereby providing a more comprehensive understanding of their interrelations within the transportation sector. This holistic approach is essential for investors and policymakers interested in the strategic planning and risk management of these key transportation entities in Hong Kong.

1 Time Series Analysis for KC Stock Price

We have retrieved stock price data for KC from Yahoo Finance, spanning from January 1, 2021, to April 20, 2024. After cleansing the data to remove any missing values, we conducted a preliminary visualization analysis on the adjusted closing prices.

Figure 1: KC Stock Price Chart

1.1 Differencing and Stationarity Test

To ensure the time series' stationarity, we applied differencing to the stock price data.

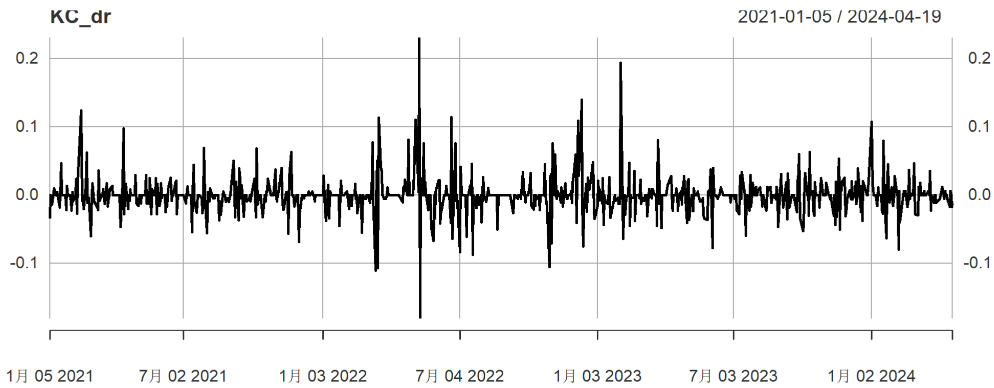

Figure 2: Differenced Stock Price Chart of Crown Corporation

The differenced data underwent a stationarity test, where the Phillips-Perron unit root test yielded a p-value below 0.05, confirming the time series' stationarity post-differencing.

```
> pp.test(KC_dr) # < 0.05 则平稳

        Phillips-Perron Unit Root Test

data:  KC_dr
Dickey-Fuller Z(alpha) = -880.95, Truncation
lag parameter = 6, p-value = 0.01
alternative hypothesis: stationary
```

Figure 3: Phillips-Perron Test Results

1.2 White Noise Test

Next, we performed a white noise test on the differenced data. The Ljung-Box test results indicated that for various lag orders, the p-values were consistently below 0.05 (with the exception of lag 11, where the p-value was slightly above at 0.05757), suggesting the series is not merely random noise and contains patterns that can be modeled.

```
	Box-Ljung test

data:  KC_dr
X-squared = 14.958, df = 2, p-value = 0.0005647


	Box-Ljung test

data:  KC_dr
X-squared = 15.839, df = 5, p-value = 0.00732


	Box-Ljung test

data:  KC_dr
X-squared = 19.121, df = 9, p-value = 0.02418


	Box-Ljung test

data:  KC_dr
X-squared = 19.202, df = 11, p-value = 0.05757
```

Figure 4: Ljung-Box Test Results

1.3 ARCH Effect Test

An ARCH effect test was also conducted. The Ljung-Box test on squared residuals showed that for all tested lags, p-values were significantly less than 0.05, indicating the presence of ARCH effects and suggesting that volatility clustering is present in the time series.

```
	Box-Ljung test

data:  KC_dr_at^2
X-squared = 74.613, df = 2, p-value < 2.2e-16


	Box-Ljung test

data:  KC_dr_at^2
X-squared = 92.985, df = 5, p-value < 2.2e-16


	Box-Ljung test

data:  KC_dr_at^2
X-squared = 93.873, df = 9, p-value = 2.22e-16


	Box-Ljung test

data:  KC_dr_at^2
X-squared = 95.4, df = 11, p-value = 1.443e-15
```

Figure 5: ARCH Effect Test Results

1.4 ARMA Model Identification and Fitting

With the presence of predictable structure confirmed, we proceeded to identify the order of the ARMA model using the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF).

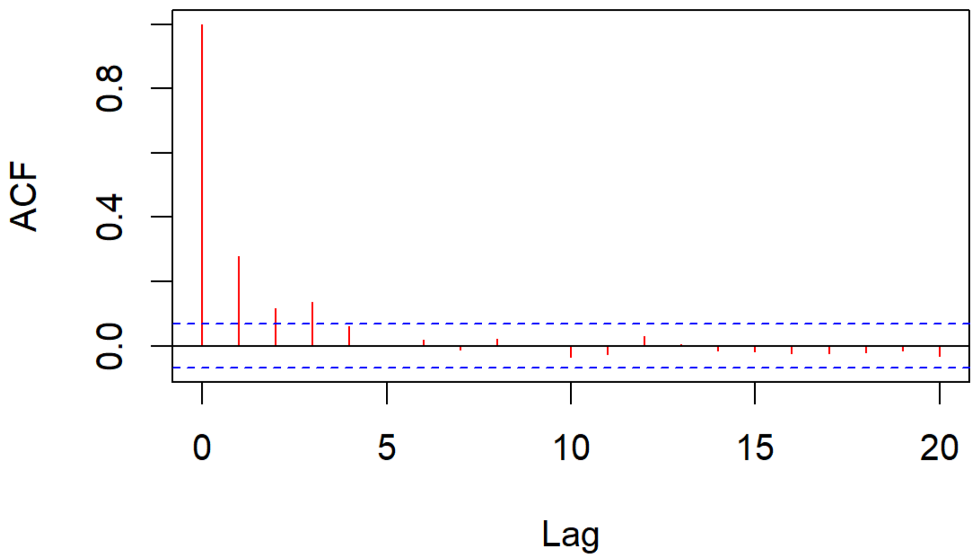

Figure 6: ACF for Differenced Series

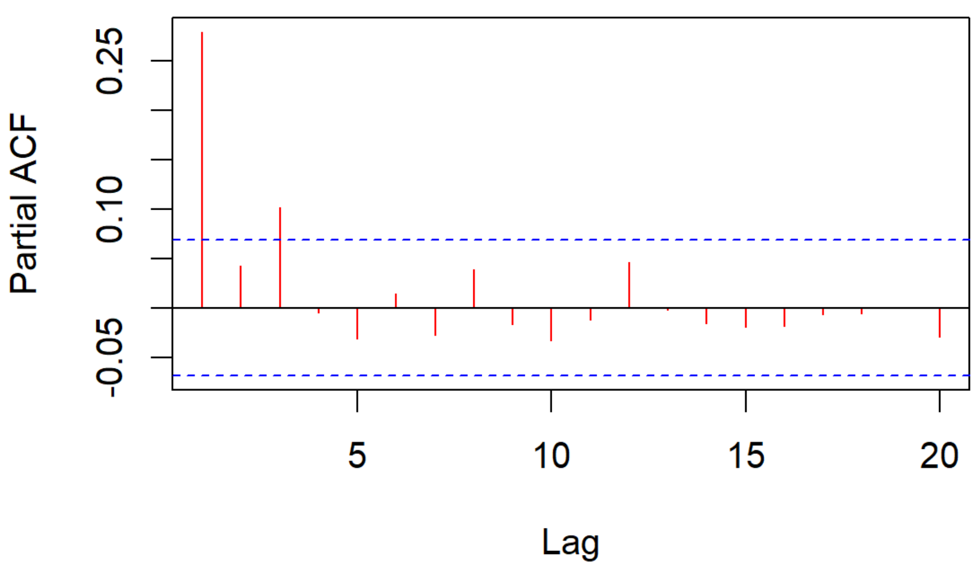

Figure 7: PACF for Differenced Series

The Extended Autocorrelation Function (EACF) table and automatic ARIMA model selection procedures suggested an ARMA(0,1) model as the best fit for the data.

```
> eacf(KC_dr)
AR/MA
  0 1 2 3 4 5 6 7 8 9 10 11 12 13
0 x o o o o o o o o o o  o  o  o
1 x o o o o o o o o o o  o  o  o
2 x o o o o o o o o o o  o  o  o
3 x x x o o o o o o o o  o  o  o
4 x x x o o o o o o o o  o  o  o
5 x x x x x o o o o o o  o  o  o
6 x x x o o o o o o o o  o  o  o
7 x x x x o o x o o o o  o  o  o
```

Figure 8: EACF Table

```
Fitting models using approximations to speed things up...

ARIMA(2,0,2) with non-zero mean : -3482.303
ARIMA(0,0,0) with non-zero mean : -3474.608
ARIMA(1,0,0) with non-zero mean : -3487.999
ARIMA(0,0,1) with non-zero mean : -3488.263
ARIMA(0,0,0) with zero mean     : -3476.618
ARIMA(1,0,1) with non-zero mean : -3487.277
ARIMA(0,0,2) with non-zero mean : -3486.544
ARIMA(1,0,2) with non-zero mean : -3485.269
ARIMA(0,0,1) with zero mean     : -3490.277
ARIMA(1,0,1) with zero mean     : -3489.297
ARIMA(0,0,2) with zero mean     : -3488.563
ARIMA(1,0,0) with zero mean     : -3490.013
ARIMA(1,0,2) with zero mean     : -3487.178

Now re-fitting the best model(s) without approximations...

ARIMA(0,0,1) with zero mean     : -3490.29

Best model: ARIMA(0,0,1) with zero mean
```

Figure 9: Automatic ARIMA Selection

Comparing different models based on the Akaike Information Criterion (AIC), we determined that the ARMA(0,1) model had the smallest AIC, indicating the best fit.

```
> Arima(KC_dr,order = c(0,0,1), include.drift = T)$aic
[1] -3486.354
> Arima(KC_dr,order = c(0,0,2), include.drift = T)$aic
[1] -3484.658
> Arima(KC_dr,order = c(1,0,1), include.drift = T)$aic
[1] -3484.716
> Arima(KC_dr,order = c(1,0,2), include.drift = T)$aic
[1] -3482.783
> Arima(KC_dr,order = c(2,0,1), include.drift = T)$aic
[1] -3482.799
> Arima(KC_dr,order = c(2,0,2), include.drift = T)$aic
[1] -3480.751
```

Figure 10: Model Comparison Based on AIC

Upon fitting the ARMA(0,1) model to the data, we found that the MA1 coefficient was significant, while the intercept and drift terms in the model with the drift were not significant.

```
Series: KC_dr
ARIMA(0,0,1) with zero mean

Coefficients:
          ma1
      -0.1427
s.e.   0.0356

sigma^2 = 0.0007802:  log likelihood = 1747.15
AIC=-3490.3   AICc=-3490.29   BIC=-3480.91
```

Figure 11: ARMA(0,1) Model Fit

The model is expressed as:

$$r_t = -0.1427\epsilon_{t-1} + \epsilon_t$$

1.5 ARMA-EGARCH Model Fitting

Due to the presence of ARCH effects, we further employed an ARMA-EGARCH model to capture the volatility structure of the time series.

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : eGARCH(1,1)
Mean Model      : ARFIMA(0,0,1)
Distribution    : sstd

Optimal Parameters
------------------------------------
        Estimate  Std. Error   t value Pr(>|t|)
mu     -0.000928    0.000380  -2.43887 0.014733
ma1    -0.081771    0.025599  -3.19428 0.001402
omega  -1.074466    0.208721  -5.14785 0.000000
alpha1  0.093742    0.203073   0.46162 0.644357
beta1   0.792939    0.046496  17.05376 0.000000
gamma1  1.992568    0.259866   7.66768 0.000000
skew    0.977053    0.024541  39.81331 0.000000
shape   2.029626    0.002117 958.92389 0.000000
```

Figure 12: EGARCH Model Fit

The fitting results indicated that the α_1 parameter (news impact curve parameter) in the EGARCH model was not significant and was thus set to zero in the final model. Other parameters were significant.

The model is formulated as:

$$\ln(\sigma_t^2) = -1.220449 + 0.808367 \ln(\sigma_{t-1}^2)$$

$$g(Z_{t-1}) = \theta Z_{t-1} + (|Z_{t-1}| - E|Z_{t-1}|)$$

$$r_t = -0.001123 + \epsilon_t$$

$$\epsilon_t = \sigma_t Z_t$$

$$Z_t \sim SSTD(\xi, \omega)$$

Parameters: - α_1 (news impact curve parameter): 0.000000 (indicating no leverage effect) - *skew* (skewness): 0.968709 - *shape* (shape): 2.099822

These estimates suggest that the model captures the asymmetry and fat tails characteristic of stock price volatility, particularly through the shape and scale parameters of the skewed Student's t-distribution.

1.6 Model Diagnostics and Validation

First, we assessed the validity of the model:

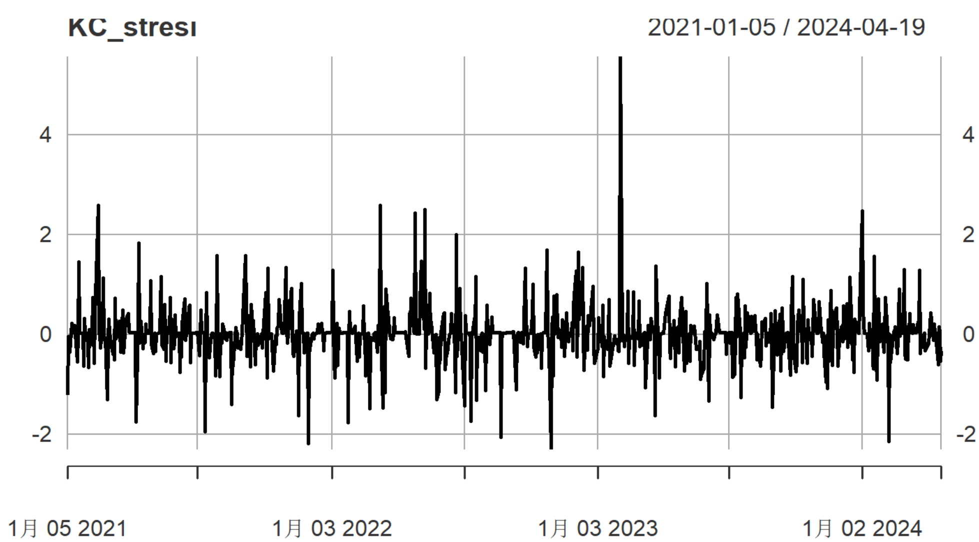

Figure 13: Standardized Residual Series of the Model

Next, we tested the white noise of the residual series:

```
        Box-Ljung test

data:  KC_stresi
X-squared = 760.56, df = 804, p-value = 0.8614
```

Figure 14: White Noise Test of the Standardized Residual Series

With a p-value greater than 0.05, the residuals are considered to be white noise.

We also tested for ARCH effects in the residual series:

```
        Box-Ljung test

data:  KC_stresi^2
X-squared = 365.42, df = 804, p-value = 1
```

Figure 15: ARCH Test of the Standardized Residual Series

With a p-value greater than 0.05, no ARCH effect remains.

The ARMA(0,1)-EGARCH(1,1) model analysis of the stock price time series data resulted in a model that adequately captures the mean and volatility of the data. The absence of leverage effect in the parameter estimates indicates that positive and negative stock price volatilities symmetrically affect the volatility. The model diagnostics suggest that the standardized residuals behave as white noise and no ARCH effects are present, indicating the model successfully captures the information in the time series without any remaining autocorrelation structures. Therefore, this model is considered an appropriate choice for modeling and forecasting the stock price time series.

1.7 Forecasting with the Model

Utilizing the fitted ARMA-EGARCH model, we forecasted the stock prices for the next three days and visualized the forecast results.

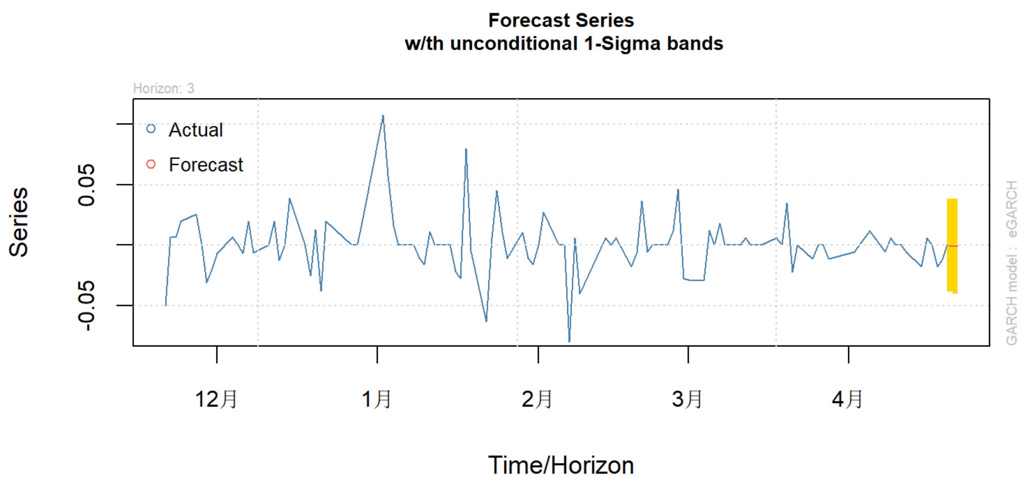

Figure 16: Forecast Visualization

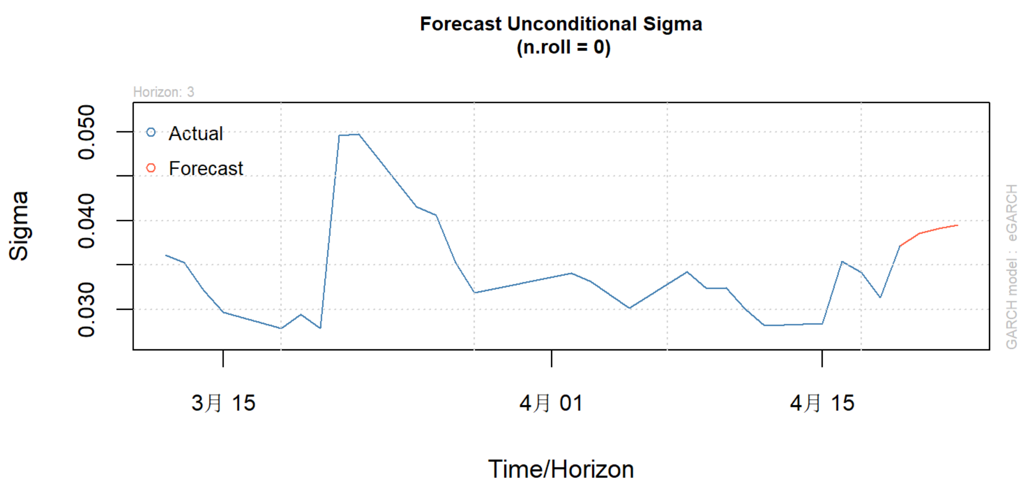

Figure 17: Detailed Forecast Visualization

This comprehensive analysis and modeling provide a robust framework for understanding and predicting the behavior of KC's stock prices based on historical data and statistical modeling techniques.

2 Time Series Analysis of MTR Corporation Stock Price

We have sourced the stock price data of MTR Corporation from Yahoo Finance, covering the period from January 1, 2015, to April 20, 2024. The data was cleansed to remove any missing values, and a preliminary visualization of the adjusted closing prices was performed.

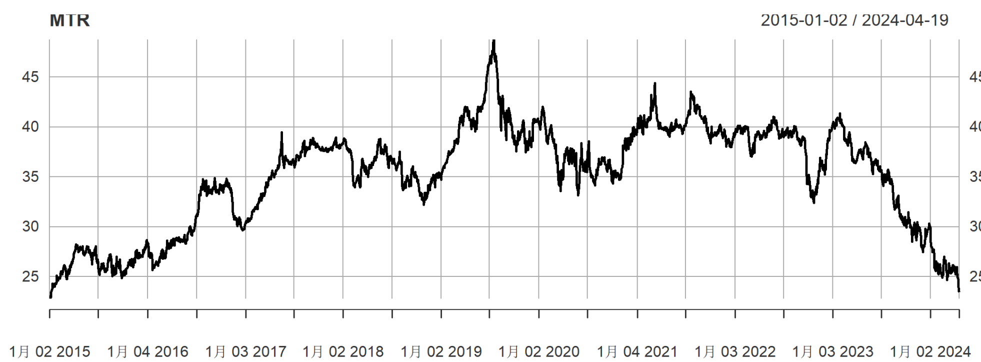

Figure 18: Adjusted Closing Prices of MTR Corporation

2.1 Differencing and Stationarity Test

Initially, we applied differencing to the stock price data of MTR Corporation:

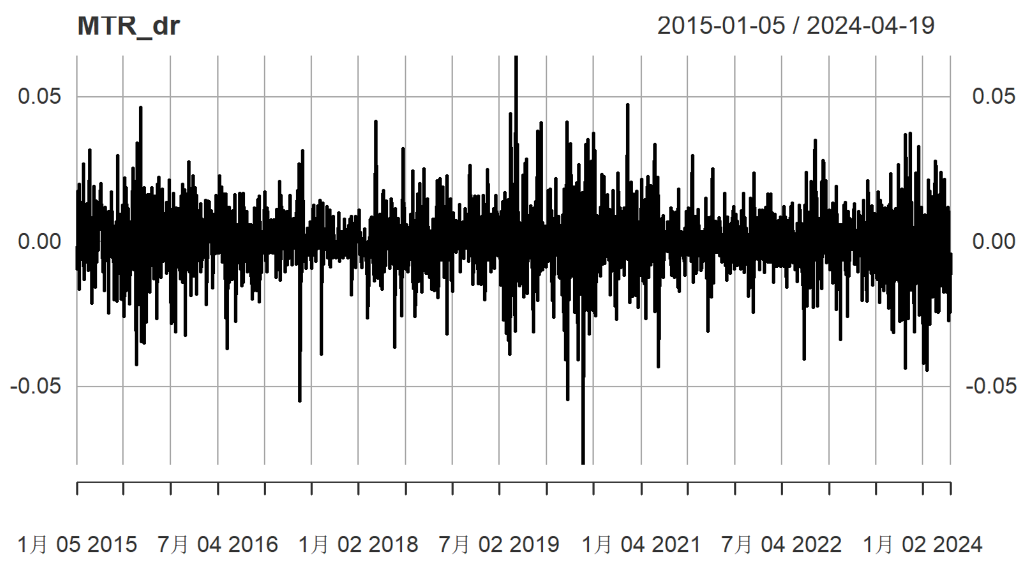

Figure 19: Differenced Stock Price Data of MTR Corporation

A stationarity test followed:

```
        Phillips-Perron Unit Root Test

data:  MTR_dr_1
Dickey-Fuller Z(alpha) = -2805, Truncation lag parameter = 8,
p-value = 0.01
alternative hypothesis: stationary
```

Figure 20: Phillips-Perron Unit Root Test Results

The results from the `ndiffs` function indicated that the data required differencing once to achieve stationarity. The p-value from the Phillips-Perron unit root test was less than 0.05, further confirming the stationarity of the differenced data.

2.2 White Noise Test

```
        Box-Ljung test

data:  MTR_dr_1
X-squared = 623.07, df = 2, p-value < 2.2e-16


        Box-Ljung test

data:  MTR_dr_1
X-squared = 630.35, df = 5, p-value < 2.2e-16


        Box-Ljung test

data:  MTR_dr_1
X-squared = 653, df = 9, p-value < 2.2e-16


        Box-Ljung test

data:  MTR_dr_1
X-squared = 658.89, df = 11, p-value < 2.2e-16
```

Figure 21: Ljung-Box Test Results

The white noise test results indicated that the differenced data is not a white noise process. The Ljung-Box test's p-values were significantly less than 0.05 for various lags, suggesting that the series contains autocorrelation.

2.3 ARCH Effect Test

```
        Box-Ljung test

data:  MTR_dr_1_at^2
X-squared = 343.2, df = 2, p-value < 2.2e-16


        Box-Ljung test

data:  MTR_dr_1_at^2
X-squared = 437.28, df = 5, p-value < 2.2e-16


        Box-Ljung test

data:  MTR_dr_1_at^2
X-squared = 513.92, df = 9, p-value < 2.2e-16


        Box-Ljung test

data:  MTR_dr_1_at^2
X-squared = 551.47, df = 11, p-value < 2.2e-16
```

Figure 22: Test for ARCH Effects

Testing the squared differenced data for autocorrelation revealed significant ARCH effects at various lags, as the Ljung-Box test's p-values were well below 0.05. This suggests that the data's volatility exhibits time clustering, necessitating a GARCH-type model to capture this volatility.

2.4 ARMA Model Identification and Fitting

Observing the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) of the differenced data:

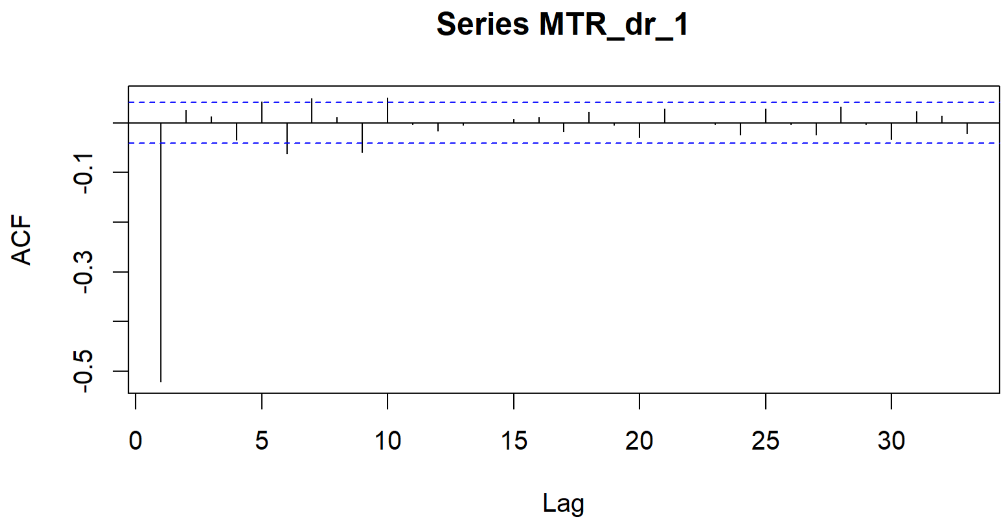

Figure 23: ACF of the Differenced Data

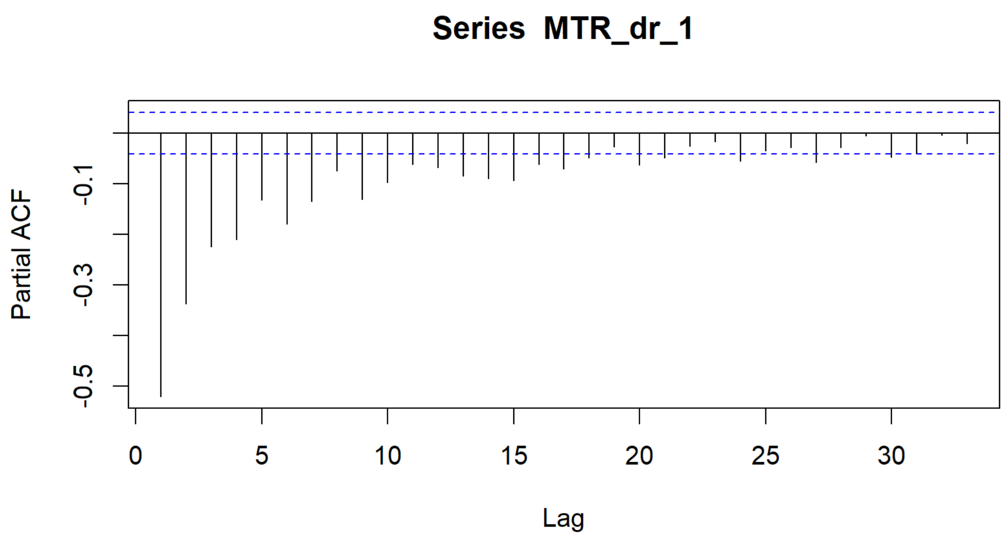

Figure 24: PACF of the Differenced Data

From the analysis of the Extended Autocorrelation Function (EACF) table, we selected the ARMA(0,1) model for fitting.

AR/MA	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	x	o	o	o	x	x	x	o	x	x	o	o	o	o
1	x	x	o	o	o	o	x	o	x	x	o	o	o	o
2	x	x	x	o	o	o	o	o	o	o	o	o	o	o
3	x	x	x	x	o	o	o	o	o	o	o	o	o	o
4	x	x	x	x	x	o	o	o	o	o	o	o	o	o
5	x	x	x	x	x	x	o	o	o	o	o	o	o	o
6	x	x	x	o	x	x	x	o	o	o	o	o	o	o
7	x	x	x	x	o	x	x	o	o	o	o	o	o	o

Figure 25: EACF Table

This model had the smallest Akaike Information Criterion (AIC) value when compared across different model orders.

2.4.1 Residual Check

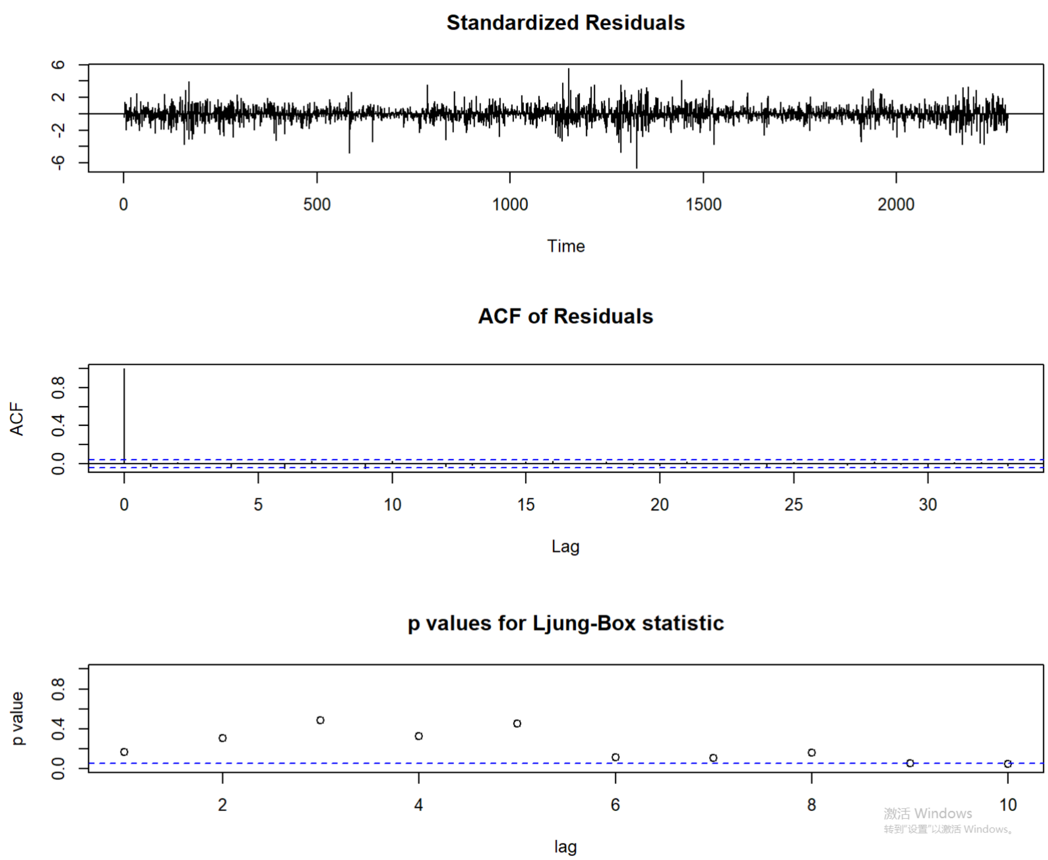

Figure 26: Residual Check

2.4.2 ARCH Effect Test

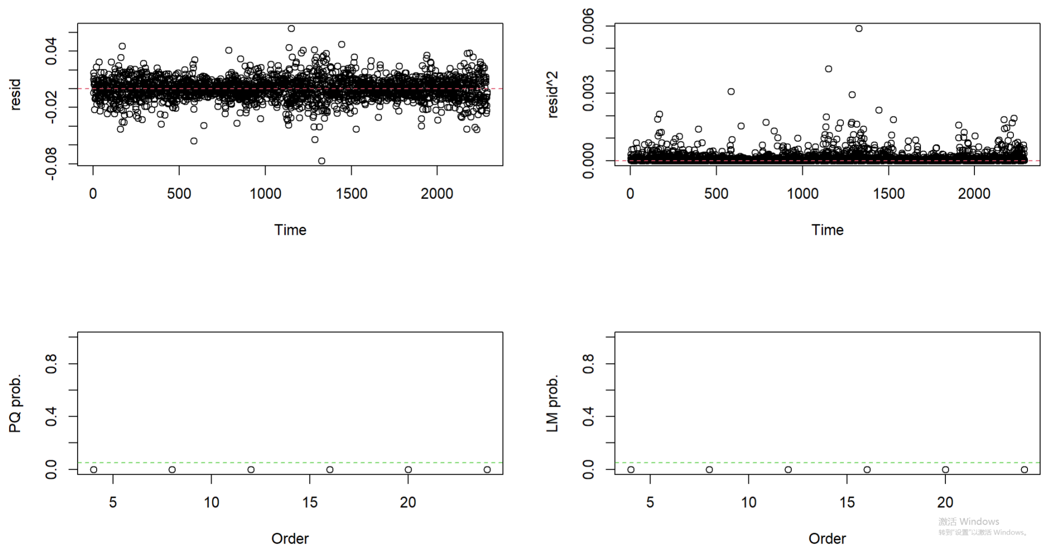

Figure 27: ARCH Effect Validation

```
> Arima(MTR_dr_1,order = c(0,0,1), include.drift = T)$aic
[1] -13903.34
> Arima(MTR_dr_1,order = c(0,0,2), include.drift = T)$aic
[1] -13903.24
> Arima(MTR_dr_1,order = c(1,0,2), include.drift = T)$aic
[1] -13899.42
```

Figure 28: Model Comparison Based on AIC

2.5 ARMA-EGARCH Model Fitting

Based on the analysis, an ARMA(0,1) model combined with an EGARCH(1,1) model was selected for fitting. Insignificant parameters, including the mean term μ and the symmetric term α_1, were fixed in the final model. The parameter estimation results showed that θ_1 (MA coefficient), β_1 (persistence coefficient in the GARCH term), and γ_1 (leverage effect coefficient) were significantly non-zero, while α_1 was set to zero, indicating the absence of a leverage effect.

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : eGARCH(1,1)
Mean Model      : ARFIMA(0,0,1)
Distribution    : sstd

Optimal Parameters
------------------------------------
        Estimate  Std. Error     t value Pr(>|t|)
mu     -0.000001    0.000001     -1.7022 0.088713
ma1    -0.990285    0.000060 -16403.9130 0.000000
omega  -0.201507    0.015939    -12.6427 0.000000
alpha1 -0.014956    0.012114     -1.2346 0.216985
beta1   0.977693    0.001752    558.0389 0.000000
gamma1  0.172528    0.024841      6.9452 0.000000
skew    0.983786    0.028845     34.1060 0.000000
shape   6.210653    0.779670      7.9657 0.000000
```

Figure 29: EGARCH Model Fit

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : eGARCH(1,1)
Mean Model      : ARFIMA(0,0,1)
Distribution    : sstd

Optimal Parameters
------------------------------------
        Estimate  Std. Error    t value Pr(>|t|)
ma1     -0.99060    0.000248 -3993.7073        0
omega   -0.21897    0.019408   -11.2826        0
alpha1   0.00000          NA         NA       NA
beta1    0.97578    0.002157   452.4575        0
gamma1   0.18286    0.027699     6.6015        0
skew     0.98953    0.031937    30.9836        0
shape    6.22232    0.846954     7.3467        0
```

Figure 30: EGARCH Model with Insignificant Variables Removed

The model equations can be expressed as:

$$
\begin{aligned}
r_t &= \mu_t + a_t, \\
\mu_t &= \mu_0 - \theta_1 \cdot a_{t-1}, \\
a_t &= \sigma_t \cdot \varepsilon_t, \\
\ln(\sigma_t^2) &= \alpha_0 + [\alpha_1 \cdot (\varepsilon_{t-1}) + \gamma_1 \cdot (|\varepsilon_{t-1}| - E|\varepsilon_{t-1}|)] + \beta_1 \cdot \ln(\sigma_{t-1}^2),
\end{aligned}
$$

where the parameters are set as follows:

- $\mu_0 = 0$, indicating no constant term.
- $\theta_1 = -0.99060$, the parameter of the MA(1) model.
- $\alpha_0 = -0.21897$, the constant term of the EGARCH model.
- $\alpha_1 = 0$, indicating no leverage effect.
- $\beta_1 = 0.97578$, the coefficient of the GARCH term, showing the persistence of volatility.
- $\gamma_1 = 0.18286$, indicating the asymmetry of volatility.

The skewness and kurtosis of the distribution:

- Skewness (skew) greater than 0 indicates a right-skewed distribution.
- Kurtosis (shape) greater than 3 indicates a leptokurtic distribution.

2.6 Residual Check

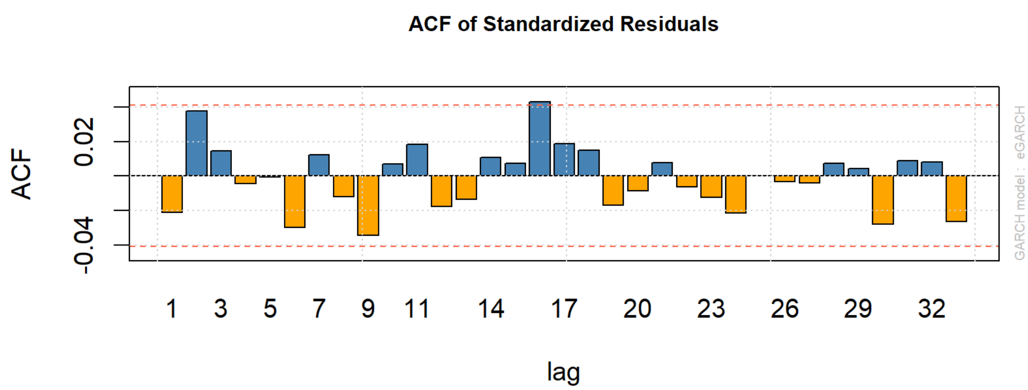

Figure 31: ACF of Residuals

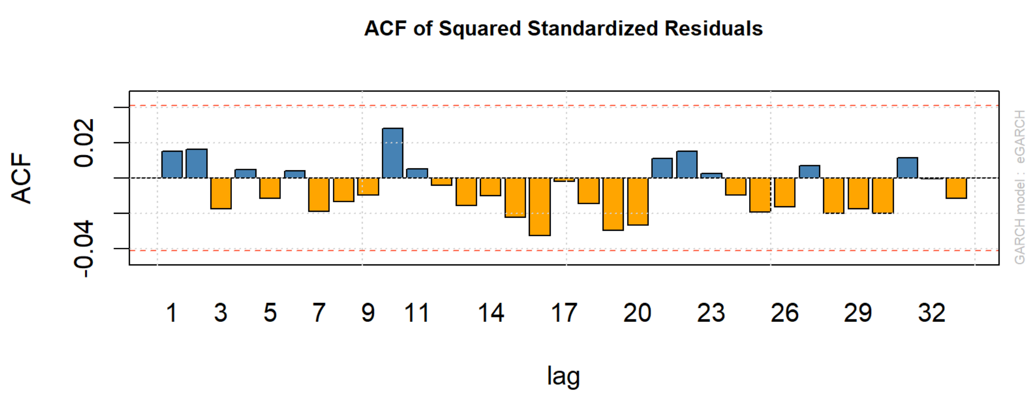

Figure 32: PACF of Residuals

The white noise test of the standardized residuals yielded a Ljung-Box p-value greater than 0.05, indicating that the residual series can be considered white noise. Furthermore, the test for squared standardized residuals did not reveal any ARCH effects, indicating that the model has adequately captured the data's volatility.

```
> Box.test(MTR_stresi,2288,type="Ljung-Box",fitdf = 4) # p-value > 0.05, white
noise

        Box-Ljung test

data:  MTR_stresi
X-squared = 2358.4, df = 2284, p-value = 0.1358

> Box.test(MTR_stresi^2,2288,type="Ljung-Box",fitdf = 4) # p-value > 0.05, rem
ains no ARCH effect

        Box-Ljung test

data:  MTR_stresi^2
X-squared = 1824.5, df = 2284, p-value = 1
```

Figure 33: Residual Diagnostics

2.7 Model Forecasting

Finally, using the fitted ARMA(0,1)-EGARCH(1,1) model, we conducted a forecast for the next three time points and graphically displayed the forecast results and confidence intervals.

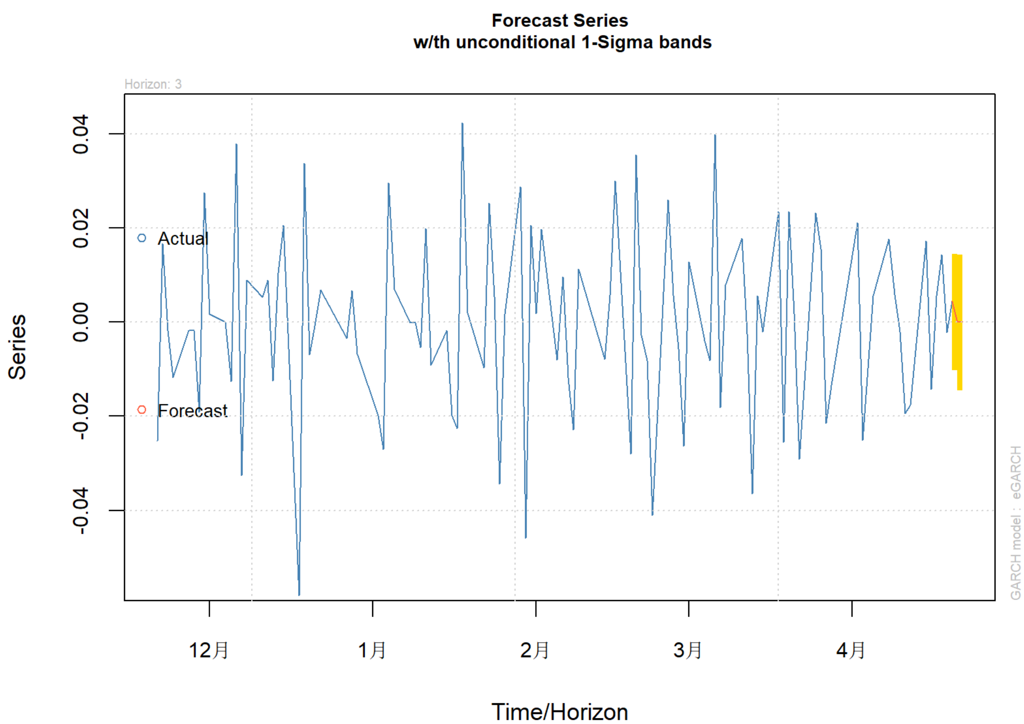

Figure 34: Forecast Visualization

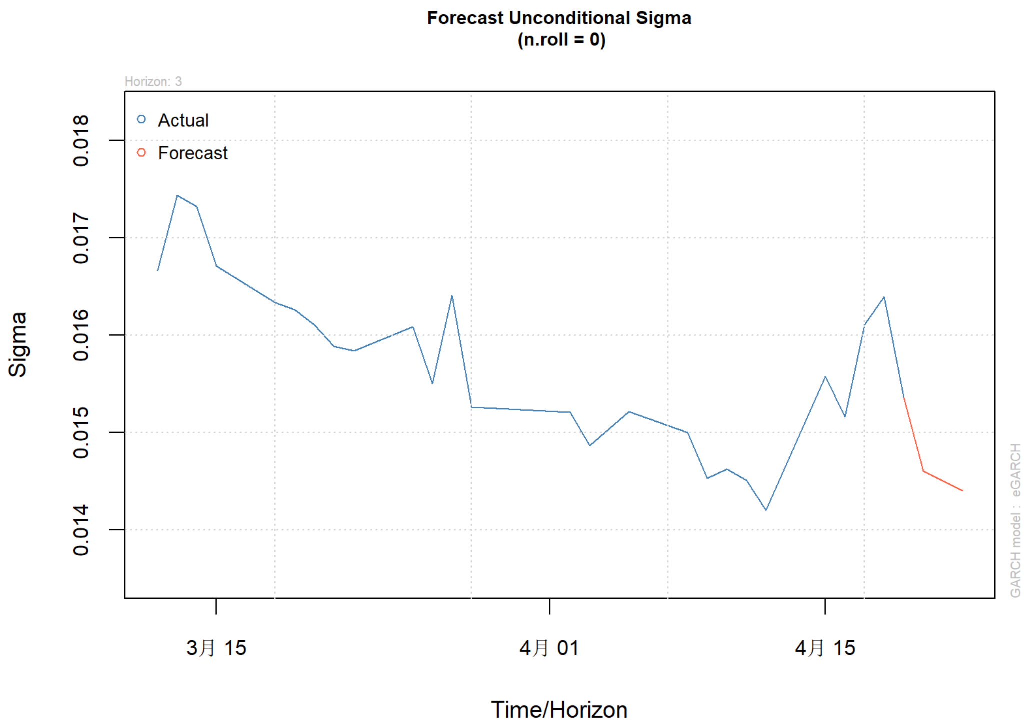

Figure 35: Detailed Forecast Visualization

3 Time Series Analysis of TI Stock Price

We have obtained the stock price data for TI from Yahoo Finance, spanning from January 1, 2015, to April 20, 2024. The data was cleaned to remove any missing values and a preliminary visualization analysis of the adjusted closing prices was conducted.

Figure 36: Adjusted Closing Prices of TI

3.1 Differencing and Stationarity Test

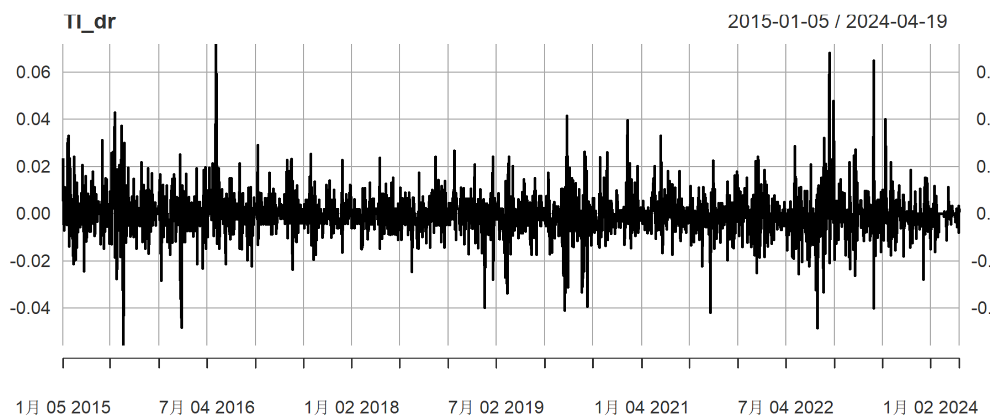

Figure 37: Differenced TI Data

To ensure the stationarity of the time series data, we first performed a first-order differencing on the raw data and calculated the ratio of the differenced data. The `ndiffs` function indicated that the data required one differencing to achieve stationarity. The length of the differenced data is 2289.

```
Phillips-Perron Unit Root Test
alternative: stationary

Type 1: no drift no trend
 lag Z_rho p.value
   8 -2728    0.01
-----
 Type 2: with drift no trend
 lag Z_rho p.value
   8 -2728    0.01
-----
 Type 3: with drift and trend
 lag Z_rho p.value
   8 -2728    0.01
---------------
Note: p-value = 0.01 means p.value <= 0.01
```

Figure 38: Phillips-Perron Unit Root Test Results

The results of the Phillips-Perron unit root test show that the p-value is less than 0.05 under no trend with no drift, with drift no trend, and with drift with trend, indicating that the data is stationary after differencing.

3.2 White Noise Test

```
Phillips-Perron Unit Root Test
alternative: stationary

Type 1: no drift no trend
 lag Z_rho p.value
   8 -2728    0.01
-----
 Type 2: with drift no trend
 lag Z_rho p.value
   8 -2728    0.01
-----
 Type 3: with drift and trend
 lag Z_rho p.value
   8 -2728    0.01
---------------
Note: p-value = 0.01 means p.value <= 0.01
```

Figure 39: Ljung-Box Test Results for TI Data

The white noise test on the differenced data using the Ljung-Box test shows that the p-values are significantly less than 0.05 for various lags (2, 5, 9, 11), indicating that the series has significant autocorrelation and is not white noise.

3.3 ARCH Effect Test

```
        Box-Ljung test

data:  TI_dr_1_at^2
X-squared = 430.78, df = 2, p-value < 2.2e-16


        Box-Ljung test

data:  TI_dr_1_at^2
X-squared = 465.45, df = 5, p-value < 2.2e-16


        Box-Ljung test

data:  TI_dr_1_at^2
X-squared = 482.73, df = 9, p-value < 2.2e-16


        Box-Ljung test

data:  TI_dr_1_at^2
X-squared = 494.36, df = 11, p-value < 2.2e-16
```

Figure 40: ARCH Effect Test for TI Data

Conducting the Ljung-Box test on the squared differenced data reveals significant ARCH effects at all lags, as p-values are well below 0.05. This indicates that the data's volatility is clustered, necessitating modeling through an appropriate GARCH model.

3.4 ARMA Model Identification and Fitting

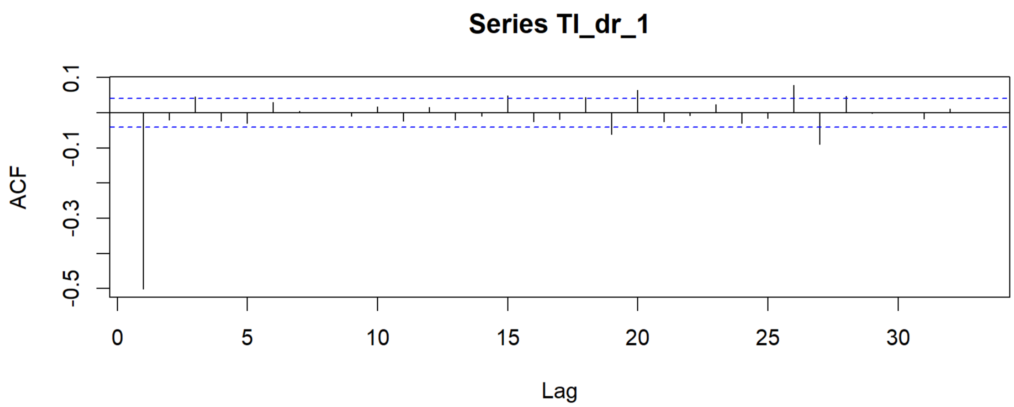

Figure 41: ACF for Differenced TI Data

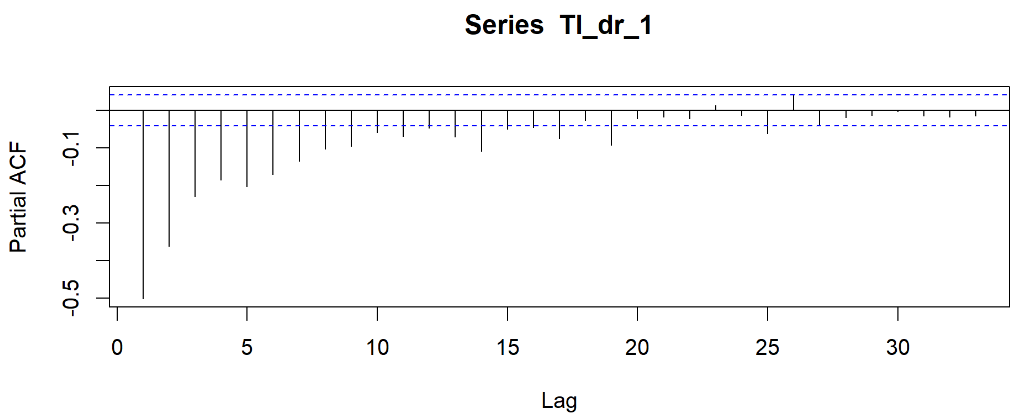

Figure 42: PACF for Differenced TI Data

```
AR/MA
  0 1 2 3 4 5 6 7 8 9 10 11 12 13
0 x o x o o o o o o o o  o  o  o
1 x x x o x o o o o o o  o  o  o
2 x x x o o o o o o o o  o  o  o
3 x x x x o o o o o o o  o  o  o
4 x x x x x o o o o o o  o  o  o
5 x x o x x x o o o o o  o  o  o
6 x x o x x x x o o o o  o  o  o
7 x x x x o x o o o o o  o  o  o
```

Figure 43: EACF Table for TI Data

Based on the analysis of the ACF, PACF plots, and the EACF table, an ARMA(0,1) model was selected for fitting.

```
> Arima(TI_dr_1,order = c(0,0,1), include.drift = T)$aic
[1] -14794.99
> Arima(TI_dr_1,order = c(0,0,3), include.drift = T)$aic
[1] -14791.79
> Arima(TI_dr_1,order = c(1,0,3), include.drift = T)$aic
[1] -14789.58
> Arima(TI_dr_1,order = c(2,0,3), include.drift = T)$aic
[1] -14791.75
```

Figure 44: Model Comparison Based on AIC for TI Data

Comparing potential models, the ARMA(0,1) model had an AIC value of -14794.99, which is lower compared to other models, suggesting a better fit.

3.5 ARMA-EGARCH Model Fitting

```
*          GARCH Model Fit         *
*----------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : eGARCH(1,1)
Mean Model      : ARFIMA(0,0,1)
Distribution    : sstd

Optimal Parameters
------------------------------------
        Estimate  Std. Error    t value Pr(>|t|)
mu     -0.000006    0.000006  -0.995180 0.319649
ma1    -0.964202    0.001638 -588.627937 0.000000
omega  -0.997416    0.213931  -4.662316 0.000003
alpha1 -0.001358    0.029814  -0.045558 0.963663
beta1   0.893073    0.022825  39.127283 0.000000
gamma1  0.448641    0.050249   8.928345 0.000000
skew    1.053115    0.028138  37.426481 0.000000
shape   3.371743    0.272216  12.386300 0.000000

Robust Standard Errors:
        Estimate  Std. Error    t value Pr(>|t|)
mu     -0.000006    0.000006  -0.951767 0.341215
ma1    -0.964202    0.001674 -575.932999 0.000000
omega  -0.997416    0.278347  -3.583348 0.000339
alpha1 -0.001358    0.043187  -0.031451 0.974910
beta1   0.893073    0.029862  29.907041 0.000000
gamma1  0.448641    0.048621   9.227314 0.000000
skew    1.053115    0.027714  37.999685 0.000000
shape   3.371743    0.275502  12.238541 0.000000

LogLikelihood : 7767.791
```

Figure 45: ARMA(0,1)-EGARCH(1,1) Model Fit for TI Data

The mathematical expression for the ARMA(0,1)-EGARCH(1,1) model is as follows:
ARMA(0,1) part:

$$r_t = \mu_t + a_t$$

where the mean part μ_t is defined as:

$$\mu_t = \mu_0 - \theta_1 \cdot a_{t-1}$$

In your model, $\mu_0 = 0$ and $\theta_1 = -0.96347$, therefore:

$$\mu_t = -0.96347 \cdot a_{t-1}$$

EGARCH(1,1):

$$\ln(\sigma_t^2) = \alpha_0 + [\alpha_1(\epsilon_{t-1}) + \gamma_1(|\epsilon_{t-1}| - \mathbb{E}[|\epsilon_{t-1}|])] + \beta_1 \ln(\sigma_{t-1}^2)$$

$\alpha_0 = -0.98647$, $\alpha_1 = 0$ (indicating no leverage effect), $\beta_1 = 0.89404$ and $\gamma_1 = 0.44817$. Therefore, the model simplifies to:

$$\ln(\sigma_t^2) = -0.98647 + \gamma_1(|\epsilon_{t-1}| - \mathbb{E}[|\epsilon_{t-1}|]) + 0.89404 \ln(\sigma_{t-1}^2)$$

The skewness and kurtosis of the distribution are:

- Skewness (skew): skew > 0, indicating a right-skewed distribution.

- Kurtosis (shape): shape > 3, indicating a leptokurtic distribution.

The ARMA(0,1) model combined with EGARCH(1,1) was chosen, including dynamic modeling of both the mean and volatility. Model parameters show that the MA parameter and β in GARCH are significant, while α is not, indicating the absence of a leverage effect.

3.6 Residual Check

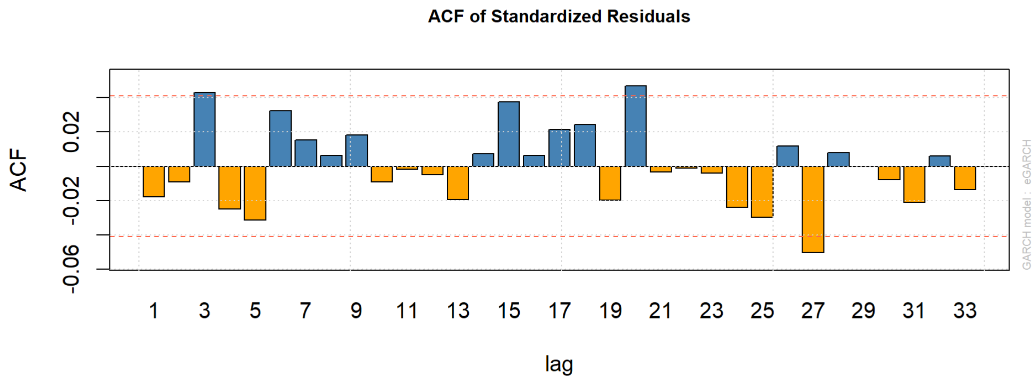

Figure 46: ACF of Residuals for TI Data

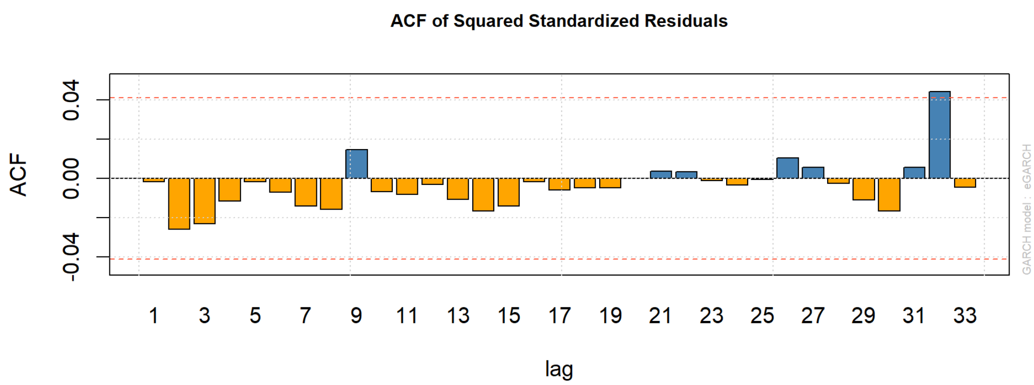

Figure 47: PACF of Residuals for TI Data

```
> Box.test(TI_stresi,2288,type="Ljung-Box",fitdf = 4) # p-value > 0.05, white noise

        Box-Ljung test

data:  TI_stresi
X-squared = 2178.7, df = 2284, p-value = 0.9421

> Box.test(TI_stresi^2,2288,type="Ljung-Box",fitdf = 4) # p-value > 0.05, remains no ARCH eff
ect

        Box-Ljung test

data:  TI_stresi^2
X-squared = 2229.3, df = 2284, p-value = 0.7897
```

Figure 48: White Noise and ARCH Effect Test for TI Data

The Ljung-Box test results for the standardized residuals indicate that the residual series can be considered white noise (p=0.9421), and no ARCH effect is found in the test for squared standardized residuals (p=0.7897), indicating the model captures the data's volatility well.

3.7 Model Forecasting

Using the fitted ARMA(0,1)-EGARCH(1,1) model, we forecasted the next three time points. The forecast results are graphically displayed, including forecast values and confidence intervals, providing a reference for future data.

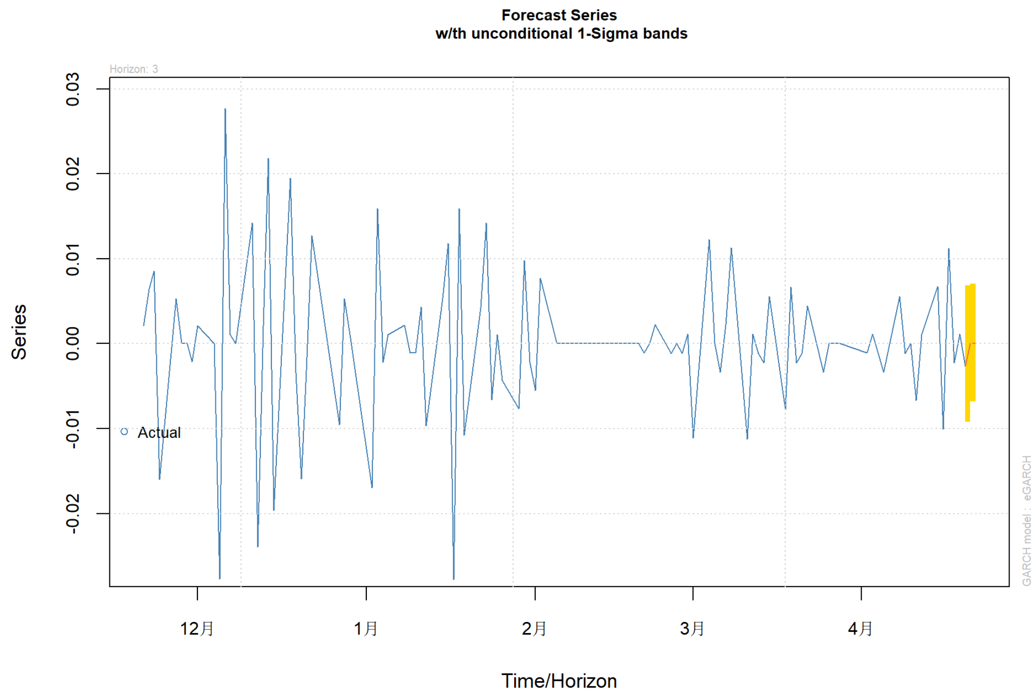

Figure 49: Forecast for TI Stock Price

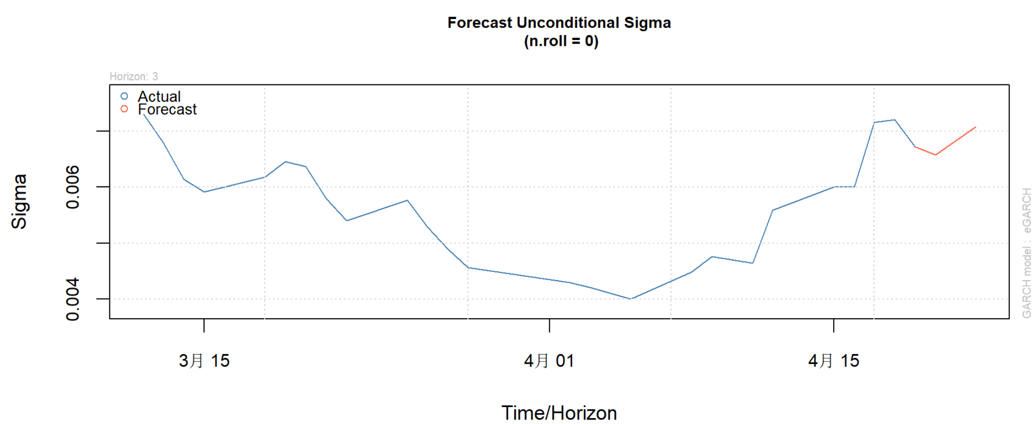

Figure 50: Detailed Forecast for TI Stock Price

4 VARMA Model Analysis of Daily Growth Rates of MTR and TI

This section discusses the analysis of the daily growth rates of MTR (Mass Transit Railway) and TI (Transport International) using the Vector Autoregressive Moving Average (VARMA) model to capture the dynamic relationships between these two time series.

Figure 51: Daily Growth Rates of MTR and TI

Analyzing daily growth rates helps stabilize variance and makes it easier for the model to capture the dynamic characteristics of the data.

4.1 White Noise Test

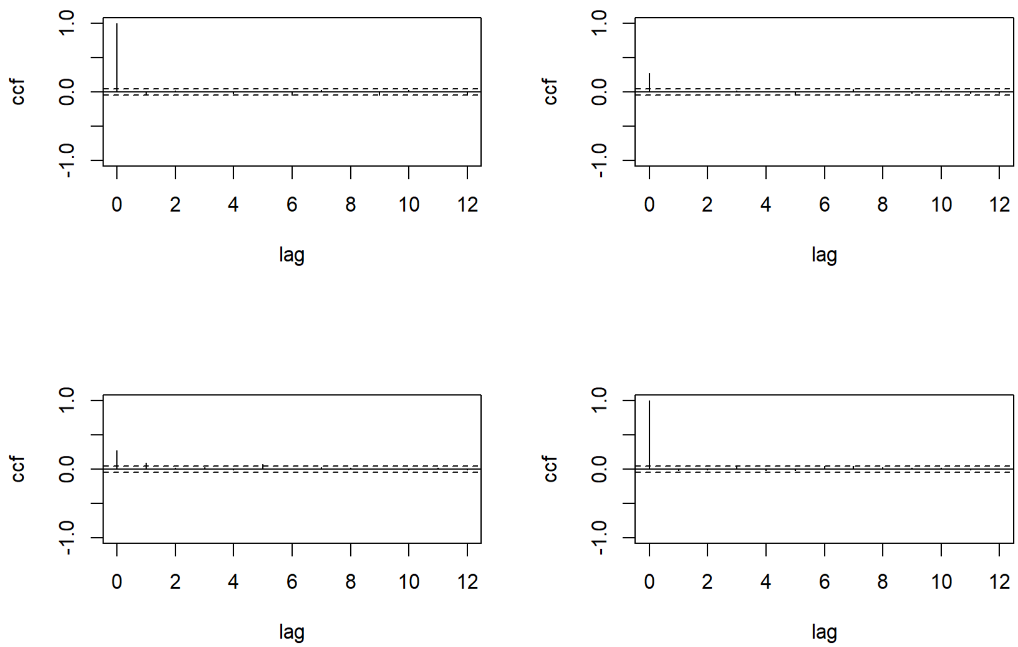

Figure 52: White Noise Test Results

A white noise test was conducted on the daily growth rates to confirm the randomness within the series.

4.2 Model Identification and Parameter Estimation

Using the `VARorder` function, the optimal lag order for the VAR model was determined. Based on the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC), recommendations were for lag orders of 7 and 1, respectively. Since AIC tends to favor more complex models and BIC simpler ones, we chose a lag order of 7 to capture more dynamic relationships.

```
> VARorder(zt, maxp = 10, output = T)
selected order: aic =  7
selected order: bic =  1
selected order: hq =  1
Summary table:
        p    AIC    BIC     HQ    M(p) p-value
 [1,]   0 0.1217 0.1217 0.1217  0.0000  0.0000
 [2,]   1 0.1107 0.1207 0.1144 32.8428  0.0000
 [3,]   2 0.1130 0.1331 0.1203  2.7280  0.6043
 [4,]   3 0.1146 0.1447 0.1256  4.3335  0.3628
 [5,]   4 0.1168 0.1568 0.1314  3.0329  0.5523
 [6,]   5 0.1108 0.1609 0.1291 21.3653  0.0003
 [7,]   6 0.1097 0.1698 0.1316 10.5503  0.0321
 [8,]   7 0.1097 0.1798 0.1352  7.9229  0.0944
 [9,]   8 0.1121 0.1923 0.1413  2.4131  0.6603
[10,]   9 0.1124 0.2026 0.1453  7.1959  0.1259
[11,]  10 0.1144 0.2146 0.1509  3.4005  0.4932
```

Figure 53: Model Selection Based on AIC and BIC

The parameters of the VAR(7) model were estimated using the `VARMA` function. Initial estimates and bounds of the parameters were provided, and the final estimates showed that most parameters are statistically significant.

```
Coefficient(s):
      Estimate  Std. Error  t value Pr(>|t|)
MTR -0.0007718   0.0237865   -0.032   0.9741
TI  -0.0125285   0.0196430   -0.638   0.5236
MTR -0.0302726   0.0217456   -1.392   0.1639
TI   0.0238697   0.0267336    0.893   0.3719
MTR  0.0137656   0.0218799    0.629   0.5293
TI  -0.0024637   0.0266896   -0.092   0.9265
MTR -0.0007261   0.0216165   -0.034   0.9732
TI   0.0302734   0.0266292    1.137   0.2556
MTR -0.0338257   0.0218831   -1.546   0.1222
TI   0.0123380   0.0266587    0.463   0.6435
MTR  0.0146816   0.0219003    0.670   0.5026
TI  -0.0562237   0.0266493   -2.110   0.0349 *
MTR -0.0472483   0.0220049   -2.147   0.0318 *
TI   0.0194376   0.0266616    0.729   0.4660
MTR  0.0174939   0.0219768    0.796   0.4260
TI   0.0362028   0.0264578    1.368   0.1712
MTR  0.0939820   0.0176873    5.314 1.08e-07 ***
TI  -0.0494566   0.0217456   -2.274   0.0229 *
MTR  0.0315721   0.0178028    1.773   0.0762 .
TI  -0.0257822   0.0217485   -1.185   0.2358
MTR  0.0214320   0.0177880    1.205   0.2283
TI   0.0226295   0.0216781    1.044   0.2965
MTR -0.0033072   0.0177917   -0.186   0.8525
TI  -0.0238682   0.0216823   -1.101   0.2710
MTR  0.0719592   0.0178131    4.040 5.35e-05 ***
TI  -0.0401833   0.0216757   -1.854   0.0638 .
MTR  0.0059029   0.0178968    0.330   0.7415
TI   0.0408924   0.0216871    1.886   0.0594 .
MTR  0.0220761   0.0178763    1.235   0.2169
TI   0.0382905   0.0215203    1.779   0.0752 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 54: Parameters of the VAR(7) Model

4.3 Parameter Significance Test

```
Number of parameters:  5
initial estimates:  -0.0562 -0.0472 0.094 -0.0495 0.072
Par. lower-bounds:  -0.1095 -0.0913 0.0586 -0.0929 0.0363
Par. upper-bounds:  -0.0029 -0.0032 0.1294 -0.006 0.1076
Final   Estimates:  -0.04057732 -0.04693625 0.09579699 -0.04789145 0.05867626

Coefficient(s):
  Estimate  Std. Error  t value Pr(>|t|)
  -0.04058     0.02465   -1.646 0.099769 .
  -0.04694     0.02020   -2.324 0.020132 *
   0.09580     0.01701    5.632 1.78e-08 ***
  -0.04789     0.02074   -2.310 0.020913 *
   0.05868     0.01643    3.572 0.000354 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
---
```

Figure 55: Parameter Significance Test

Further parameter significance testing indicated that some parameters are significant at the 95% confidence level. The model was optimized using the `refVARMA` function, reducing the number of parameters and enhancing the model's explanatory power.

4.4 Residual Testing

Residual testing was conducted using the `MTSdiag` function, and the results indicated that the residuals are insignificant at multiple lag orders, suggesting that the model captures the data's dynamics well and that the residual series is close to white noise.

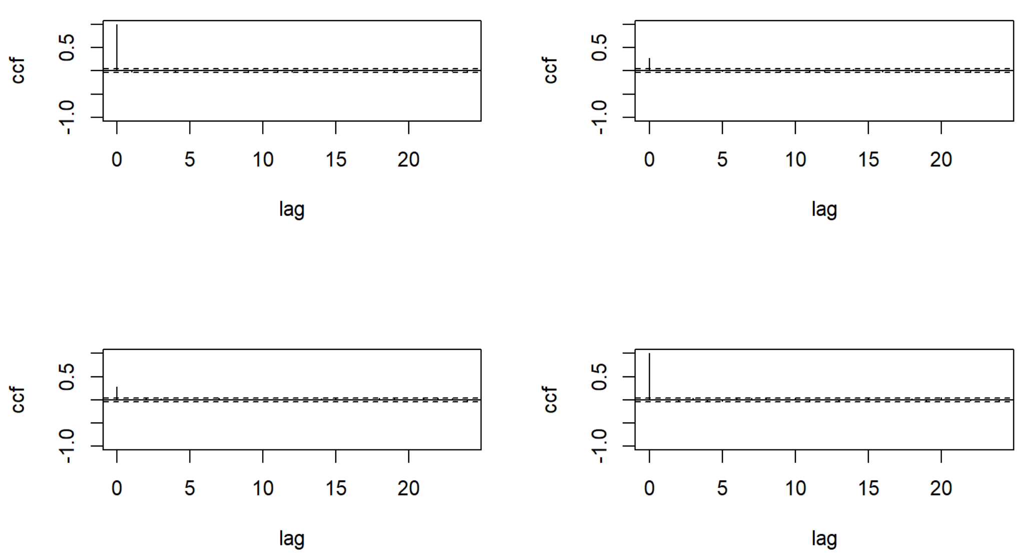

Figure 56: Residual Test Results

Residual tests were performed to check the adequacy of the model fitting. The residuals appeared to be white noise, indicating that the model captures the dynamics of the data adequately.

4.5 Model Forecasting

A four-day forecast was performed for the daily growth rates of MTR and TI. The forecast results show the potential future trends of both series, along with the standard errors, providing a quantified uncertainty for future values.

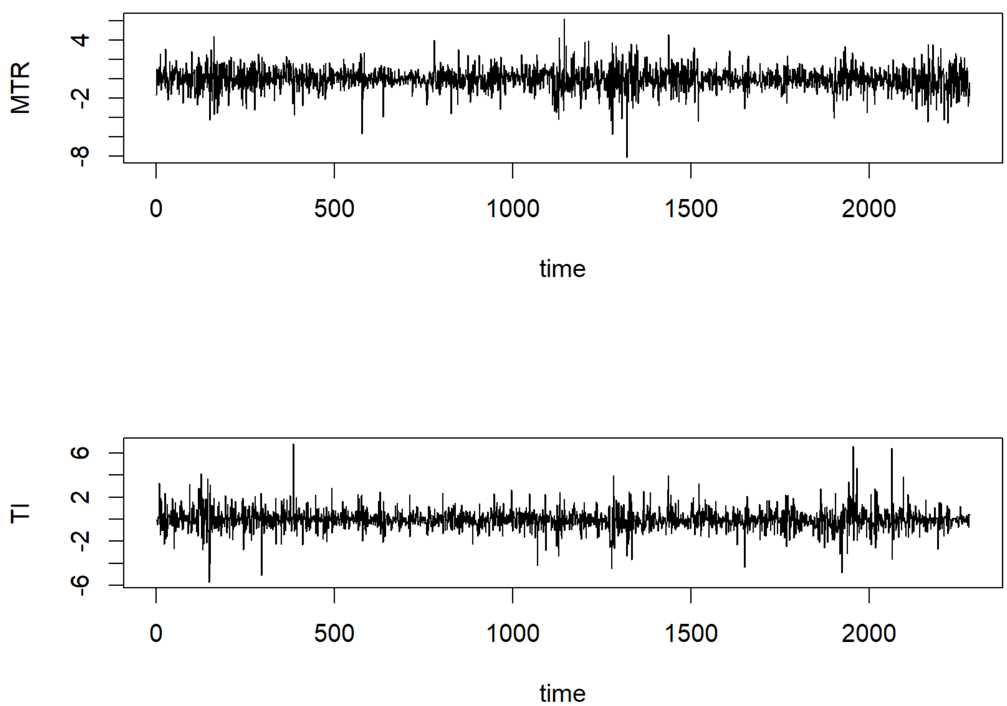

Figure 57: Forecast of Daily Growth Rates

4.6 Significance of the Study

The analysis shows that there is a significant interaction between MTR and TI. The performance of each company is not only dependent on its past performance but also on the past performance of the other company. This relationship provides valuable insights for risk management, investment strategies, and operational planning.

Future work will involve expanding the model to include additional external factors such as economic indicators and transportation policies to further understand the influences on MTR and TI's growth rates.

5 Conclusion

In this report, we analyze the daily return data of three stocks (KC, MTR and TI) and look for hidden patterns. The data of MTR and TI are differentiated once to pass the KPSS test.

For the KC stock daily returns data, we fit an ARMA(0,1)-EGARCH(1,1) model to capture the volatility structure. And for the 1 time differed data of both MTR and TI stocks' daily returns, we fit the same ARMA(0,1)-EGARCH(1,1) models as well.

Since the fluctuation trends of MTR and TI returns are similar at the image level, we can further conduct VARMA models on them. After trying the parameters of AR and MA many times, we finally selected the optimized VAR(7) as the model, the model also has good results in model evaluation and prediction.

All models mentioned above are able to pass the Stationarity Test, White Noise Test, ARCH Effect Test and Model Diagnostics.

It should be noted that the data length are selected to pass the white noise test, that is, the KC stock price in the past 4 years and the MTR/TI stock price in the past 9 years. In fact, for the stock price data before the epidemic, the practical significance and research value may not be so obvious. We only do analyses based on the ideal model. For the real application, we require more careful and meticulous processing. If we choose a monthly interest rate or a different length, the conclusions may be different.

A Appendix

A.1 Data source

Data source: Yahoo finance.

A.2 Codes

```
# File Format: gbk
# Please click File -> Reopen with Encoding... -> CP936
library(quantmod)
##############
# KC
setSymbolLookup(KC=list(name="0306.HK",src="yahoo"))
getSymbols("KC", from = "2021-01-01", to = "2024-04-20")
KC=na.omit(KC)
KC
KC=KC$"0306.HK.Adjusted"
plot(KC)

# MTR
setSymbolLookup(MTR=list(name="0066.HK",src="yahoo"))
getSymbols("MTR", from = "2015-01-01", to = "2024-04-20")
MTR=na.omit(MTR)
```

```r
MTR
MTR=MTR$"0066.HK.Adjusted"
plot(MTR)

# TI
setSymbolLookup(TI=list(name="0062.HK",src="yahoo"))
getSymbols("TI", from = "2015-01-01", to = "2024-04-20")
TI=na.omit(TI)
TI
TI=TI$"0062.HK.Adjusted"
plot(TI)


############################################## KC
library(forecast)

KC_dr = diff(KC) / lag(KC)
KC_dr = na.omit(KC_dr)
plot(KC_dr)

ndiffs(KC_dr)
length(KC_dr)

library(tseries)
pp.test(KC_dr)

for( i in c(2,5,9,11) ){
  print(Box.test(KC_dr,lag=i,type="Ljung-Box"))
}

KC_dr_at=KC_dr-mean(KC_dr)
acf(KC_dr_at^2,20,main="",col="red")
pacf(KC_dr_at^2,20,main="",col="red")
for( i in c(2,5,9,11) ){
  print(Box.test(KC_dr_at^2,lag=i,type="Ljung-Box"))
}

acf(KC_dr)
pacf(KC_dr)
library(TSA)
eacf(KC_dr)
auto.arima(KC_dr,trace = T)
```

```r

Arima(KC_dr,order = c(0,0,1), include.drift = T)$aic
Arima(KC_dr,order = c(0,0,2), include.drift = T)$aic
Arima(KC_dr,order = c(1,0,1), include.drift = T)$aic
Arima(KC_dr,order = c(1,0,2), include.drift = T)$aic
Arima(KC_dr,order = c(2,0,1), include.drift = T)$aic
Arima(KC_dr,order = c(2,0,2), include.drift = T)$aic

# Arima()
KC_md = Arima(KC_dr,order = c(0,0,1), include.drift = T)

t = abs(KC_md$coef)/sqrt(diag(KC_md$var.coef))
df_t = length(KC_dr)-length(KC_md$coef)
# pt()
pt(t,df_t,lower.tail = F)

# library(stats)
# tsdiag(KC_md)

# library(aTSA)
# tent = arima(KC_dr,order = c(0,0,1), method = 'ML')
# arch.test(tent, output = T)

library(fGarch)
library(rugarch)
KC_spec=ugarchspec(variance.model=list(model="eGARCH",garchOrder = c(1,1)),
                   mean.model=list(armaOrder=c(0,1),include.mean = TRUE),
                   distribution.model = "sstd")

KC_md_2=ugarchfit(spec=KC_spec,data=KC_dr)
KC_md_2  ### 去除不显著部分alpha1

KC_spec=ugarchspec(variance.model=list(model="eGARCH",garchOrder = c(1,1)),
                   mean.model=list(armaOrder=c(0,1),include.mean = TRUE),
                   distribution.model = "sstd", fixed.pars = c(alpha1 = 0))

KC_md_2=ugarchfit(spec=KC_spec,data=KC_dr)
```

```r
KC_md_2

# model: ARMA(0,1)-EGARCH(1,1)
# r_t =  _t + a_t
#  _t =  _0 -  _1 * a_t-1
# a_t =  _t *  _t
# ln[( _t)^2] =  _0 + [ _1*( _t-1) +  _1(| _t-1| - E| _t-1|)] +  _1*ln[( _t-1)^2]
# where  _0 = -0.001123,  _1 = -0.080011,  _0 = -1.220427,  _1 = 0,  _1 = 0.808370,  _1 = 1.00000
# skew > 0; shape < 3
#  _1 = 0

plot(KC_md_2, which = 10)
plot(KC_md_2, which = 11)

KC_stresi=residuals(KC_md_2,standardize=T)
plot(KC_stresi,type="l")
Box.test(KC_stresi,808,type="Ljung-Box",fitdf = 4) # p-value > 0.05, white noise
Box.test(KC_stresi^2,808,type="Ljung-Box",fitdf = 4) # p-value > 0.05, remains no ARCH effect

forecast = ugarchforecast(KC_md_2, n.ahead = 3, data=KC_dr)
plot(forecast, which = 1)
plot(forecast, which = 3)


############################################ MTR
MTR_dr = diff(MTR) / lag(MTR)
MTR_dr = na.omit(MTR_dr)
plot(MTR_dr)

ndiffs(MTR_dr)
MTR_dr_1 = diff(MTR_dr)
MTR_dr_1 = na.omit(MTR_dr_1)
length(MTR_dr_1)

pp.test(MTR_dr_1)

for( i in c(2,5,9,11) ){
  print(Box.test(MTR_dr_1,lag=i,type="Ljung-Box"))
```

```
}

MTR_dr_1_at=MTR_dr_1-mean(MTR_dr_1)
acf(MTR_dr_1_at^2,20,main="",col="red")
pacf(MTR_dr_1_at^2,20,main="",col="red")
for( i in c(2,5,9,11) ){
  print(Box.test(MTR_dr_1_at^2,lag=i,type="Ljung-Box"))
}

acf(MTR_dr_1)
pacf(MTR_dr_1)
eacf(MTR_dr_1)

Arima(MTR_dr_1,order = c(0,0,1), include.drift = T)$aic
Arima(MTR_dr_1,order = c(0,0,2), include.drift = T)$aic
Arima(MTR_dr_1,order = c(1,0,2), include.drift = T)$aic

MTR_md = Arima(MTR_dr_1,order = c(0,0,1), include.drift = F)

t = abs(MTR_md$coef)/sqrt(diag(MTR_md$var.coef))
df_t = length(MTR_dr_1)-length(MTR_md$coef)
# pt()
pt(t,df_t,lower.tail = F)

# library(stats)
# tsdiag(MTR_md)

# library(aTSA)
# tent = arima(MTR_dr_1,order = c(0,0,1), method = 'ML')
# arch.test(tent, output = T)

MTR_spec=ugarchspec(variance.model=list(model="eGARCH",garchOrder = c(1, 1)),
                   mean.model=list(armaOrder=c(0,1),include.mean = TRUE),
                   distribution.model = "sstd")

MTR_md_2=ugarchfit(spec=MTR_spec,data=MTR_dr_1)

MTR_spec=ugarchspec(variance.model=list(model="eGARCH",garchOrder = c(1, 1)),
                    mean.model=list(armaOrder=c(0,1),include.mean =
```

```
                        FALSE),
                     distribution.model = "sstd", fixed.pars = c(mu = 0, alpha1 = 0))

MTR_md_2=ugarchfit(spec=MTR_spec,data=MTR_dr_1)
MTR_md_2

# model: ARMA(0,1)-EGARCH(1,1)
# r_t =  _t + a_t
#  _t =  _0 -  _1 * a_t-1
# a_t =  _t *  _t
# ln[( _t)^2] =  _0 + [ _1*( _t-1) +  _1(| _t-1| - E| _t-1|)] +  _1*ln[( _t-1)^2]
# where  _0 = 0,  _1 = -0.99060,  _0 = -0.21897,  _1 = 0,  _1 = 0.97578,  _1 = 0.18286
# skew > 0; shape > 3
#  _1 = 0

plot(MTR_md_2, which = 10)
plot(MTR_md_2, which = 11)

MTR_stresi=residuals(MTR_md_2,standardize=T)
plot(MTR_stresi,type="l")
Box.test(MTR_stresi,2288,type="Ljung-Box",fitdf = 4) # p-value > 0.05, white noise
Box.test(MTR_stresi^2,2288,type="Ljung-Box",fitdf = 4) # p-value > 0.05, remains no ARCH effect

forecast = ugarchforecast(MTR_md_2, n.ahead = 3, data=MTR_dr_1)
plot(forecast, which = 1)
plot(forecast, which = 3)

############################################## TI
TI_dr = diff(TI) / lag(TI)
TI_dr = na.omit(TI_dr)
plot(TI_dr)

ndiffs(TI_dr)
TI_dr_1 = diff(TI_dr)
TI_dr_1 = na.omit(TI_dr_1)
length(TI_dr_1)

```

```
pp.test(TI_dr_1)

for( i in c(2,5,9,11) ){
  print(Box.test(TI_dr_1,lag=i,type="Ljung-Box"))
}

TI_dr_1_at=TI_dr_1-mean(TI_dr_1)
acf(TI_dr_1_at^2,20,main="",col="red")
pacf(TI_dr_1_at^2,20,main="",col="red")
for( i in c(2,5,9,11) ){
  print(Box.test(TI_dr_1_at^2,lag=i,type="Ljung-Box"))
}

acf(TI_dr_1)
pacf(TI_dr_1)
eacf(TI_dr_1)

Arima(TI_dr_1,order = c(0,0,1), include.drift = T)$aic
Arima(TI_dr_1,order = c(0,0,3), include.drift = T)$aic
Arima(TI_dr_1,order = c(1,0,3), include.drift = T)$aic
Arima(TI_dr_1,order = c(2,0,3), include.drift = T)$aic

TI_md = Arima(TI_dr_1,order = c(0,0,1), include.drift = F)

t = abs(TI_md$coef)/sqrt(diag(TI_md$var.coef))
df_t = length(TI_dr_1)-length(TI_md$coef)
# pt()
pt(t,df_t,lower.tail = F)

# library(stats)
# tsdiag(TI_md)

# library(aTSA)
# tent = arima(TI_dr_1,order = c(0,0,1), method = 'ML')
# arch.test(tent, output = T)

TI_spec=ugarchspec(variance.model=list(model="eGARCH",garchOrder = c(1,
    1)),
                    mean.model=list(armaOrder=c(0,1),include.mean =
                       TRUE),
                    distribution.model = "sstd")

```

```r
TI_md_2=ugarchfit(spec=TI_spec,data=TI_dr_1)

TI_spec=ugarchspec(variance.model=list(model="eGARCH",garchOrder = c(1,1)),
                   mean.model=list(armaOrder=c(0,1),include.mean = FALSE),
                   distribution.model = "sstd", fixed.pars = c(mu = 0, alpha1 = 0))

TI_md_2=ugarchfit(spec=TI_spec,data=TI_dr_1)
TI_md_2

# model: ARMA(0,1)-EGARCH(1,1)
# r_t =  _t + a_t
#  _t =  _0 -  _1 * a_t-1
# a_t =  _t *  _t
# ln[( _t)^2] =  _0 + [ _1*( _t-1) +  _1(| _t-1| - E| _t-1|)] +  _1*ln[( _t-1)^2]
# where  _0 = 0,  _1 = -0.96347,  _0 = -0.98647,  _1 = 0,  _1 = 0.89404,  _1 = 0.44817
# skew > 0; shape > 3
#  _1 = 0

plot(TI_md_2, which = 10)
plot(TI_md_2, which = 11)

TI_stresi=residuals(TI_md_2,standardize=T)
plot(TI_stresi,type="l")
Box.test(TI_stresi,2288,type="Ljung-Box",fitdf = 4) # p-value > 0.05, white noise
Box.test(TI_stresi^2,2288,type="Ljung-Box",fitdf = 4) # p-value > 0.05, remains no ARCH effect

forecast = ugarchforecast(TI_md_2, n.ahead = 3, data=TI_dr_1)
plot(forecast, which = 1)
plot(forecast, which = 3)

############################################## VARMA
library(mvtnorm)
library(MTS)

zt = as.matrix(cbind(MTR,TI))
```

```r
colnames(zt) = c( "MTR", "TI")
zt = diff(log(zt))*100

plot(as.xts(zt), type="l",
     multi.panel=TRUE, theme="white",
     main="日增长率")

ccm(zt)
mq(zt)

VARorder(zt, maxp = 10, output = T)
# VMAorder(zt,lag=20)
# m2=Eccm(zt,maxp=8,maxq=6)
m2=VARMA(zt,p=7,q=0)

m2b=refVARMA(m2,thres=1.96) # refine further the fit.
MTSdiag(m2b, adj=5)
# or mq
r2b=m2b$residuals
mq(r2b,adj=5)
VARMApred(m2b, h=4)
```