Analyzing Commodities: A Time Series Analysis

Caleb King

Mathematics and Statistics, Northern Arizona University, Flagstaff, Arizona

May 11, 2023

Abstract

This paper analyzes the levels of various commodities using time series analysis techniques. The study uses historical data on the prices of the commodities to explore trends, patterns, and cycles in their movements. The paper also attempts to forecast future values for each commodity using time series models. The analysis provides insights into the behavior of commodity prices over time, and how they are affected by various factors. The results show that time series analysis can be an effective tool for analyzing commodity markets and making predictions about future price movements. Overall, the findings have important implications for investors, policymakers, and other stakeholders interested in understanding and managing commodity price risk.

KEY WORDS: SARIMA, Forecast, ACF/PACF, Stationarity, Normality.

1 Introduction

A commodity is a raw material that is used in commerce for trading, can be bought or sold for a variety of other economic contributors. Commodities play a vital role in the global economy, serving as essential inputs in various industries and acting as indicators of economic health. Understanding the behavior of commodity prices over time is crucial for investors, policymakers, and market participants to make informed decisions. The analysis of commodity levels using time series techniques and models offers valuable insights into the dynamics of these markets, including identifying trends, patterns, and cycles, as well as forecasting future price movements.

I decided to start my research on the effectiveness of time series analysis in studying commodity prices, employing various models and methods to uncover meaningful patterns. I wanted these studies to examine the behavior of different commodities, ranging from energy resources such as oil and natural gas to agricultural products like wheat and corn. By capturing the inherent time dependencies and autocorrelations within the data, time series analysis proved to provide a powerful framework for analyzing commodity price movements.

In this study, I aimed to contribute to the existing body of knowledge by conducting a comprehensive analysis of commodity levels using time series analysis. Specifically, I wanted to focus on examining historical data spanning for more than fifty years for a selected set of commodities, employing advanced time series models to understand their behavior and make predictions about their future values. This study would allow me to understand what kind of role these commodities have in the global market today. Analyzing these levels can help me in the future with diversification in an investment portfolio to see if commodities can be predicted and are a good diversification factor in any investment portfolio.

By studying the long-term trends, cyclicality, and volatility of commodity prices, we can shed light on the potential risk and return characteristics associated with these markets. Such insights are crucial for investors seeking to diversify their portfolios and manage commodity price risk. Moreover, policymakers can leverage these findings to make informed decisions regarding commodity price stabilization measures and supply chain management. Being able to predict these prices can prove useful to almost anyone who wants to involve themselves in investments.

2 The Data

I decided to use data from the World Bank for this project. The data for the World Bank is collected from a wide range of sources, including surveys, censuses, administrative records, and other publicly available data sources. The World Bank is a widely recognized and respected organization, known for its expertise in development and its commitment to producing high-quality and reliable data. The data provided is widely used by researchers, policymakers, and practitioners around the world to inform their work, and is considered to be among the most comprehensive and reliable sources of development data available.

For data collection I first had to obtain the correct dataset. I decided to use monthly prices of all the commodities my data set had to offer. Using monthly data compared to annual data offered me a lot more information I could use to make better analysis but would include seasonality. Seasonality refers to a pattern in time series data where the values tend to exhibit regular and predictable fluctuations that recur over a specific period, such as a year, quarter, month, week, or day. It is important to get rid of seasonality in time series data because it can obscure the underlying trend and make it difficult to identify the true drivers of the data.

With the use of Rstudio, I was able to read the large set of data and translate it into a time series format allowing me to make reliable analysis on the data. I decided to logarithmically transform the data because it helped fix a normality issue that arose when obtaining the results. The following plot (Figure 1.1) is a visualization of the time series for the World Bank commodity index (excluding precious metals).

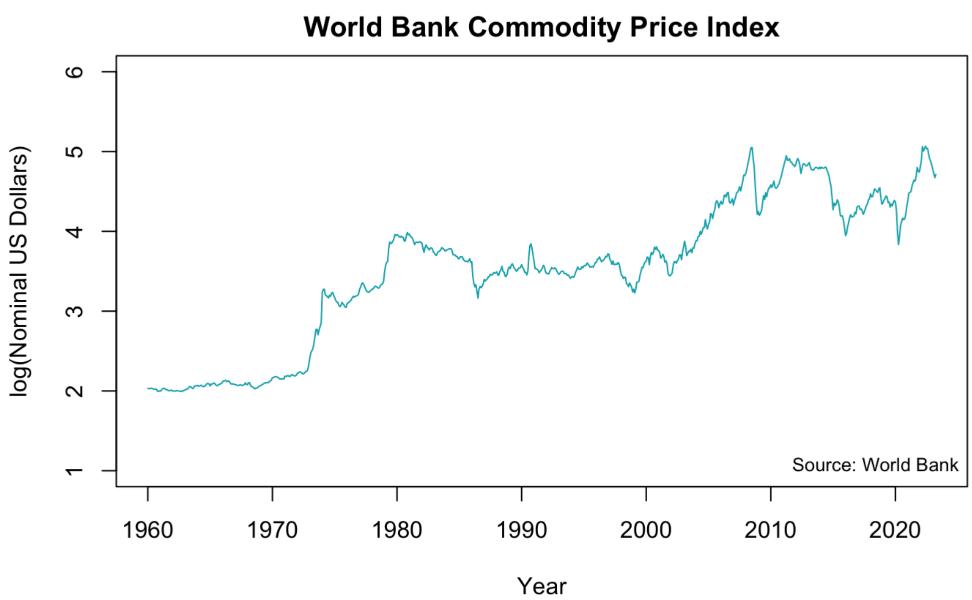

Figure 1.1: Time series plot of World Bank Commodity Monthly Price Index over Time

Since the data does not include any precious metal prices, I decided to look at that separately as well. The time plot (Figure 1.2) for the price of platinum gold and silver can be seen below.

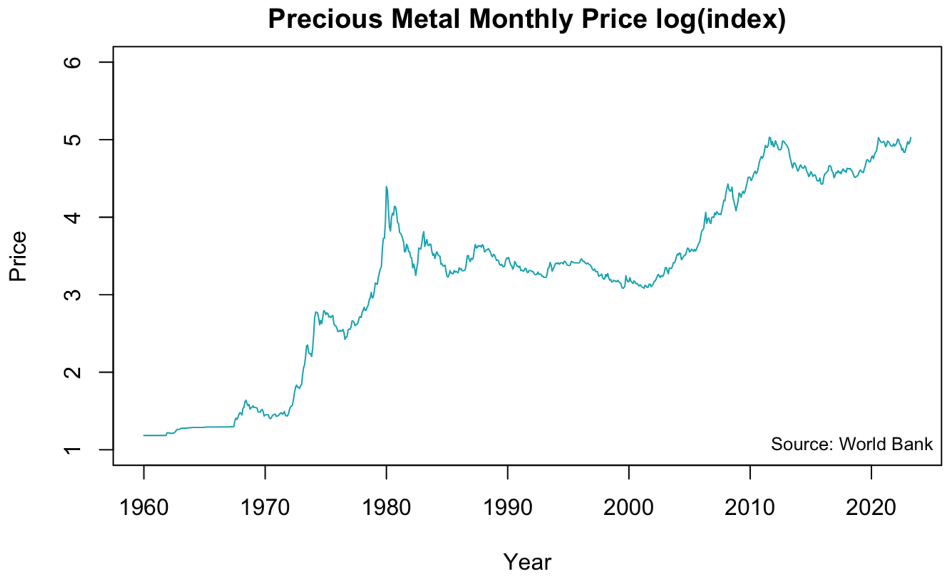

Figure 1.2: Time series plot of Precious Metal Monthly Price Index over Time

3 Methods

The first step in the process of analyzing this dataset was to utilize the Augmented Dickey-Fuller test to see if the transformed data is stationary. Stationarity is a crucial assumption in time series analysis, and ensuring that the data is stationary is important for accurate modeling, forecasting, and interpretation of the data. Specifically, the ADF test is used to analyze whether a time series has a unit root or not, which is a characteristic of non-stationarity. The ADF test is based on the following autoregressive model:

$$y[t] = \rho y[t-1] + \delta t + \varepsilon[t] \qquad (1)$$

In (1), y[t] represents the value of the time series at time t, ρ is the coefficient of the first lag of the time series, δ is a constant term, and $\varepsilon[t]$ is a random error term. The null hypothesis of the ADF test is that $\rho = 1$, which implies that the time series has a unit root and is non-stationary.

The next step of analysis after performing the Augmented Dickey-Fuller Test was to remove seasonality from the data in order to make the analysis as precise as it could be. The method I chose to do was to use seasonal differencing, which involves taking the difference between the value at time t and the value at time t minus the length of the seasonal cycle (e.g., 12 for monthly data). After removing the seasonality component from my data set I needed to test if there was a linear trend in my data. For this I performed the Augmented Dickey-Fuller Unit Root Test which assumes that the time series has a first-order autoregressive (AR(1)) model.

Once these tests were completed it was time for me to fit an appropriate model to the data. Since I was working with seasonal data, it was clear that I would have to fit a seasonal autoregressive integrated moving average (SARIMA) model to the data. The general form of a SARIMA model is:

$$\text{SARIMA}(p,d,q)(P,D,Q)m \tag{2}$$

In (2), ‘p’ represents the order of the autoregressive (AR) component of the model, the symbol ‘d’ represents the order of differencing required to make the data stationary, and ‘q’ represents the order of the moving average (MA) component of the model. Additionally, the symbol ‘P’ represents the order of the seasonal autoregressive (SAR) component of the model, ‘D’ represents the order of seasonal differencing required to make the data stationary, ‘Q’ represents the order of the seasonal moving average (SMA) component of the model, and ‘m’ represents the number of time periods in each season.

From this SARIMA model I was able to find the best fitting model from comparing the AICC and BIC values from the different models I tried. The model with the lower AICC or BIC value is preferred because it indicates a better balance between model fit and complexity, leading to potentially more reliable and generalizable results. After finding the best fitting model, I performed additional tests to confirm that the model I had created would be the best fitting model. These tests included the Ljung-Box test, residual inspection, Shapiro-Wilk, and Kolmogorov-Smirnov tests.

Once I concluded that I had a good model I decided to forecast these commodity prices for the next 14 months. I did this utilizing the predict() command in Rstudio accompanied with the best fit model. I decided on adding upper and lower 95% bounds to forecast values, which provides a useful measure of uncertainty and helps to communicate the range of possible outcomes. Overall, using these methods proved useful to obtain the best fitting model for analyzing and forecasting commodity data over time.

4 Results

All numerical results, tables and plots were derived from my analysis using Rstudio. From figure 1.1 and 1.2, the data would seem to be stationary just by a visual diagnostic. I wanted to take it a step further and test for stationarity with the Augmented Dickey-Fuller Test (1). Stationarity was true for all of the commodities tested which included precious metal, non-energy, and energy commodities. Shown below is the ADF test on precious metals (Figure 2.1)

```
	Augmented Dickey-Fuller Test

data:  metalMonth.t
Dickey-Fuller = -2.5192, Lag order = 12, p-value = 0.3585
alternative hypothesis: stationary
```

Figure 2.1: Precious Metals ADF Test

```
        Augmented Dickey-Fuller Test

data:  nonfuelMonth.t
Dickey-Fuller = -2.8793, Lag order = 12, p-value = 0.2061
alternative hypothesis: stationary
```

Figure 2.2: Non-Energy ADF Test

```
        Augmented Dickey-Fuller Test

data:  energyMonth.t
Dickey-Fuller = -2.1163, Lag order = 12, p-value = 0.5291
alternative hypothesis: stationary
```

Figure 2.3: Energy ADF Test

Since the test for stationarity is completed, it is important to take the sample autocorrelation function (ACF) and partial autocorrelation function (PACF). This test can confirm if there is seasonal data from a visualization of these functions. Figure 2.4 shows the result from the data on non-energy commodities.

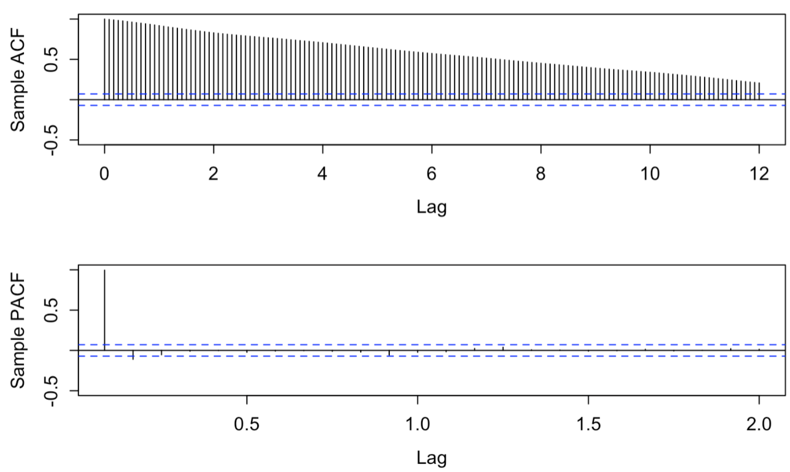

Figure 2.4: Sample ACF and PACF for Metals

After determining seasonality it is necessary to take the lag-12 differencing if the data exhibits seasonality. Figure 2.5 and 2.6 show the visualization I used for this lag 12 differenced data for precious metals and energy.

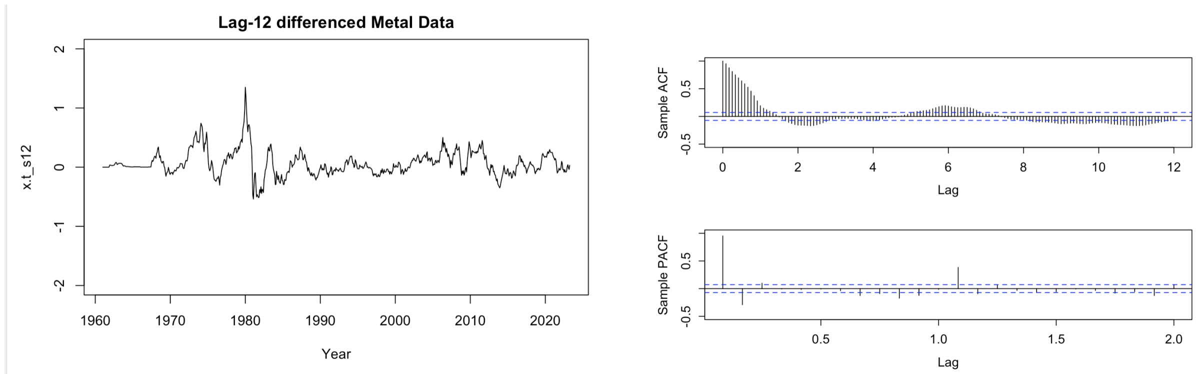

Figure 2.5: Lag-12 Visualization for Metal data

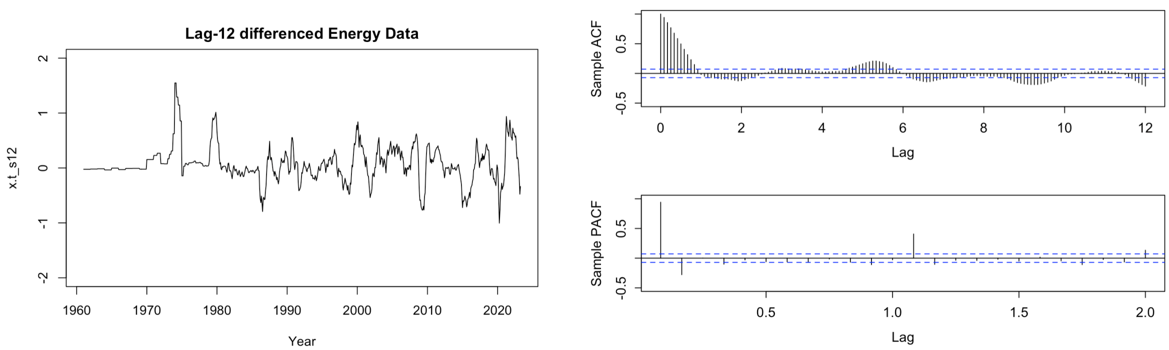

Figure 2.6: Lag-12 Visualization from Energy Data

For a result that goes past just visualization I decided to take a numerical summary of the lag-12 differenced data. Figures 2.7, 2.8 and 2.9 show this numerical summary:

```
[1] 0.05991024
[1] 0.2102751
[1] 748
```

Figure 2.7: Precious Metal Numerical summary

```
[1] 0.02977037
[1] 0.1431598
[1] 748
```

Figure 2.8: Non-Energy Numerical Summary

```
[1] 0.06702503
[1] 0.321825
[1] 748
```

Figure 2.9: Energy Numerical Summary

Since I tested for seasonality in the data, next I would like to test for linear trend to see if it is appropriate to take the lag-1 differencing of the lag-12 differenced data. Figure 2.10 shows the L on non-energy commodities:

```
Value of test-statistic is: -3.3725 5.6995

Critical values for test statistics:
      1pct  5pct 10pct
tau2 -3.43 -2.86 -2.57
phi1  6.43  4.59  3.78
```

Figure 2.10: Unit Root test on Non-Energy

Once the computation for linear trend is tested, it is time to start forming an appropriate for the data. I did this using the ARIMA function for seasonal data and applying the correct number in order to find the best fitting model. Figures 2.11, 2.12, 2.13 show the models that best fit with precious metals, non-energy and energy commodities:

```
Call:
arima(x = metalMonth.t, order = c(1, 0, 1), seasonal = list(order = c(2, 1,
    0), period = 12))

Coefficients:
         ar1     ma1     sar1     sar2
      0.9727  0.3041  -0.5884  -0.3429
s.e.  0.0084  0.0360   0.0346   0.0342

sigma^2 estimated as 0.0027:  log likelihood = 1146.51,  aic = -2283.03
```

Figure 2.11: Best Fitting SARIMA Model for Precious Metals

```
Call:
arima(x = nonfuelMonth.t, order = c(2, 0, 1), seasonal = list(order = c(2, 1,
    0), period = 12))

Coefficients:
         ar1      ar2      ma1     sar1     sar2
      1.5069  -0.5254  -0.1110  -0.6208  -0.4088
s.e.  0.0709   0.0700   0.0833   0.0341   0.0341

sigma^2 estimated as 0.0007092:  log likelihood = 1645.57,  aic = -3279.13
```

Figure 2.12: Best Fitting SARIMA Model for Non-Energy

```
Call:
arima(x = energyMonth.t, order = c(2, 0, 0), seasonal = list(order = c(2, 1,
    0), period = 12))

Coefficients:
         ar1      ar2     sar1     sar2
      1.2161  -0.2455  -0.6372  -0.3877
s.e.  0.0356   0.0356   0.0341   0.0340

sigma^2 estimated as 0.007143:  log likelihood = 782.26,  aic = -1554.53
```

Figure 2.13: Best Fitting SARIMA Model for Energy

Summarizing the residuals is the next important part of the process. Figure 2.14 shows the mean and standard deviation of the residuals:

```
[1] 0.002542782
[1] 0.05193851
      Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
-0.1972862 -0.0231462  0.0004647  0.0025428  0.0261401  0.3759608
```

Figure 2.14: Numerical Summary of Residuals for Precious metals

I also decided to include some visualization for these residuals as figure 2.15 shows below:

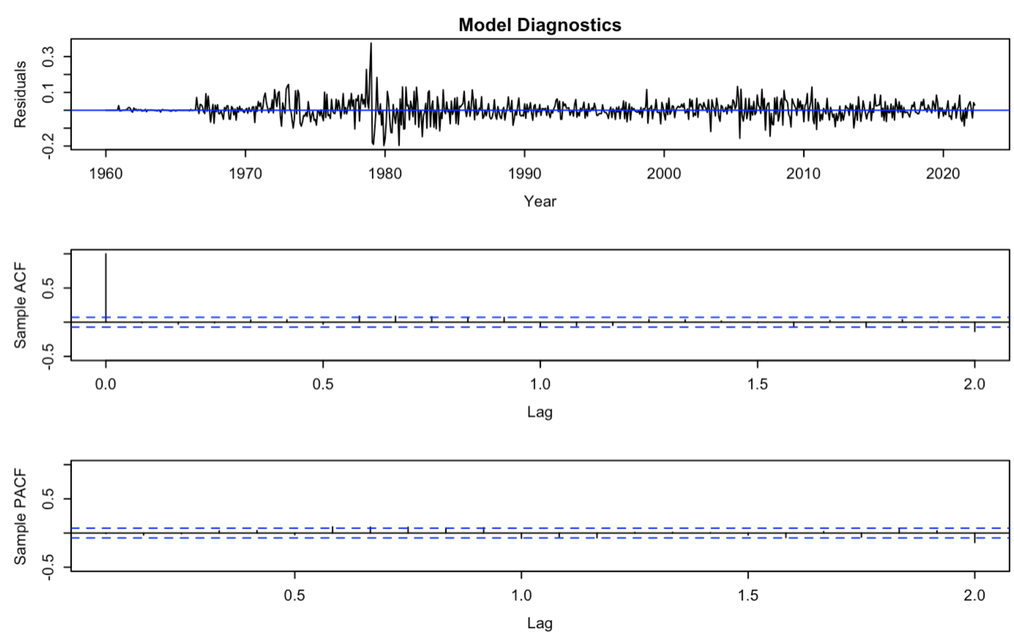

Figure 2.15: Model Diagnostics for Precious Metals

For an additional test for the model I decided to perform the Ljung-Box Test. The Ljung-Box test is a useful tool for assessing the presence of significant autocorrelation in time series data and for evaluating the adequacy of time series models.

```
        Box-Ljung test

data:  w.t
X-squared = 22.523, df = 10, p-value = 0.01265

X-squared
    0.001
```

Figure 2.16: Ljung-Box Test for Precious Metals

```
        Box-Ljung test

data:  w.t
X-squared = 15.252, df = 10, p-value = 0.1231

X-squared
   0.0093
```

Figure 2.17: Ljung-Box Test for Non-Energy

```
        Box-Ljung test

data:  w.t
X-squared = 9.6764, df = 10, p-value = 0.4693

X-squared
    0.139
```

Figure 2.18: Ljung-Box Test for Energy

To ensure I fit the best model, I decided to check the normality. I did this with a histogram of the residuals, normal Q-Q plot, Shapiro-Wilk and Kolmogorov-Smirnov tests. Figures 2.19, 2.20, and 2.21 display the results of these tests I performed.

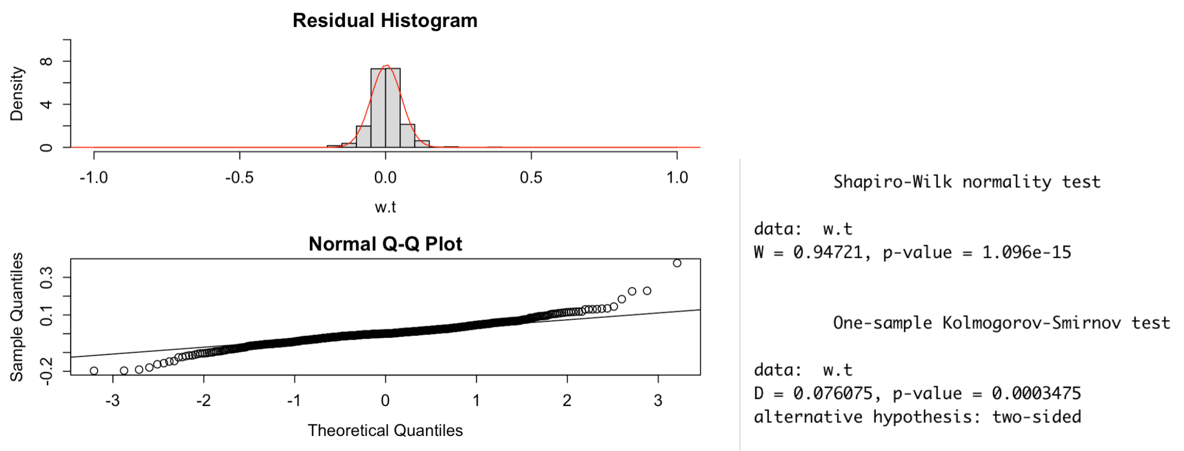

Figure 2.19: Normality Check for Precious Metals

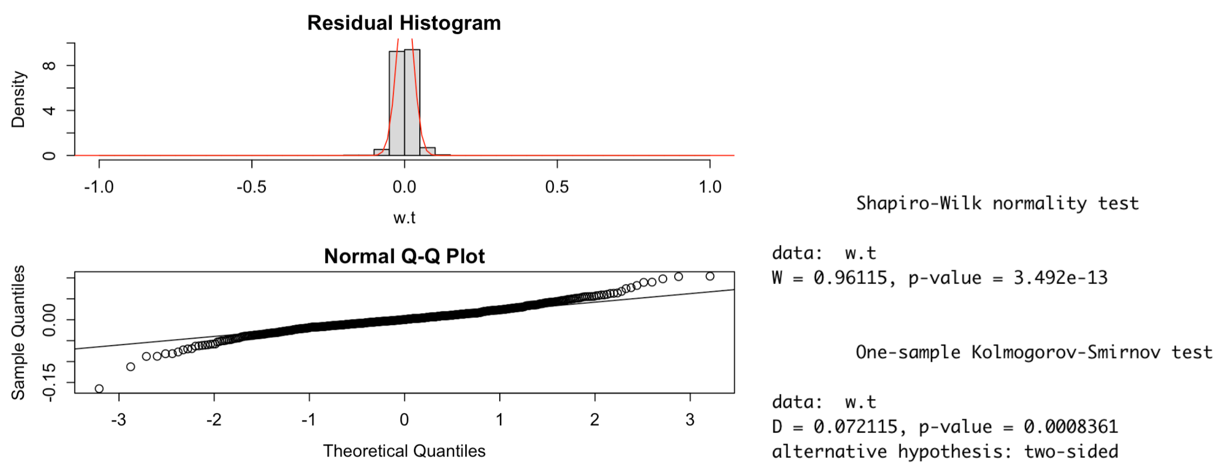

Figure 2.20: Normality Check for Non-Energy

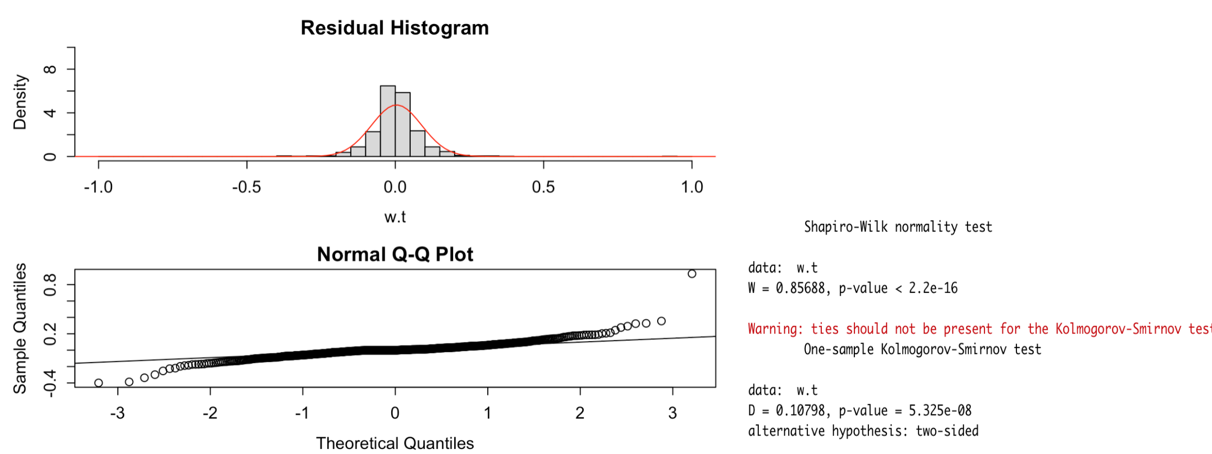

Figure 2.21: Normality Check for Energy

Finally after understanding the results fully and how it correlates with the data, it is time to forecast future values. Figures 2.22, 2.23, 2.24 forecast future values using the best fitting model that was created:

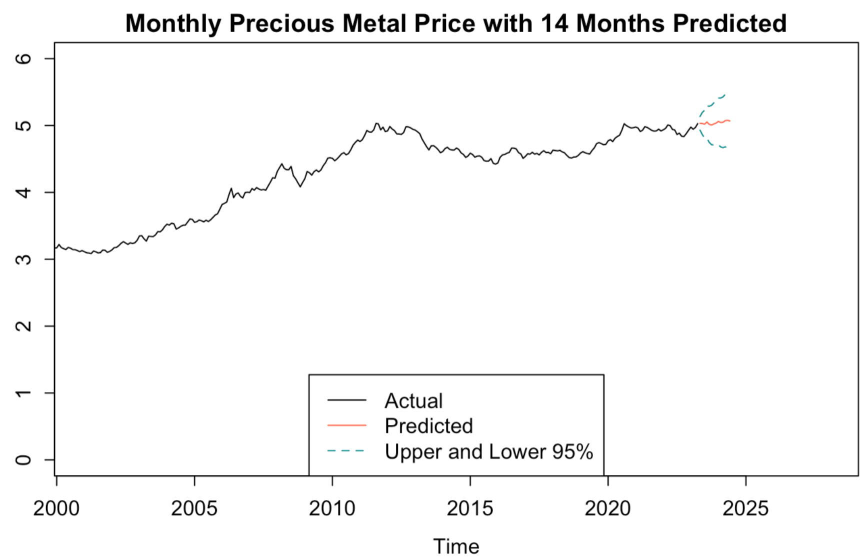

Figure 2.22: Monthly Precious Metal Price with 14 Months Predicted

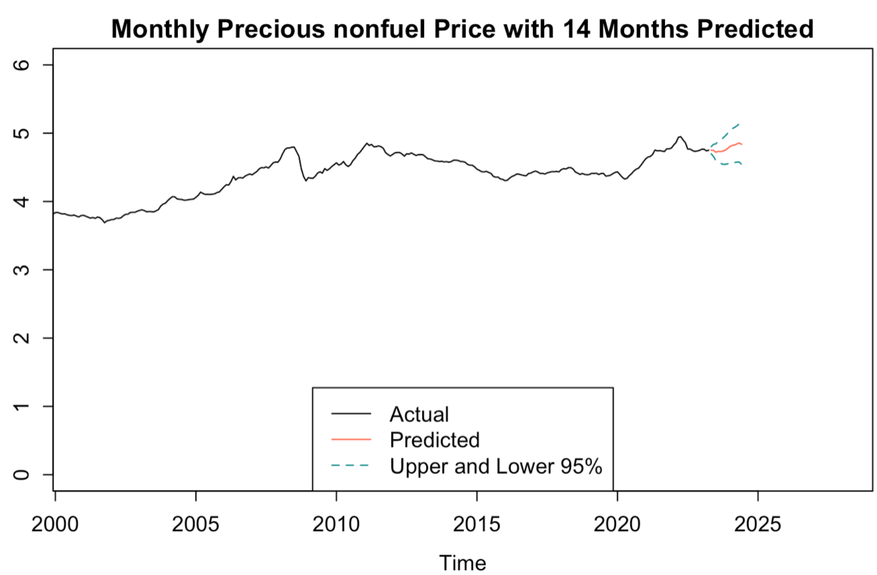

Figure 2.23: Monthly Nonfuel Price with 14 Months Predicted

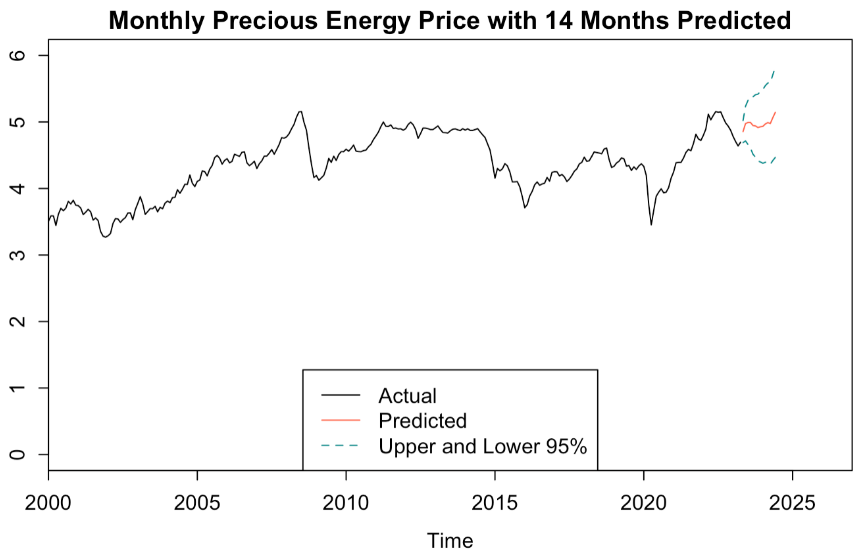

Figure 2.24: Monthly Energy Price with 14 Months Predicted

5 Conclusions and Discussions

From testing the stationarity in figures 2.1, 2.2 and 2.3 it is clear that the log transformed data meets the stationarity criteria. Each of the p values are pretty low and the Dickey-Fuller values all range around the same area. This is good to know because it can confirm that this is good data to build a model off of with the methods discussed previously. When looking at figure 2.4 we can see slow decay in the sample ACF and exponential decay in the sample PACF. The slowly decaying ACF suggests that there is some autocorrelation in the time series data. The immediate decay of the PACF suggests that the effects of intermediate lags can be adequately explained by the earlier lags, indicating that the AR model may be appropriate. These plots also show that there is seasonality in the time series data. This was the same for the three commodities I analyzed so I decided to include only one figure for this visualization.

Since there is seasonality in the data it is appropriate to take the lag-12 differencing of the data in order to remove seasonality. From figure 2.5 and 2.6, the data seems to be close to zero except for certain years. The sample ACF and PACF of the data have some significant spikes after lag 1 and some variation in the ACF. In figures 2.7, 2.8 and 2.9 I can see that the first value of each commodity is close to zero. It is also apparent that the lag 12 differencing was taken because the sample size of the data went from 760 observations to 748. In figure 2.10 I decided to take the Augmented Dickey-Fuller Unit Root Test on the differenced data. For each of the commodities, removing the linear trend was not necessary. Specifically, for figure 2.10, because the ADF test statistic value -3.3725 is less than the 5th percentile critical value -2.86, at alpha=0.05 significance level, we do reject the null hypothesis H0 : the series x.t has a unit root. Therefore, it is not supported to apply the lag-1 differencing to the lag-12 differenced series nonfuelMonth.t.

After testing for linear trend, it is time to start forming a sufficient model for the time series data. I decided to use a SARIMA model because the data was seasonal and I am most familiar with ARIMA models. In figure 2.11, the best fitting model for precious metal prices of time was SARIMA(1,0,1)(2,1,0)12. In figure 2.12 the best fitting model for Non-Energy commodities was a SARIMA (2,0,1)(2,1,0)12. Finally in figure 2.13 the best fitting model for Energy commodities was a SARIMA (2,0,0)(2,1,0)12. I found it interesting that each of these commodities had a different ARIMA model. The precious metals model includes a non-seasonal AR term at lag 1, a non-seasonal MA term at lag 1, and two seasonal AR terms at lags 12 and 24, with one seasonal difference at lag 12. The non-energy model includes a non-seasonal AR term at lags 1 and 2, a non-seasonal MA term at lag 1, and two seasonal AR terms at lags 12 and 24, with one seasonal difference at lag 12. The energy model includes a non-seasonal AR term at lags 1 and 2, and two seasonal AR terms at lags 12 and 24, with one seasonal difference at lag 12. There are no MA terms in this model.

Analyzing the numerical summary and model diagnostics visualization of the residuals, I could further prove my model is the best fit. In figure 2.14 you can see that the mean of the model is close to zero and the standard deviation of the model is close to sigma. This was true for all the commodities I analyzed so I decided to just keep one figure for all the commodities. Next the model diagnostics and visualization of the residual turned out well. You can see that the residual plot stays around zero. For the sample ACF and PACF it is apparent that there are no significant lags except slightly at lag 2. From looking at figures 2.16, 2.17 and 2.18, you can see that each model passes the Ljung-Box test. In figures 2.19, 2.20 and 2.21 the normality checks were mostly met. My p values for precious metals and non energy commodities was close to zero, which assumes normality. As for my p value with energy commodities the p value was much higher than zero. I assume this has to be a problem with the data itself because of the warning I received from this test. But from looking at the residual histogram, Q-Q plot and the other normality test, I conclude that the data I analyzed was normal.

Since the models passed the tests that I made I decided to use the model to forecast prices 14 months into the future. This can be seen in figures 2.22, 2.23, 2,24. The predicted lines seem to follow the trend of the previous values but the upper and lower 95% confidence intervals sway a bit from this predicted line. I figure this has to do with some of my issues with normality as there are some times when prices skyrocketed and plummeted. Predicting for those scenarios can be quite difficult but barring any major fluctuation, these models help for predicting future prices. Overall, the results of this analysis provide valuable insights into the trends of commodity prices over time and demonstrate the usefulness of time series analysis techniques in forecasting future commodity prices.

Appendix: R Codes

```
---

title: "BankAnalysis"

author: "Caleb King"

date: "2023-04-29"

output: pdf_document

---


```{r setup, include=FALSE}

knitr::opts_chunk$set(echo = TRUE)

```


Get data set up for Commodity 1960-2023

Set up

```{r, warning=FALSE}

# Set working directory

setwd("/Users/caleblking/Desktop/finalTime/")
```
```

```
library(readxl)

library(tseries)

library(forecast)

library(urca)
```

Import data from World Bank

```{r}
dataMonth <- read_excel("CMO-Historical-Data-Monthly.xlsx",

            sheet = "Monthly Indices",

            skip = 9)
```

Get data for commodities needed

Monthly Precious bank

```{r}
```

````
head(dataMonth$...2)

```

Translate to time series format

Monthly Precious bank

```{r}

# Create a time series of Precious metal monthly level

bankMonth.t <- ts( dataMonth$...2,start=1960,freq=12 )

#transform data

bankMonth.t <- log(bankMonth.t)

tail(bankMonth.t)

```

Numerical summary

Sample size

```{r}
````

Sample size

n <- length(bankMonth.t)

n

```

Mean and standard deviation

```{r}

mean(bankMonth.t)

sd(bankMonth.t)

```

Five-number (minimum, 1st quartile, median, 3rd quartile, maximum) summary

```{r}

# Five-number (minimum, 1st quartile, median, 3rd quartile, maximum) summary and mean

summary( bankMonth.t )

```
```

Visualization

We use a time plot to visualize a time series

Nominal US dollar term

```{r}

# Time plot using plot.ts()

plot.ts(bankMonth.t, ylim=c(1,6), col="#00AFBB", xlab="Year",

ylab="log(Nominal US Dollars)", main="World Bank Commodity Price Index")

legend("bottomright", legend="Source: World Bank", bty="n", cex=0.8,

col="gray40")

```

Test for stationary or not

```{r}

adf.test(bankMonth.t, k=12)

```

Sample ACF and PACF of time series data

```{r}
# Sample ACF and PACF
par( mfrow=c(2,1),mex=0.75 )
acf( bankMonth.t,lag.max=144,ylim=c(-0.5,1),ylab="Sample ACF",main="" )
pacf( bankMonth.t,lag.max=24,ylim=c(-0.5,1),ylab="Sample PACF",main="" )
```

Taking the differencing for seasonality because it is monthly data

```{r}
# Lag-12 differenced series (Note: the sample size is decreased by 12 due to lag-12
differencing)
x.t_s12 <- diff(bankMonth.t, 12)

# Time plot of lag-12 difference series
plot.ts( x.t_s12,ylim=c(-2,2),xlab="Year",main="Lag-12 differenced series" )

# Sample ACF and PACF of lag-12 differenced series
par( mfrow=c(2,1),mex=0.75 )
```

````
acf( x.t_s12,lag.max=144,ylim=c(-0.5,1),ylab="Sample ACF",main="" )

pacf( x.t_s12,lag.max=24,ylim=c(-0.5,1),ylab="Sample PACF",main="" )
```
````

Numerical Summary of Lag-12 differenced data

````
```{r}
# Numerical summary of lag-12 differenced series
x.t_s12.mean <- mean( x.t_s12 )
x.t_s12.sd <- sd( x.t_s12 )
n_s12 <- length( x.t_s12 )

x.t_s12.mean
x.t_s12.sd
n_s12

summary( x.t_s12 )
```
````
````

Test for Linear trend on differenced data

```{r}
# Augmented Dickey-Fuller test for unit roots
adf.out_0 <- ur.df( x.t_s12,type="drift",lags=0 )        # Augmented Dickey-Fuller test with p=0 ("lags=0")
summary( adf.out_0 )
```

Because the ADF test statistic value -4.0586 is less than the 5th percentile critical value -2.86, at alpha=0.05 significance level, we do reject the null hypothesis H0 : the series x.t has a unit root. Therefore, it is not supported to apply the lag-1 differencing to the lag-12 differenced series bankMonth.t.

Function to compute AICC

```{r}
# Function to compute AICC
arma.AICC <- function(fit) {
  n.par <- length(fit$coef)-sum((fit$coef==0))     # no of parameters in ARMA fit, corrected for sub-models
  n <- fit$nobs                                    # sample size
```

```
  aicc <- -2*fit$loglik+2*(n.par+1)*n/(n-n.par-2)  # AICC
  return(aicc)
}
```

Fit an appropriate model

```{r}
auto.arima(bankMonth.t, D = 1)

fit_101.210 <- arima(bankMonth.t,order=c(2,0,0),seasonal=list(order=c(2,1,0),
period=12))
fit_101.210
```

Analyze the BIC and AICC values (smaller values are better)

```{r}
arma.AICC(fit_101.210)
BIC(fit_101.210)
```

```

Create variable for the best fitting data

```{r}

fit.bst <- fit_101.210

```

Compute residuals and summary of residuals

```{r}

# Residuals

w.t <- ts( fit.bst$resid[13:n],start=c(1960,1),freq=12 )

head( w.t )

# Numerical summary of residuals

w.mean <- mean(w.t)                         # should be close to zero

w.sd <- sd(w.t)                         # should be close to sigma

w.mean

w.sd

summary(w.t)

```

Model diagnostics

```{r}
# Model diagnostics: visual inspection
par( mfrow=c(3,1),mex=0.75 )
plot.ts(
w.t,ylim=c(min(w.t),max(w.t)),xlab="Year",ylab="Residuals",main="Model
Diagnostics" )
abline( h=0,col="blue" )
acf( w.t,lag.max=24,ylim=c(-0.5,1),ylab="Sample ACF",main="" )
pacf( w.t,lag.max=24,ylim=c(-0.5,1),ylab="Sample PACF",main="" )
```

Ljung-Box test
```

```{r}
# Model diagnostics: Ljung-Box portmanteau test (repeat with different m)
m <- 10                                         # Ljung-Box portmanteau test's "m"
n.par <- length(fit.bst$coef)                        # number of parameters in fitted
ARMA(p,q), p+q
LB.out <- Box.test(w.t,lag=m,type="Ljung-Box")              # Ljung-Box
portmanteau test
LB.out                                          # Box.test uses wrong df if applied to
residuals
LB.pval <- pchisq(q=LB.out$statistic,df=m-n.par,lower.tail=FALSE) # correct
p-value
round( LB.pval, 4 )
```

Normality Check

```{r}
# Model diagnostics: normality check
par(mfrow=c(2,1),mex=0.75)
hist(w.t,freq=FALSE,                                # histogram of residuals
```

````
    breaks=seq(-1,1,0.05),
    col="grey85",ylim=c(0,10),
    main="Residual Histogram")
z <- seq(-8,8,length=1000)
lines(z,dnorm(z,mean=w.mean,sd=w.sd),lty=1,col="red")          # add theoretical
normal density
qqnorm(w.t)                                   # normal Q-Q plot
qqline(w.t)
```
````

Normality test (Shapiro-Wilk and Kolmogorov-Smirnov tests)

````
```{r}
# Normality test
shapiro.test(w.t)                             # Shapiro-Wilk normality test
ks.test(w.t,"pnorm",mean=w.mean,sd=w.sd)                #
Kolmogorov-Smirnov normality test
```
````
````

Future value prediction (Forecast)

```{r}
# Forecast 12 future values of original series x.t with 95% PI (only when model diagnostics is passed)
x.fct <- predict( fit.bst,n.ahead=14 )
x.L95 <- x.fct$pred-1.96*x.fct$se
x.U95 <- x.fct$pred+1.96*x.fct$se


par( mfrow=c(1,1), mex=0.75 )
ts.plot( bankMonth.t,x.fct$pred,
col=c("black","tomato"),ylim=c(0,6),xlim=c(2001,2028),
        main="Monthly World Bank Price with 14 Months Predicted" )
lines( x.U95,col="cyan4",lty="dashed" )
lines( x.L95,col="cyan4",lty="dashed" )
legend( "bottom", lty=c("solid","solid","dashed"),
       legend=c("Actual", "Predicted", "Upper and Lower 95%"), col=c("black",
"tomato", "cyan4"))
```

References

World Bank. “Commodity Markets.” Commodity Prices, 4 May 2023, www.worldbank.org/en/research/commodity-markets.