ECON 144 Project 2

Ragini Srinivasan, Jiarui Song, Siyu Xiong, Adriana Villavicencio

2023-11-16

I. INTRODUCTION

The data we chose was the price of crude oil (WTI - Cushing, Oklahoma) and conventional gasoline (U.S. Gulf Coast, Regular) – monthly, not seasonally adjusted. Specifically, the observations range from January of 1987 through October of 2023. We obtained this data from the U.S. Energy Information Administration. We chose this data because it worked well with our VAR analysis and because it contained all 3 components of trend, seasonality, and cycles. It had an upward trend, a possible (but not certain) multiplicative seasonality, and an autoregressive process.

From past knowledge, we can recognize that the prices of gas and oil may have a positive relationship as crude oil is the primary material source for producing gasoline. Due to the recent increase in demand for crude oil (and supply lag earlier in the year), their prices have increased which in turn has increased the gas prices exponentially. This increase is apparent in the upward trend of both variables as mentioned above.

II. RESULTS

Preliminary Work

```
# load necessary libraries
library(readxl)
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(tsibble)
```

```
##
## Attaching package: 'tsibble'
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, union
```

```
library(forecast)
library(strucchange)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following object is masked from 'package:tsibble':
##
##     index
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: sandwich
```

```
library(vctrs)
library(vars)
```

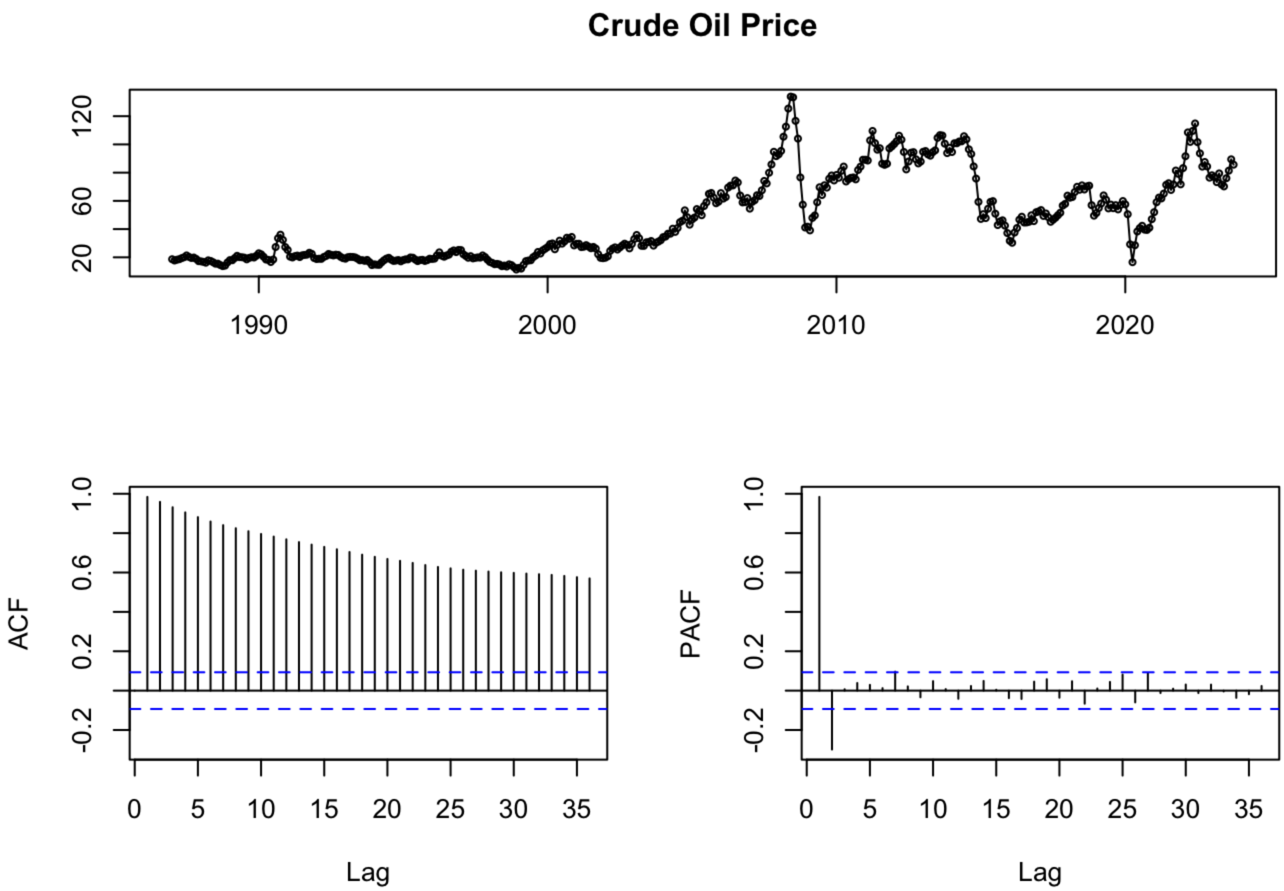
```
## Loading required package: MASS
```

```
## Loading required package: urca
```

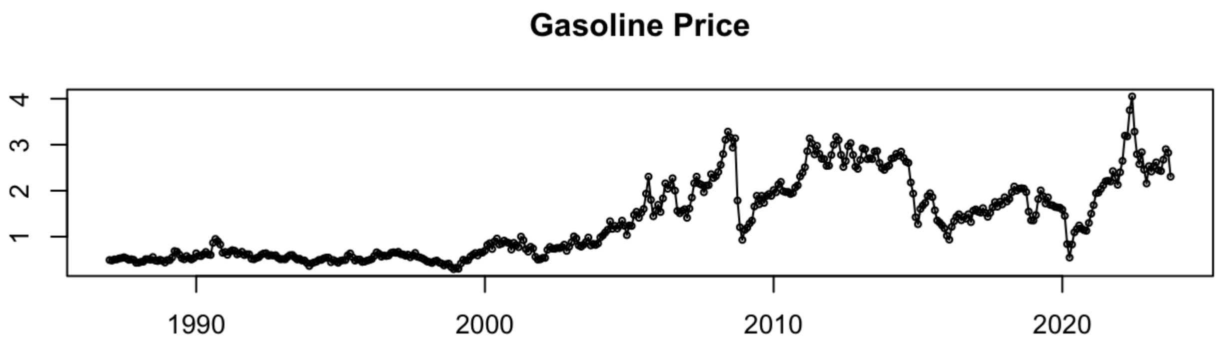
```
## Loading required package: lmtest
```

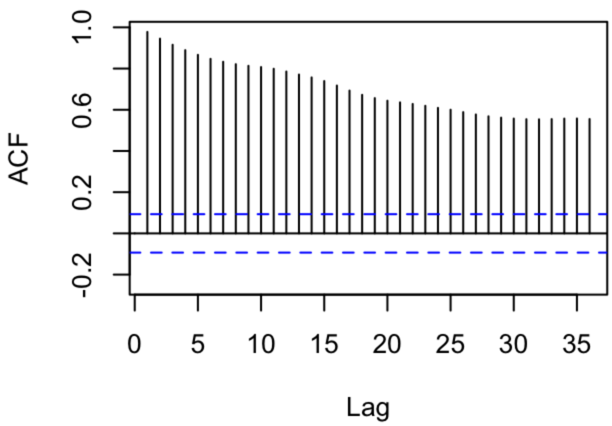
(a)

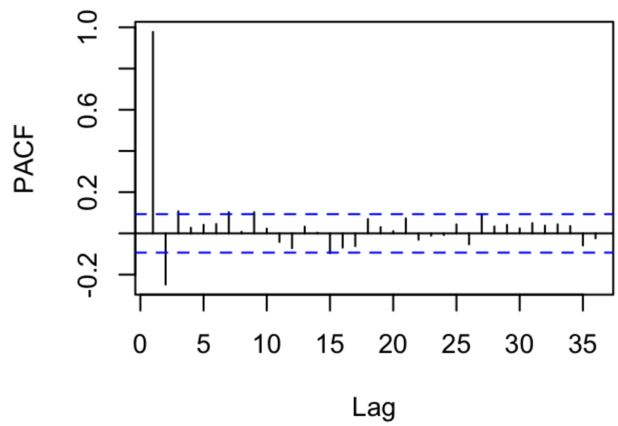
```
# load the data
crude.oil <- read_xls("Crude Oil and Gasline.xls", sheet = 2)
gasoline <- read_xls("Crude Oil and Gasline.xls", sheet = 3)

# convert the data to time series format
oil <- ts(crude.oil$`Cushing, OK WTI Spot Price FOB (Dollars per Barrel)`, start = c(198
7, 1), frequency = 12)
gas <- ts(gasoline$`U.S. Gulf Coast Conventional Gasoline Regular Spot Price FOB (Dollar
s per Gallon)`, start = c(1987, 1), frequency = 12)

# plot ACF and PACF for crude oil
tsdisplay(oil, main = "Crude Oil Price")
```




```
# plot ACF and PACF for Gasoline
tsdisplay(gas, main = "Gasoline Price")
```

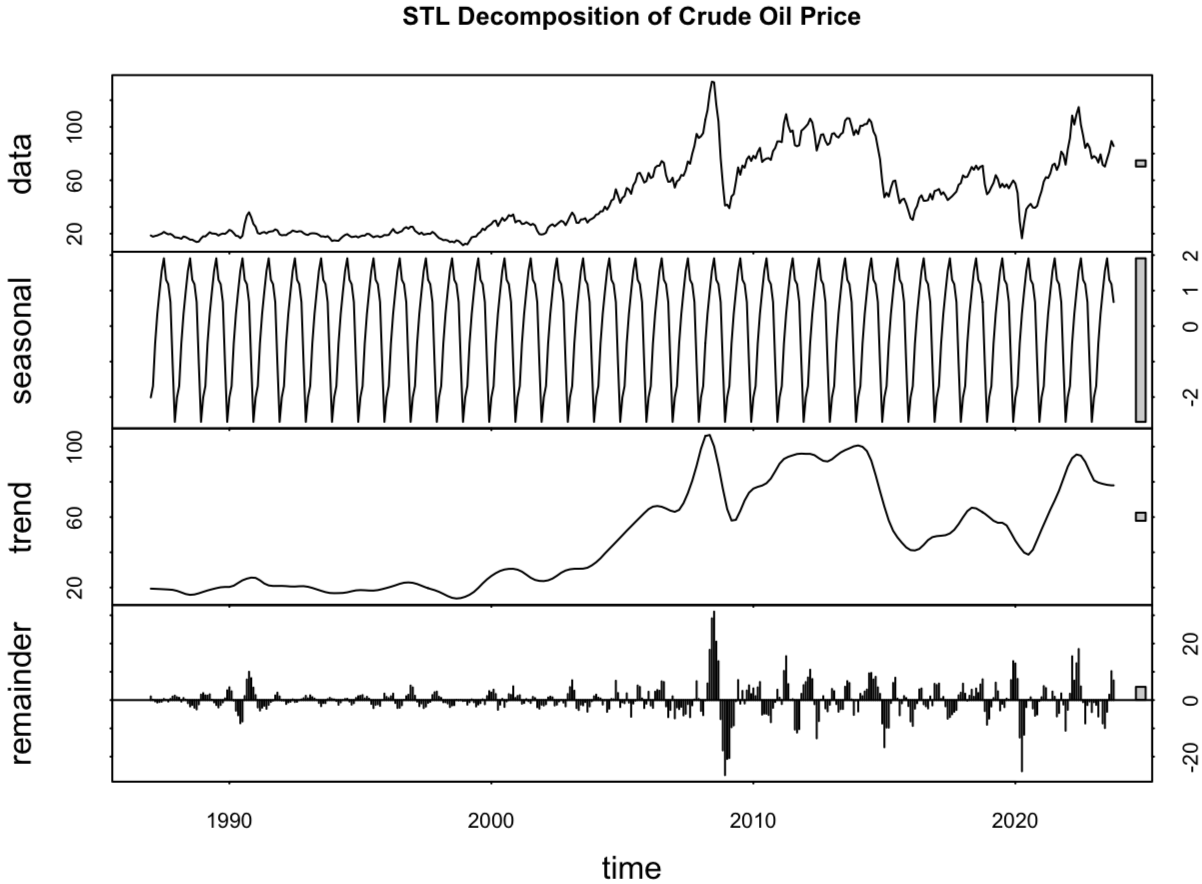




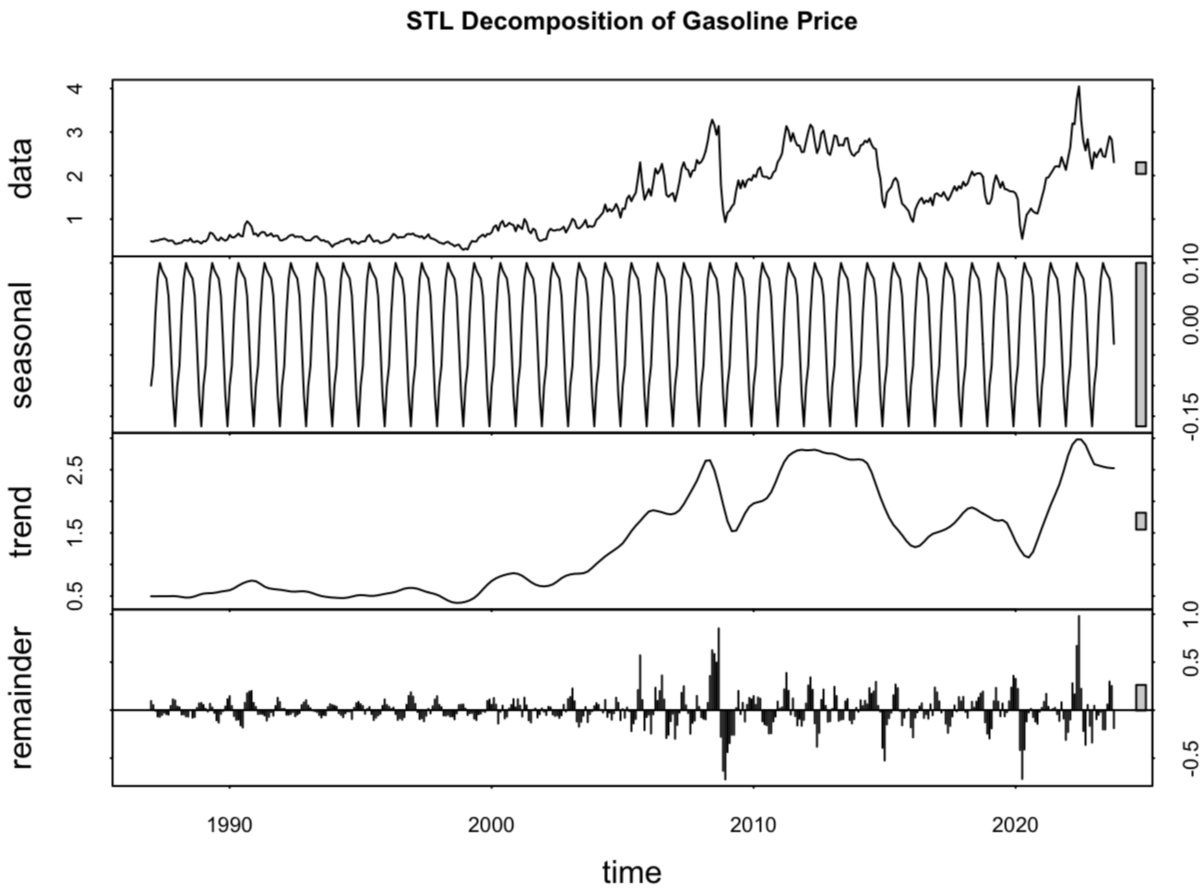


(b)

```
# plot the stl decomposition
stl_oil <- stl(oil, s.window = "periodic")
stl_gas <- stl(gas, s.window = "periodic")

plot(stl_oil, main = "STL Decomposition of Crude Oil Price")
```




```
plot(stl_gas, main = "STL Decomposition of Gasoline Price")
```



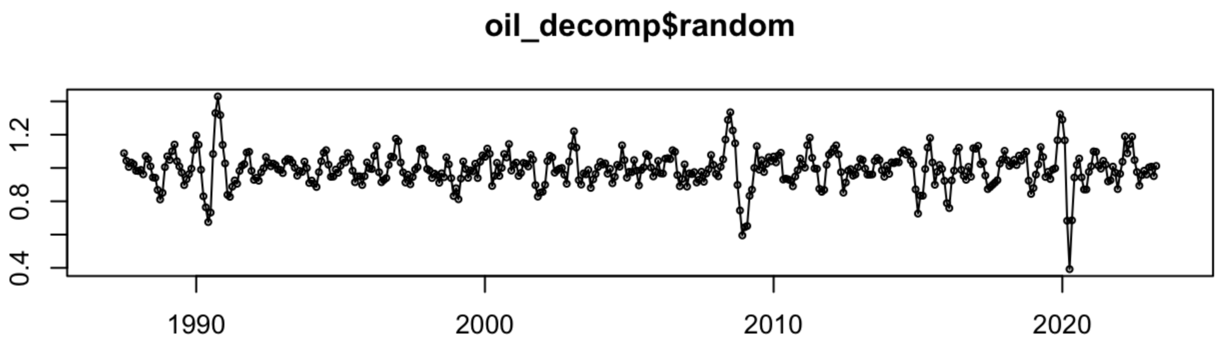
For the STL decomposition plots of crude oil price and gasoline, we can observe seasonality, an increasing trend, and cycles for both data, so we should include all the three components in our models. The data plots show that the variance of the data varies a lot, so we should try a multiplicative method for seasonality.

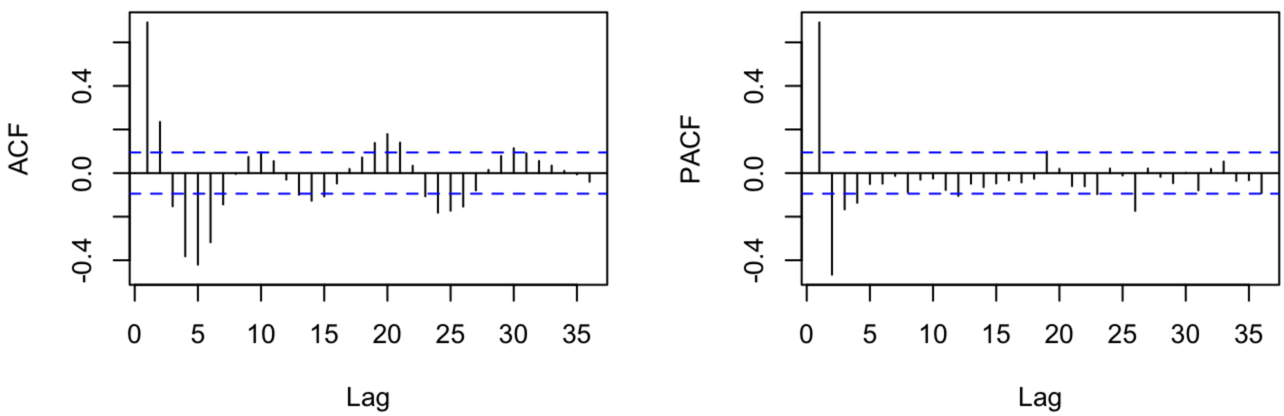
(c)

```
# decompose the data
oil_decomp <- decompose(oil, type = "multiplicative")
gas_decomp <- decompose(gas, type = "multiplicative")

# For crude oil price
# Fit the model for seasonality
oil_t1 <- tslm(oil ~ season)
summary(oil_t1)
```

```
##
## Call:
## tslm(formula = oil ~ season)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.751 -26.993  -8.022  21.677  84.628
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 45.02297    4.90271   9.183   <2e-16 ***
## season2      0.47784    6.93348   0.069    0.945
## season3      1.84919    6.93348   0.267    0.790
## season4      2.80324    6.93348   0.404    0.686
## season5      3.54784    6.93348   0.512    0.609
## season6      4.22892    6.93348   0.610    0.542
## season7      4.68595    6.93348   0.676    0.500
## season8      4.19757    6.93348   0.605    0.545
## season9      4.19162    6.93348   0.605    0.546
## season10     3.82486    6.93348   0.552    0.581
## season11     1.38786    6.98147   0.199    0.843
## season12    -0.07881    6.98147  -0.011    0.991
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29.82 on 430 degrees of freedom
## Multiple R-squared:  0.003353,   Adjusted R-squared:  -0.02214
## F-statistic: 0.1315 on 11 and 430 DF,  p-value: 0.9997
```

```
# There is no seasonal dummy variable that is significant, so we do not represent season
ality in the model.

# Fit the model for cycles and trends
# plot the acf and pacf of the randoms
tsdisplay(oil_decomp$random)
```





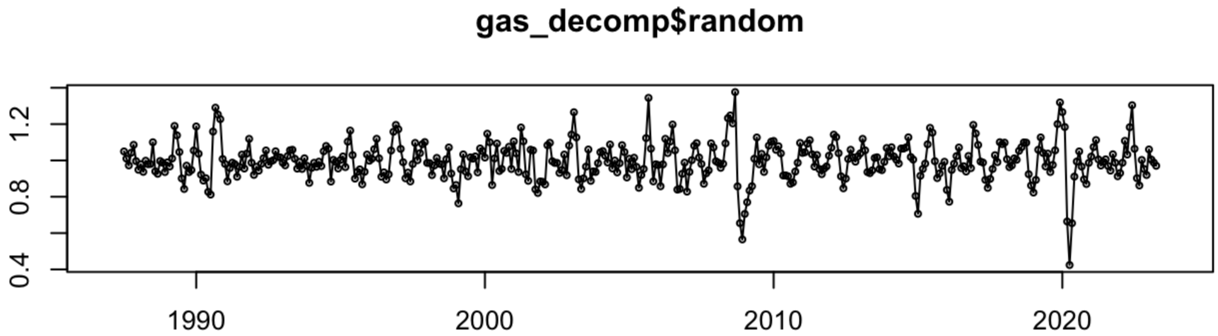

```
# Observing the ACF and PACF plots of oil price, we should use AR(4) + SAR(1)
oil_t2 <- arima(oil, order = c(4, 1, 0), seasonal = list(order = c(1, 0, 0)))
summary(oil_t2)
```

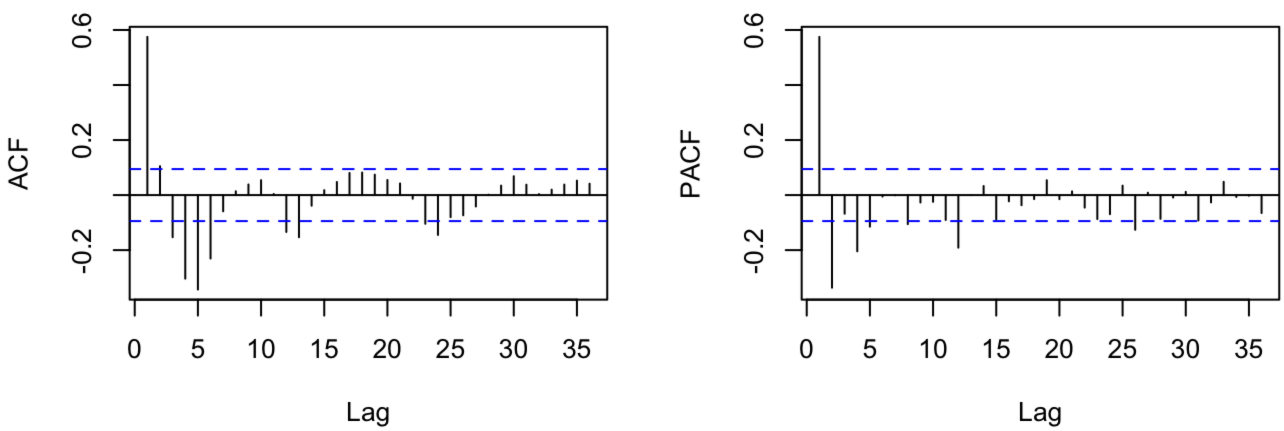
```
##
## Call:
## arima(x = oil, order = c(4, 1, 0), seasonal = list(order = c(1, 0, 0)))
##
## Coefficients:
##          ar1     ar2      ar3      ar4    sar1
##       0.3374  0.0180  -0.0560  -0.0311  0.0044
## s.e.  0.0476  0.0506   0.0503   0.0477  0.0484
##
## sigma^2 estimated as 19.94:  log likelihood = -1285.76,  aic = 2583.51
##
## Training set error measures:
##                     ME     RMSE      MAE         MPE     MAPE      MASE
## Training set 0.1054472 4.460751 3.065757 0.005111557 6.566181 0.9741118
##                      ACF1
## Training set -0.001161947
```

```
# For gasoline price
# Fit the model for seasonality
gas_t1 <- tslm(gas ~ season)
summary(gas_t1)
```

```
##
## Call:
## tslm(formula = gas ~ season)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.0689 -0.7714 -0.2017  0.6060  2.5987
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.239838   0.140117   8.849   <2e-16 ***
## season2      0.037811   0.198156   0.191    0.849
## season3      0.127919   0.198156   0.646    0.519
## season4      0.189297   0.198156   0.955    0.340
## season5      0.218757   0.198156   1.104    0.270
## season6      0.210459   0.198156   1.062    0.289
## season7      0.206622   0.198156   1.043    0.298
## season8      0.204054   0.198156   1.030    0.304
## season9      0.179973   0.198156   0.908    0.364
## season10     0.106757   0.198156   0.539    0.590
## season11    -0.002838   0.199527  -0.014    0.989
## season12    -0.048588   0.199527  -0.244    0.808
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8523 on 430 degrees of freedom
## Multiple R-squared:  0.01224,    Adjusted R-squared:  -0.01303
## F-statistic: 0.4845 on 11 and 430 DF,  p-value: 0.9129
```

```
# There is no seasonal dummy variable that is significant, so we do not represent season
ality in the model.

# Fit the model for cycles and trend
# plot the acf and pacf of the randoms
tsdisplay(gas_decomp$random)
```





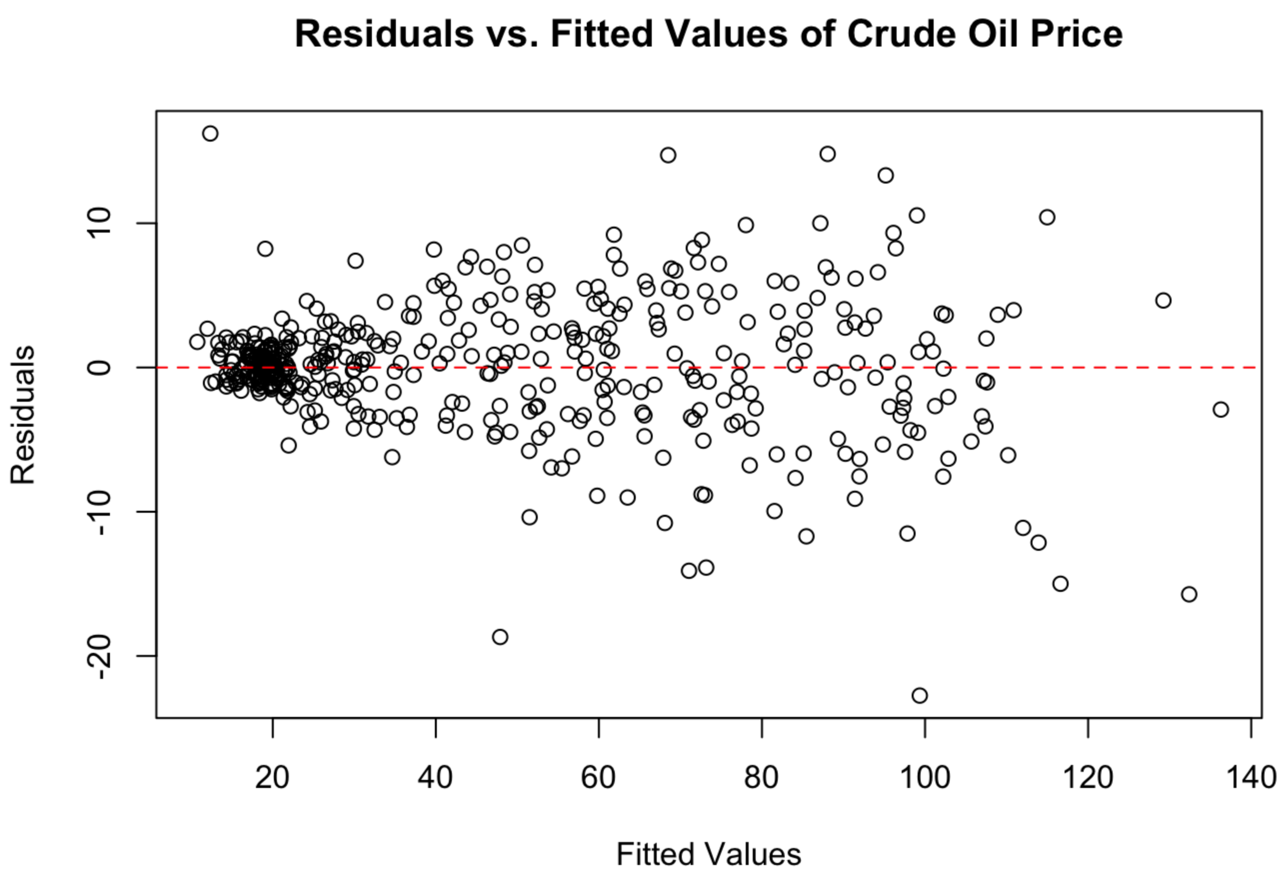

```
# Observing the ACF and PACF plots of oil prices, we should use an AR(2) + SAR(1) model
with monthly frequency (i.e., a frequency of 12).
gas_t2 <- arima(gas, order = c(2, 1, 0), seasonal = list(order = c(1, 0, 0)))
summary(gas_t2)
```

```
##
## Call:
## arima(x = gas, order = c(2, 1, 0), seasonal = list(order = c(1, 0, 0)))
##
## Coefficients:
##          ar1      ar2     sar1
##       0.2602  -0.1339  -0.0011
## s.e.  0.0478   0.0480   0.0494
##
## sigma^2 estimated as 0.02611:  log likelihood = 178.04,  aic = -348.08
##
## Training set error measures:
##                       ME      RMSE       MAE        MPE    MAPE      MASE
## Training set 0.003465212 0.1613962 0.1040422 -0.1977409 7.99161 0.9559476
##                     ACF1
## Training set -0.00444569
```

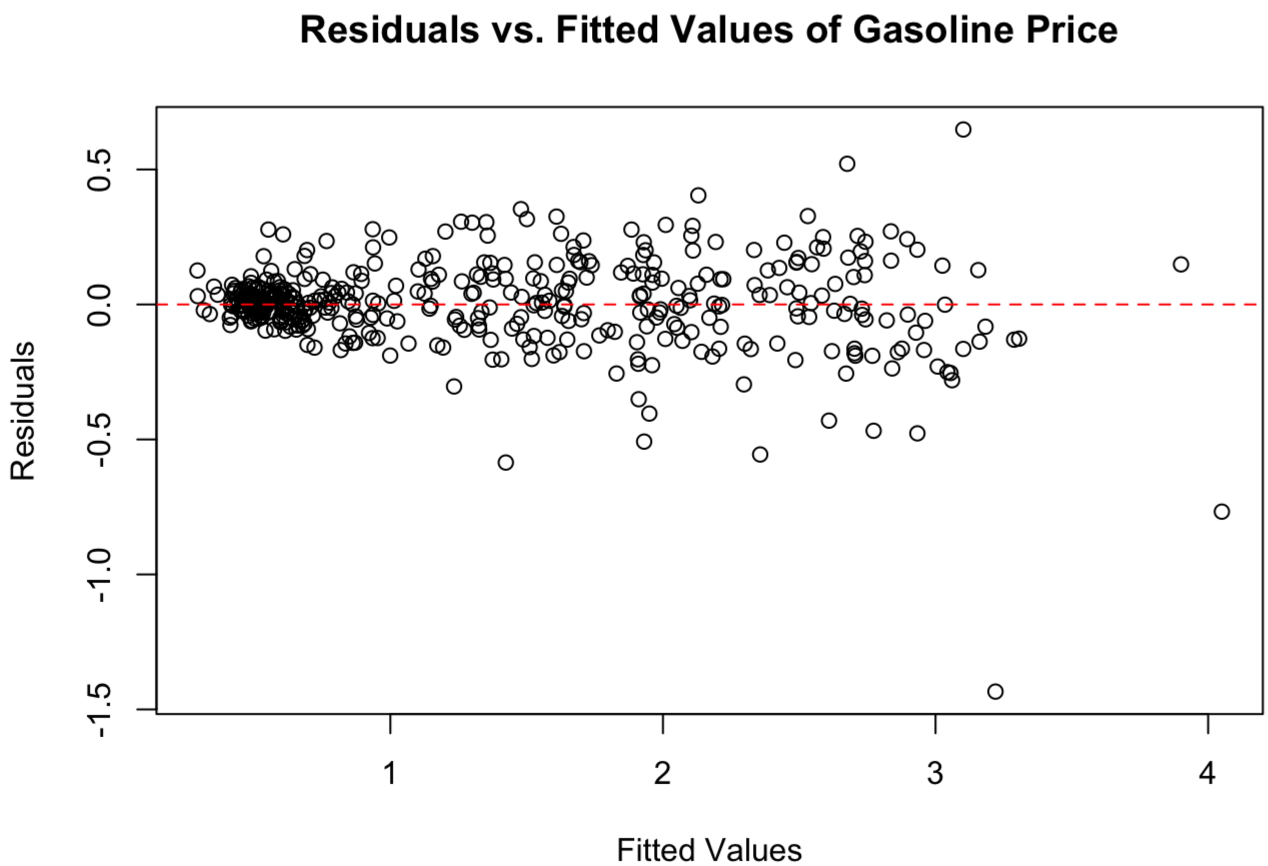
For the crude oil prices, none of the seasonal dummy variables are significant, so we do not include them in the model. Observing the ACF and PACF plots of the random terms after decomposition, we use an AR(4) + Seasonal AR(1) model for the cycles and seasonality. In addition, we include the trend in the ARIMA through an integration of I(1). For the gasoline price, none of the seasonal dummy variables are significant, so we do not include them in the model. Observing the ACF and PACF plots of the random terms after decomposition, we use an AR(2) + Seasonal AR(1) model for the cycles and seasonality. In addition, we include the trend in the ARIMA through an integration of I(1).

(e)

```
# find the residuals
oil_resid <- resid(oil_t2)
gas_resid <- resid(gas_t2)

# Plotting Residuals vs. Fitted Values for Crude Oil Price
plot(fitted(oil_t2), oil_resid, main = "Residuals vs. Fitted Values of Crude Oil Price",
xlab = "Fitted Values", ylab = "Residuals")
abline(h = 0, col = "red", lty = 2)
```



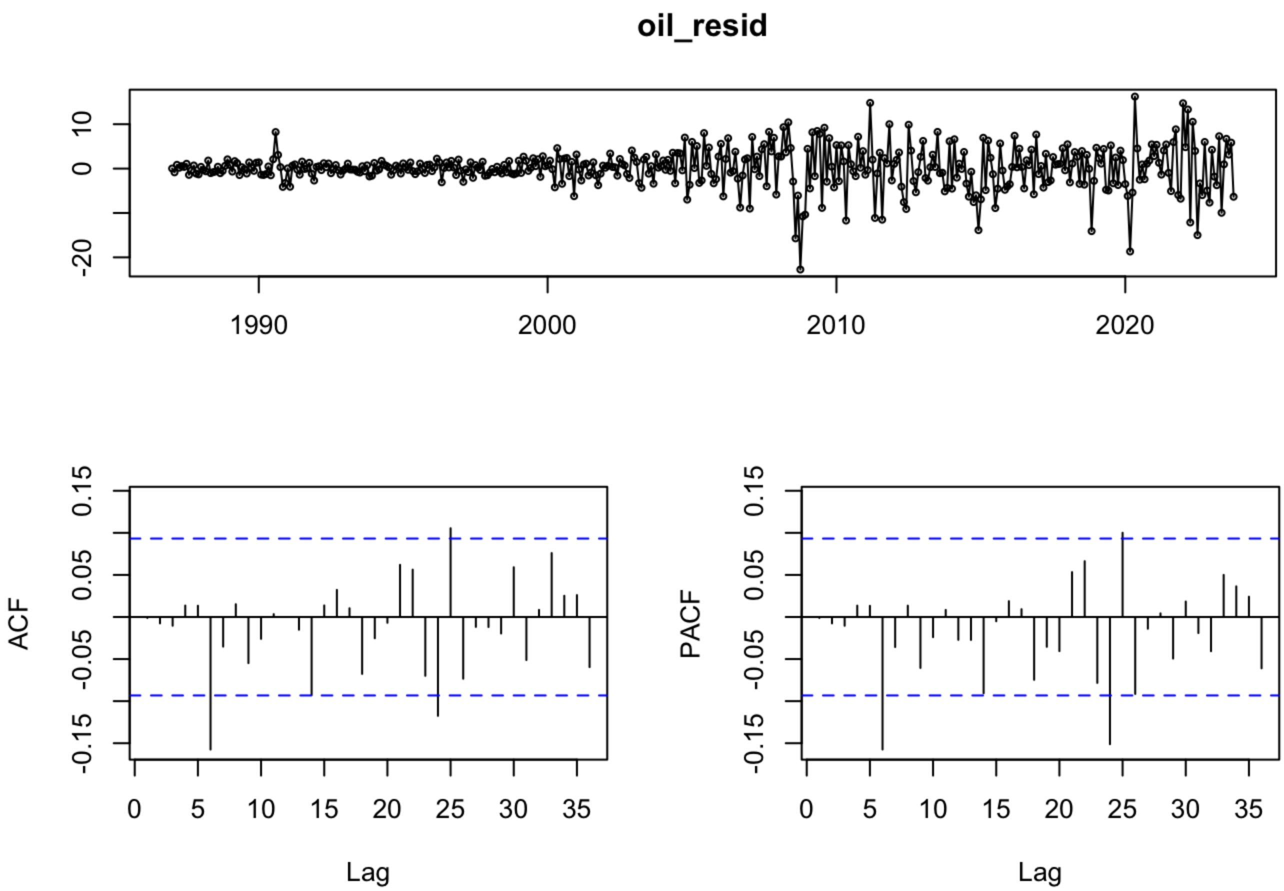

```
# Plotting Residuals vs. Fitted Values for Gasoline Price
plot(fitted(gas_t2), gas_resid, main = "Residuals vs. Fitted Values of Gasoline Price",
xlab = "Fitted Values", ylab = "Residuals")
abline(h = 0, col = "red", lty = 2)
```



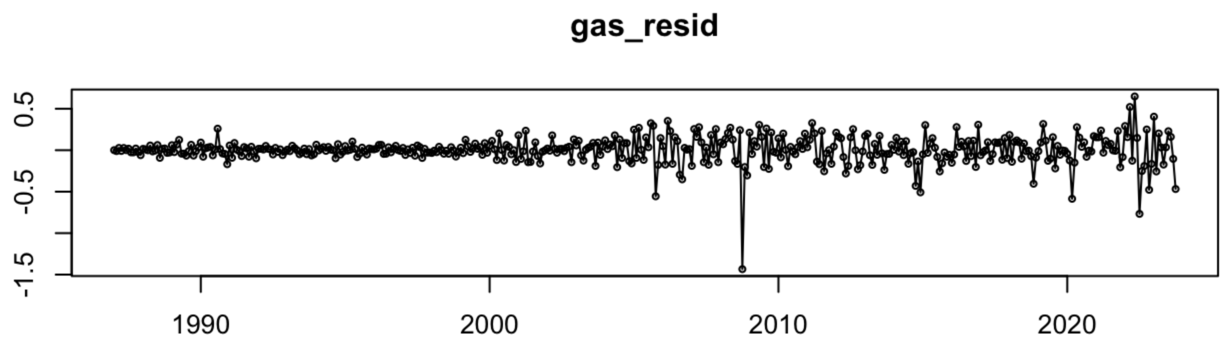
Observing the plot of residuals vs. fitted values, we cannot observe any obvious patterns, and the residuals seem randomly scattered around zero. This suggests that the the residuals may just be white noise. However, the variance is not constant, so it is difficult to say so with certainty.

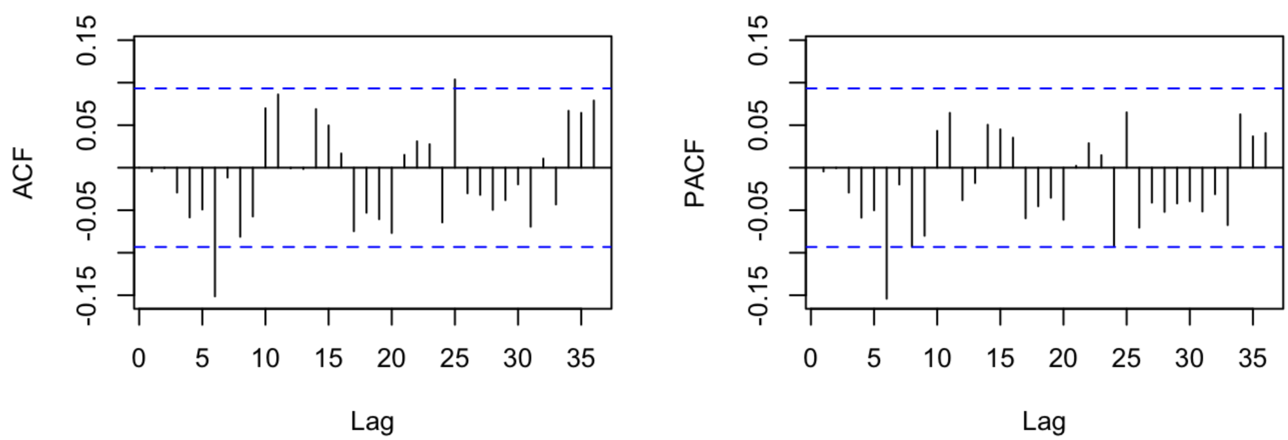
(f)

```
# ACF and PACF plots for residuals of oil price
tsdisplay(oil_resid)
```




```
# ACF and PACF plots for residuals of gasoline price
tsdisplay(gas_resid)
```

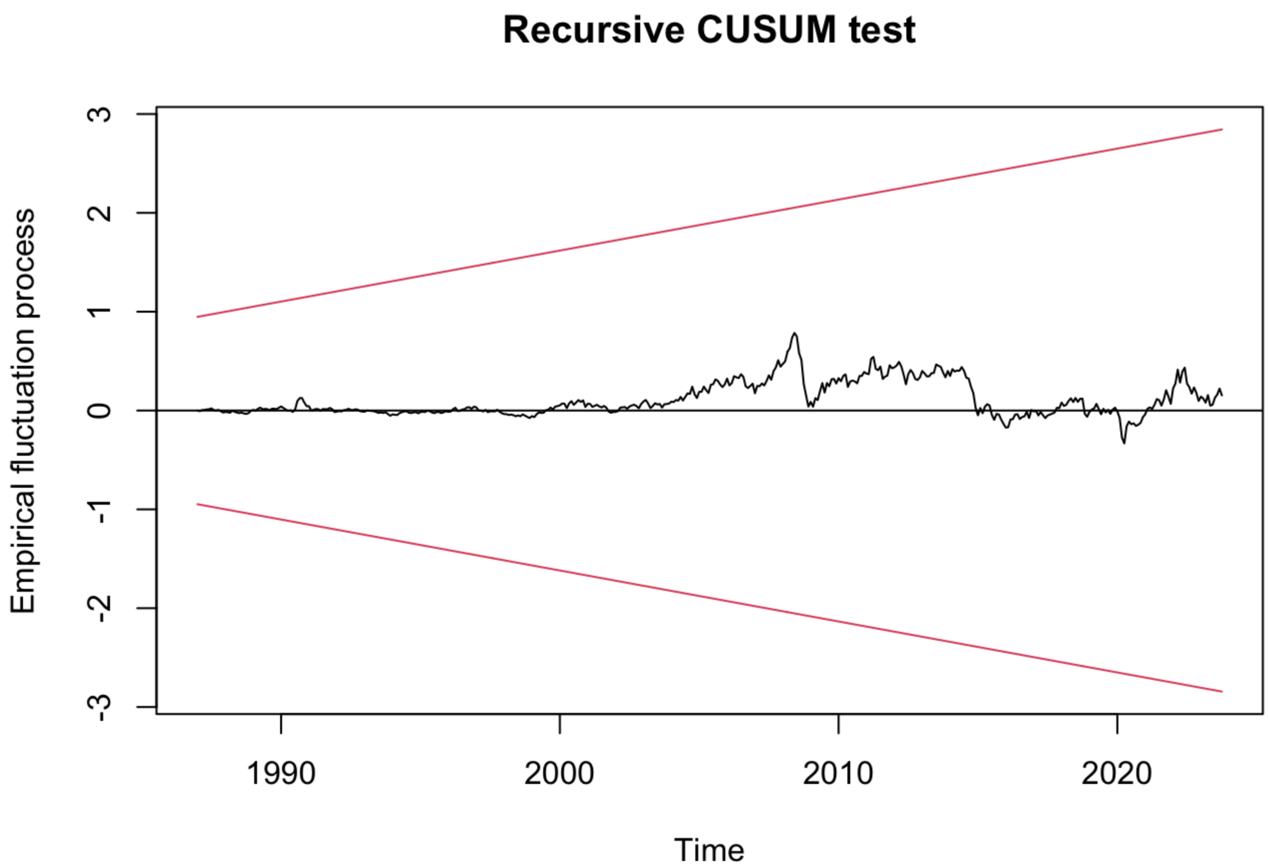




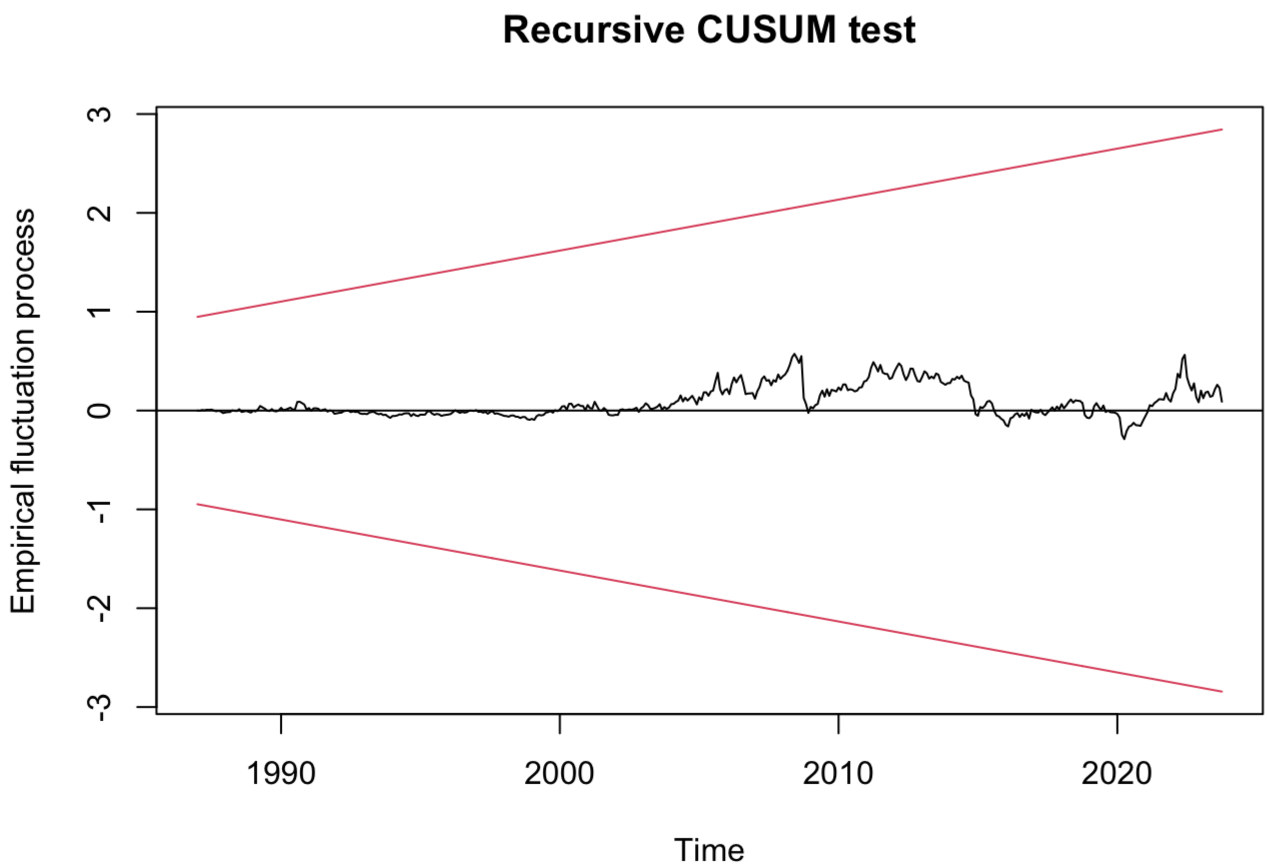
The ACF and PACF plots for the crude oil price suggest that the residuals are white noise, meaning that our model did a good job. However, in the ACF and PACF plots for the gasoline prices, even though there are no high spikes, we can observe some up and down patterns from the plots, meaning that we still need improvements for the gas price model.

(g)

```
# CUSUM plot for crude oil price
plot(efp(oil_t2$res~1, type = "Rec-CUSUM"))
```



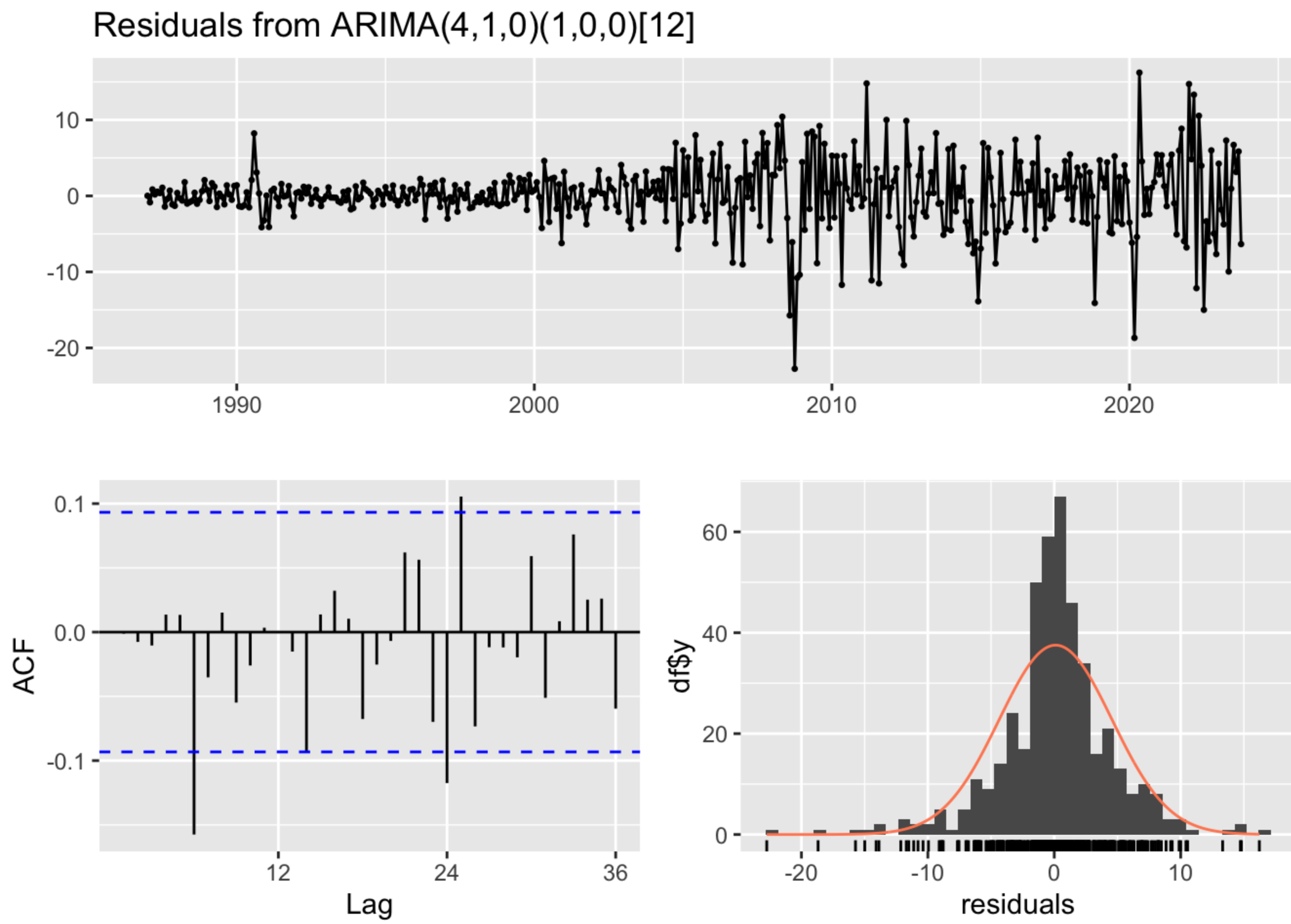

```
# CUSUM plot for gasoline price
plot(efp(gas_t2$res~1, type = "Rec-CUSUM"))
```



According to plots of the CUSUM test for both crude oil and gasoline, the cumulative sum of both series fluctuates around zero and does not cross the bans. Therefore, we can conclude that the models we built for each series don’t break over time, and they both catch the dynamics in the crude oil series and gasoline series fairly well.

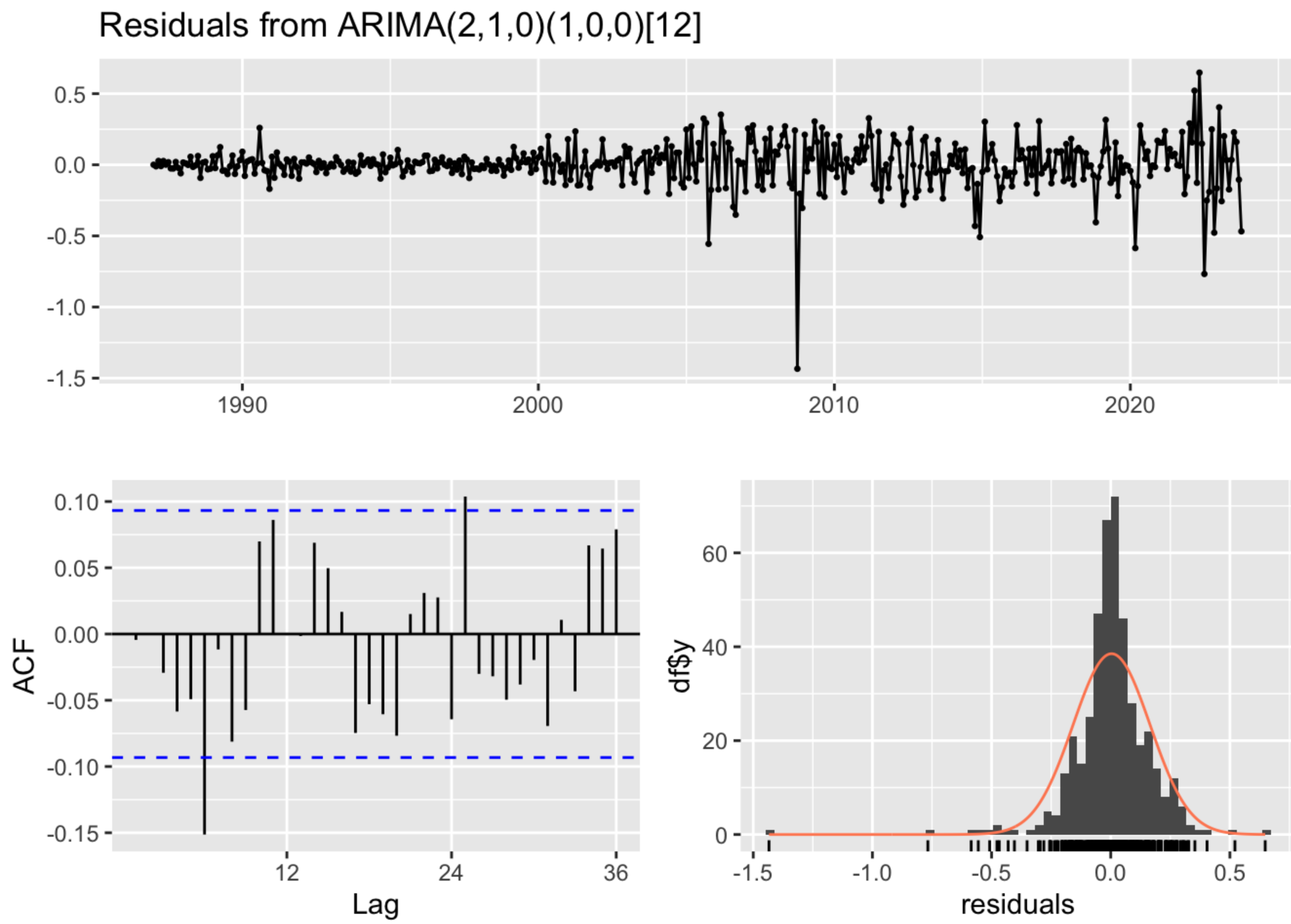
(h)

```
# diagnostic plots for oil price
oil_diag <- checkresiduals(oil_t2)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(4,1,0)(1,0,0)[12]
## Q* = 32.9, df = 19, p-value = 0.02469
##
## Model df: 5.   Total lags used: 24
```

```
# diagnostic plots for gas price
gas_diag <- checkresiduals(gas_t2)
```

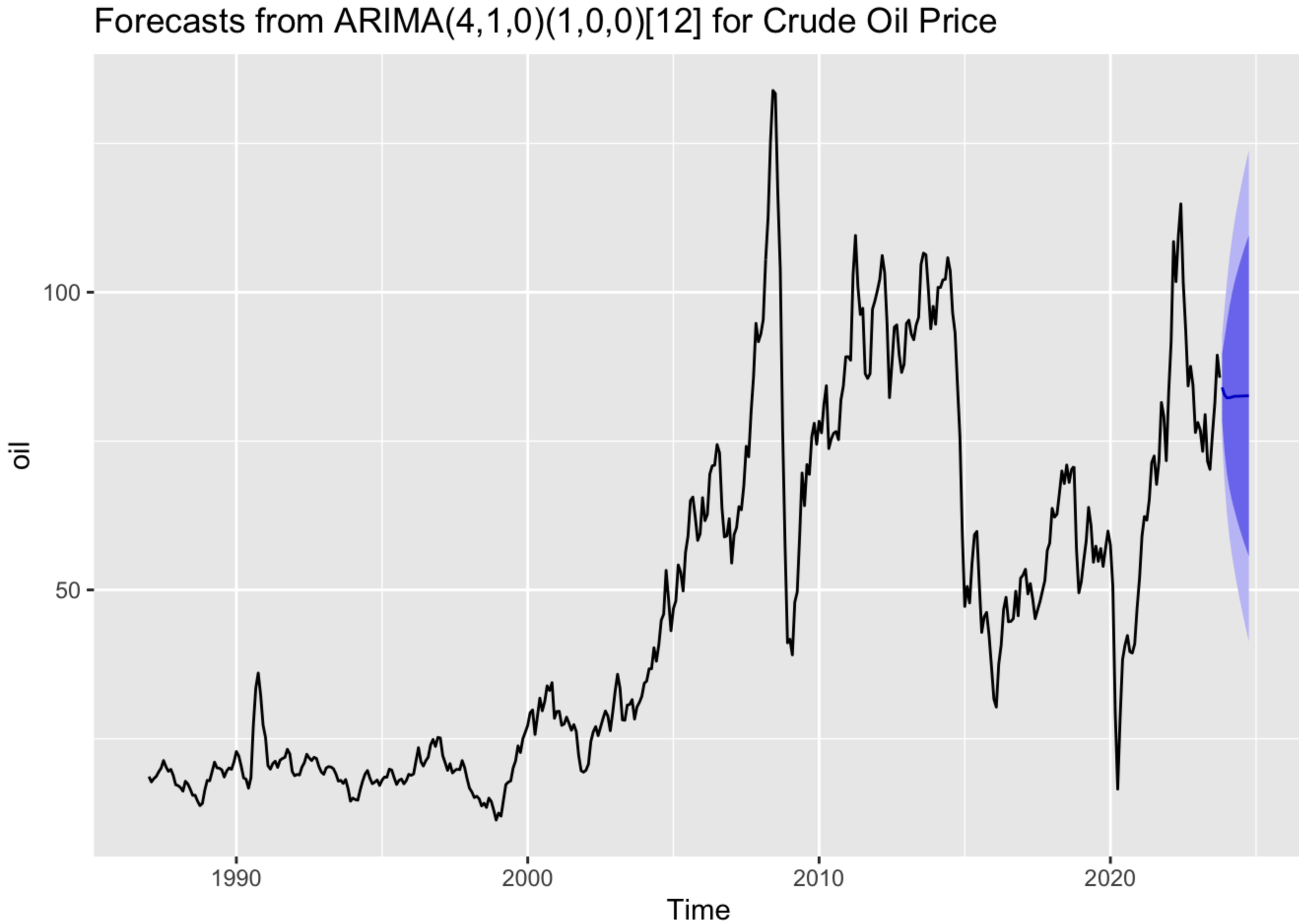



```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(2,1,0)(1,0,0)[12]
## Q* = 38.049, df = 21, p-value = 0.01272
##
## Model df: 3.   Total lags used: 24
```

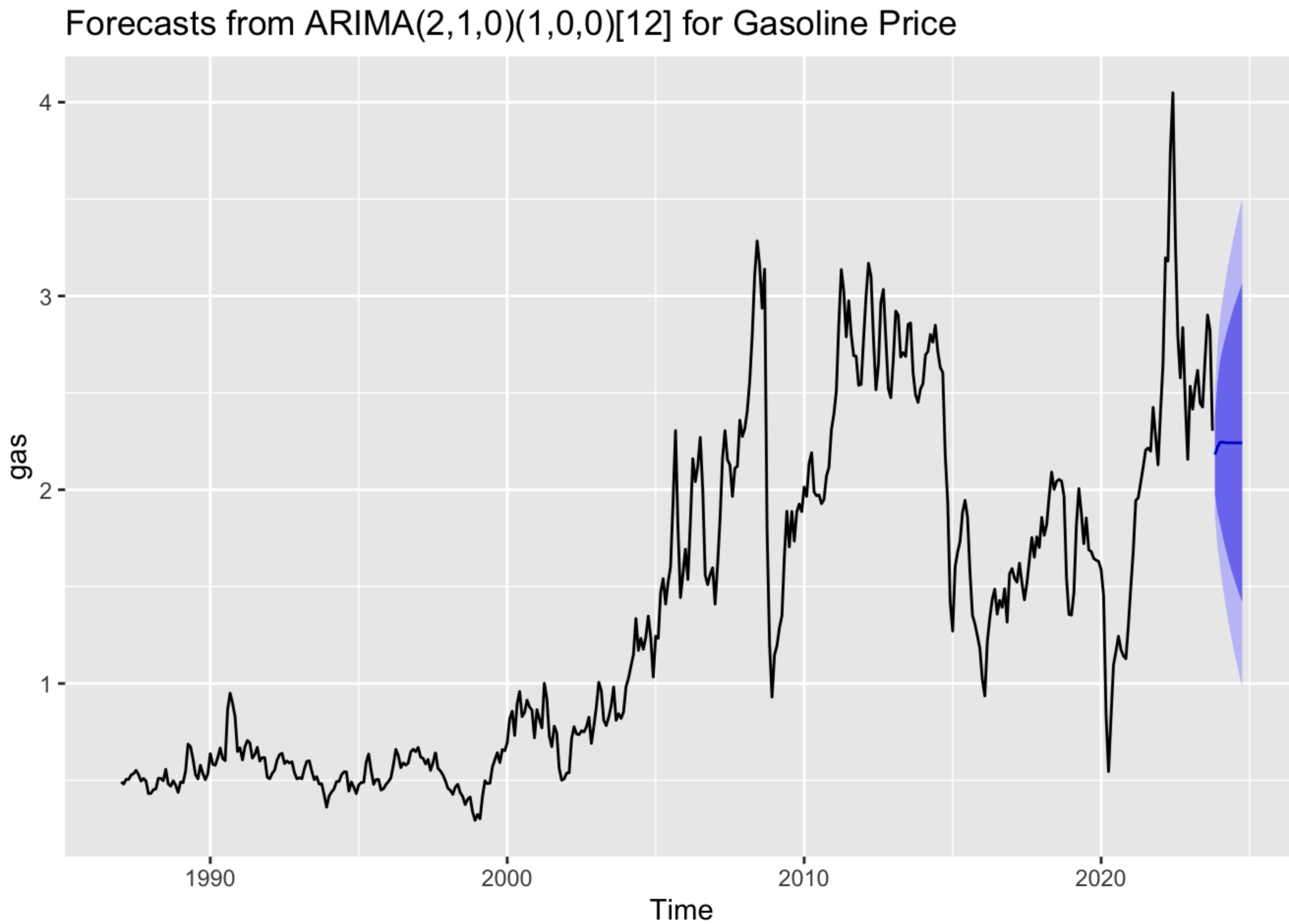
For both plots, we can observe that except for a handful of points, most of the residuals are randomly scattered around 0, and the ACF plots look similar to white noise. The distribution of y is approximately normal. However, the Ljung-Box test gives us p-values that are smaller than 0.05, meaning that there still exists significant autocorrelation in the residuals.

(i)

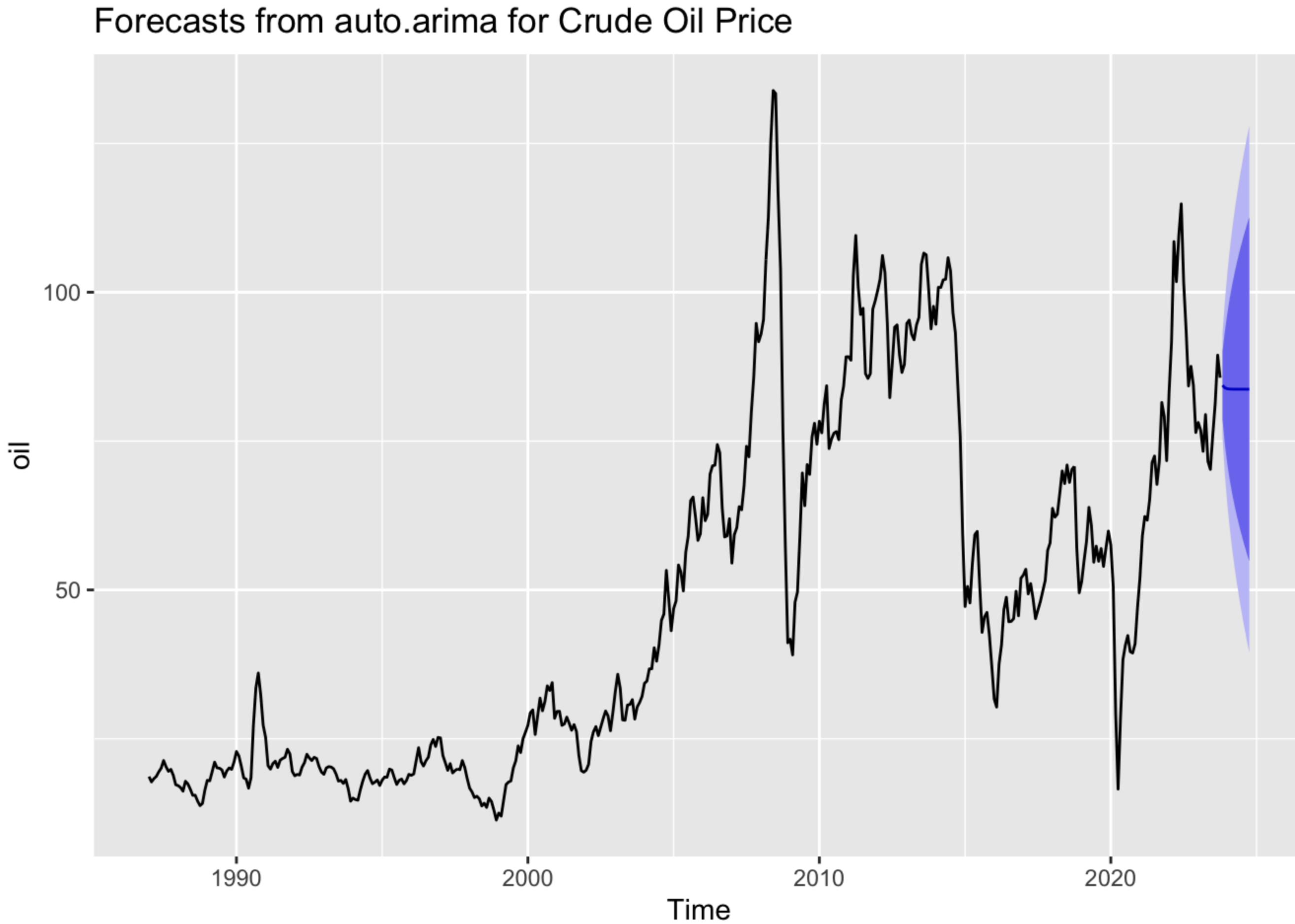
```
# 12-step forecast based on our model for crude oil price
oil_forc <- forecast(oil_t2, h = 12)
autoplot(oil_forc, main = "Forecasts from ARIMA(4,1,0)(1,0,0)[12] for Crude Oil Price")
```



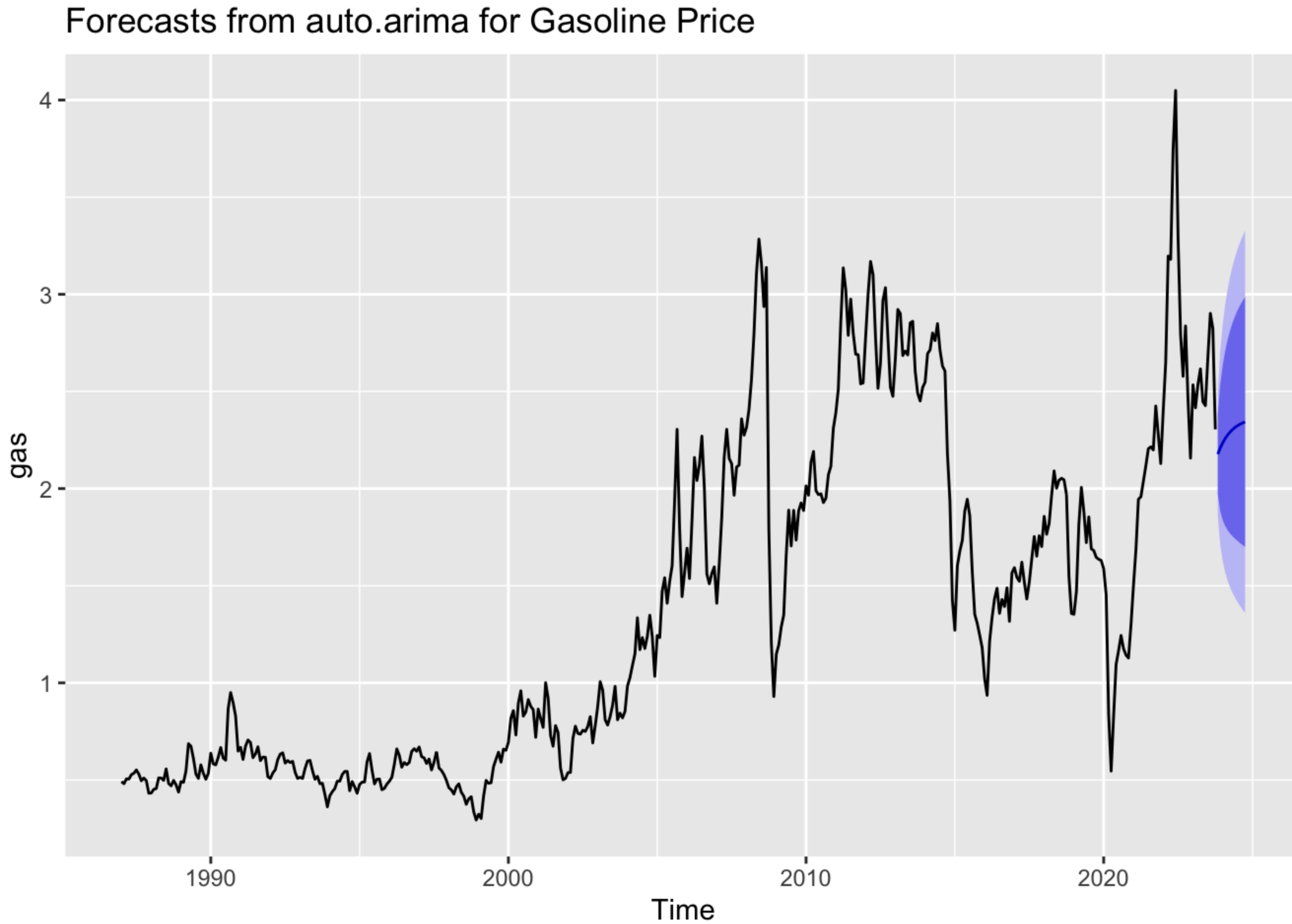

```
# 12-step forecast based on our model for gasoline price
gas_forc <- forecast(gas_t2, h = 12)
autoplot(gas_forc, main = "Forecasts from ARIMA(2,1,0)(1,0,0)[12] for Gasoline Price")
```




```
# build the auto.arima models
oil_auto.arima <- auto.arima(oil)
gas_auto.arima <- auto.arima(gas)

# 12-step forecast based on auto.arima for crude oil price
oil_auto_forc <- forecast(oil_auto.arima, h = 12)
autoplot(oil_auto_forc, main = "Forecasts from auto.arima for Crude Oil Price")
```




```
# 12-step forecast based on auto.arima for crude oil price
gas_auto_forc <- forecast(gas_auto.arima, h = 12)
autoplot(gas_auto_forc, main = "Forecasts from auto.arima for Gasoline Price")
```



(j)

```
# compute the MAPE of oil price models
oil_mape <-data.frame(accuracy(oil_forc))$MAPE
oil_mape
```

```
## [1] 6.566181
```

```
oil_auto_mape <- data.frame(accuracy(oil_auto.arima))$MAPE
oil_auto_mape
```

```
## [1] 6.561658
```

```
# compute the MAPE of gas price models
gas_mape <-data.frame(accuracy(gas_forc))$MAPE
gas_mape
```

```
## [1] 7.99161
```

```
gas_auto_mape <- data.frame(accuracy(gas_auto.arima))$MAPE
gas_auto_mape
```

```
## [1] 7.969732
```

Comparing the MAPE of our models with that of the auto.arima models, we found that auto.arima models have slightly smaller MAPE for both crude oil and gasoline prices. Thus, auto.arima produces marginally more precise estimates.

(k)

```
# crude oil price
# combine the forecasts by adding them up and take the average
comb_forc <- (oil_forc$mean + forecast(oil_auto.arima, h=12)[["mean"]])/2
comb_mape <- mean(abs((oil_forc$mean - oil_auto_forc$mean) / comb_forc)) * 100
comb_mape
```

```
## [1] 1.394119
```

```
# gasoline price
# combine the forecasts by adding them up and take the average
comb_forc_gas <- (gas_forc$mean + forecast(gas_auto.arima, h=12)[["mean"]])/2
comb_mape_gas <- mean(abs((gas_forc$mean - gas_auto_forc$mean) / comb_forc_gas)) * 100
comb_mape_gas
```

```
## [1] 2.264359
```

The MAPE of the combined forecasts is smaller than the individual ones for both the crude oil and gasoline price.

(l)

```
# combine the two time series into one and convert to data frame
y <- data.frame(cbind(oil, gas))
# fit the VAR model
VARselect(y, lag.max = 5)
```

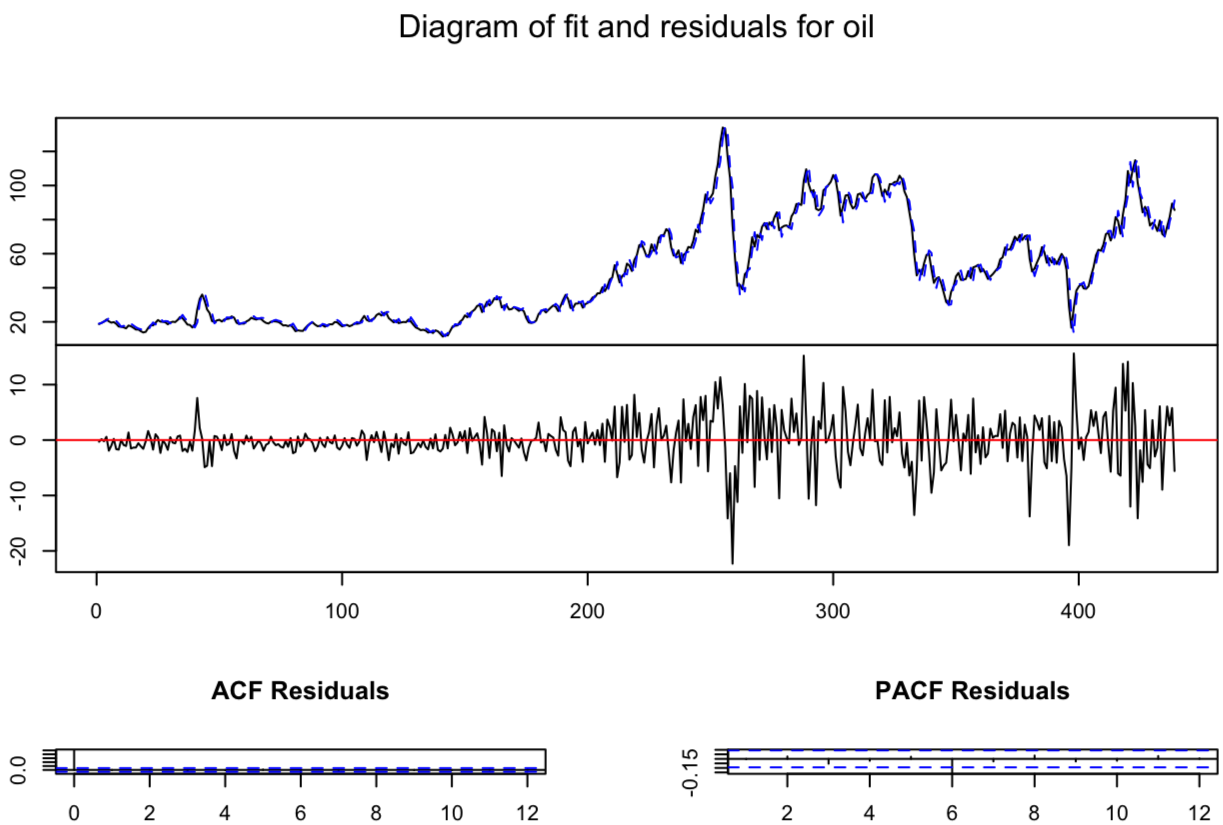
```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      3      3      2      3
##
## $criteria
##                 1          2          3          4          5
## AIC(n) -1.5052458 -1.6554345 -1.6813168 -1.6691698 -1.6709271
## HQ(n)  -1.4831407 -1.6185927 -1.6297383 -1.6028545 -1.5898751
## SC(n)  -1.4492284 -1.5620722 -1.5506096 -1.5011176 -1.4655300
## FPE(n)  0.2219628  0.1910094  0.1861297  0.1884056  0.1880766
```

```
# most of the criterias support lag = 3, so we use lag = 3
y_m1 <- VAR(y, p = 3)
summary(y_m1)
```

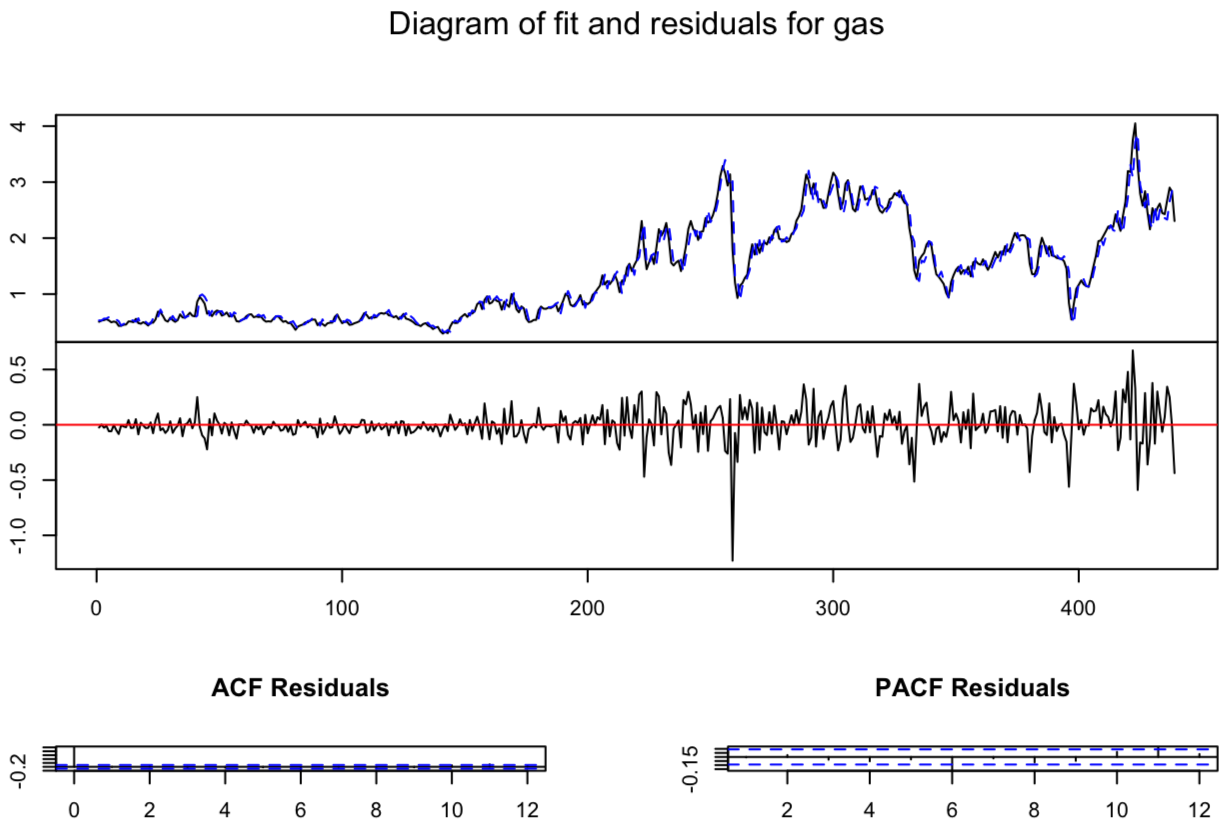
```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: oil, gas
## Deterministic variables: const
## Sample size: 439
## Log Likelihood: -860.729
## Roots of the characteristic polynomial:
## 0.9728 0.8123 0.4808 0.4808 0.3939 0.007992
## Call:
## VAR(y = y, p = 3)
##
##
## Estimation results for equation oil:
## ====================================
## oil = oil.l1 + gas.l1 + oil.l2 + gas.l2 + oil.l3 + gas.l3 + const
##
##        Estimate Std. Error t value Pr(>|t|)
## oil.l1  1.22369    0.07973  15.347   <2e-16 ***
## gas.l1  3.77070    2.24960   1.676   0.0944 .
## oil.l2 -0.14777    0.11596  -1.274   0.2032
## gas.l2 -6.80018    3.06470  -2.219   0.0270 *
## oil.l3 -0.14634    0.07944  -1.842   0.0661 .
## gas.l3  4.91502    2.29422   2.142   0.0327 *
## const   0.89106    0.40829   2.182   0.0296 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 4.461 on 432 degrees of freedom
## Multiple R-Squared: 0.9774,  Adjusted R-squared: 0.9771
## F-statistic:  3120 on 6 and 432 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation gas:
## ====================================
## gas = oil.l1 + gas.l1 + oil.l2 + gas.l2 + oil.l3 + gas.l3 + const
##
##         Estimate Std. Error t value Pr(>|t|)
## oil.l1  0.005689   0.002809   2.026 0.043426 *
## gas.l1  1.083632   0.079241  13.675  < 2e-16 ***
## oil.l2  0.007363   0.004084   1.803 0.072138 .
## gas.l2 -0.550745   0.107953  -5.102 5.05e-07 ***
## oil.l3 -0.009579   0.002798  -3.423 0.000677 ***
## gas.l3  0.325654   0.080813   4.030 6.60e-05 ***
## const   0.029985   0.014382   2.085 0.037663 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.1571 on 432 degrees of freedom
## Multiple R-Squared: 0.966,   Adjusted R-squared: 0.9656
```

```
## F-statistic:  2047 on 6 and 432 DF,  p-value: < 2.2e-16
##
##
##
## Covariance matrix of residuals:
##         oil     gas
## oil 19.8981 0.55911
## gas  0.5591 0.02469
##
## Correlation matrix of residuals:
##        oil    gas
## oil 1.0000 0.7977
## gas 0.7977 1.0000
```

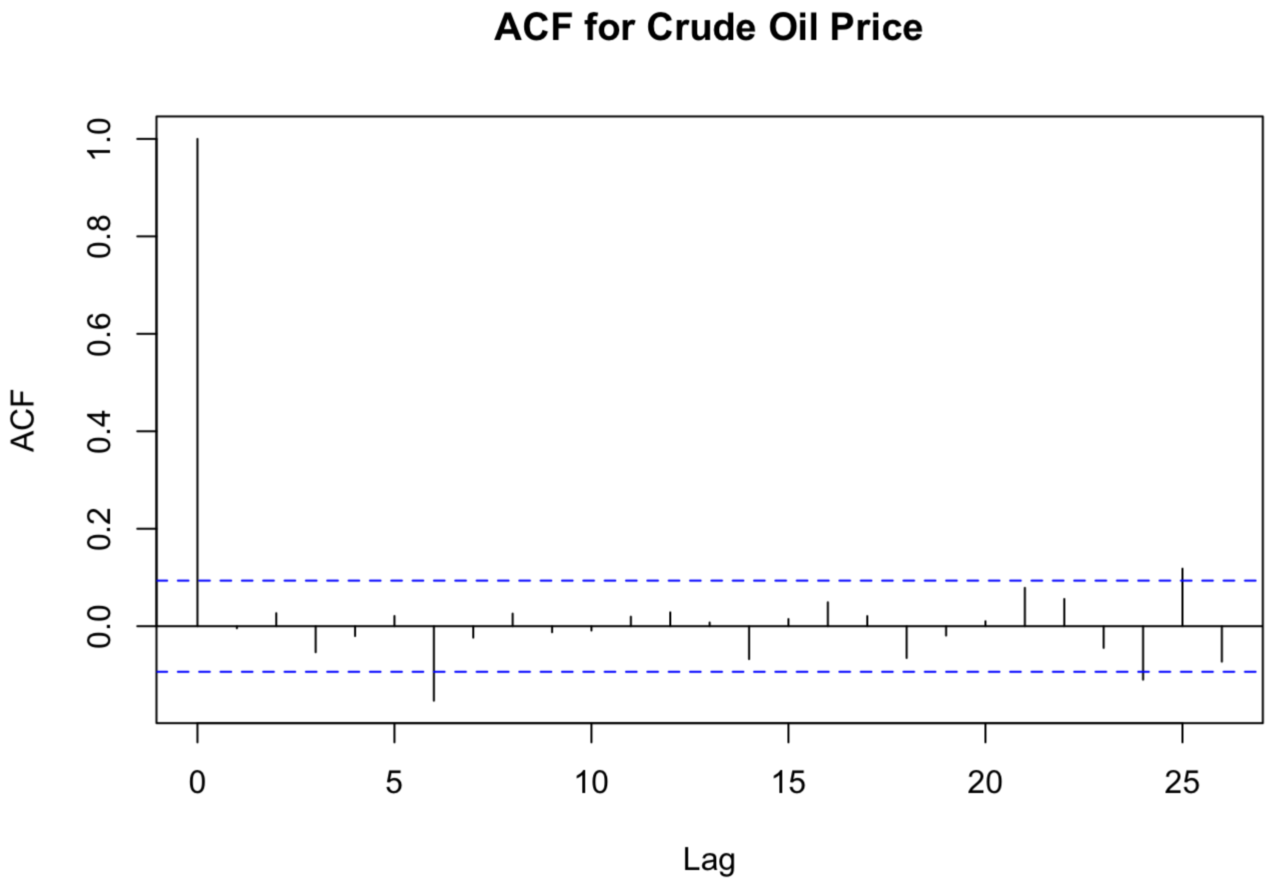
```
plot(y_m1, names = "oil")
```



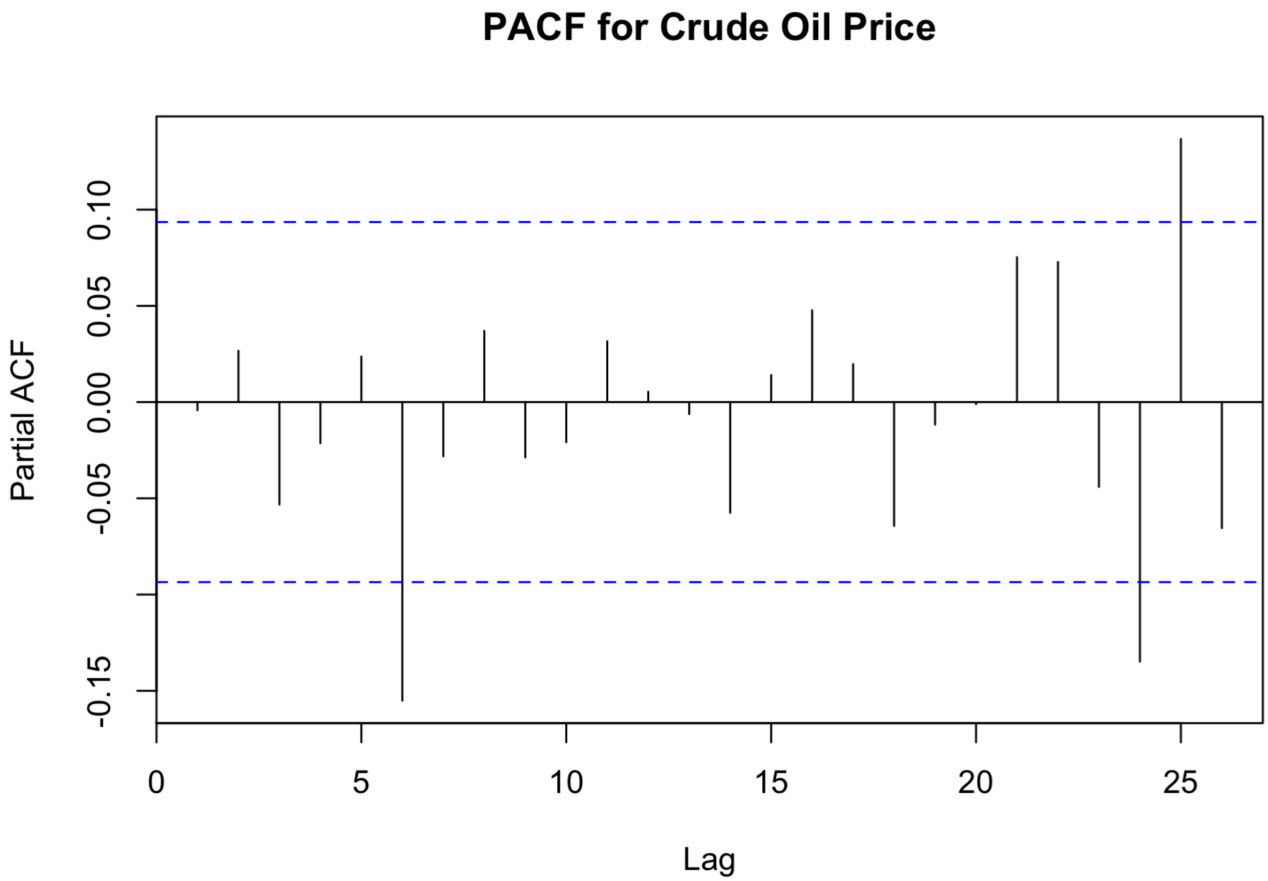

```
plot(y_m1, names = "gas")
```



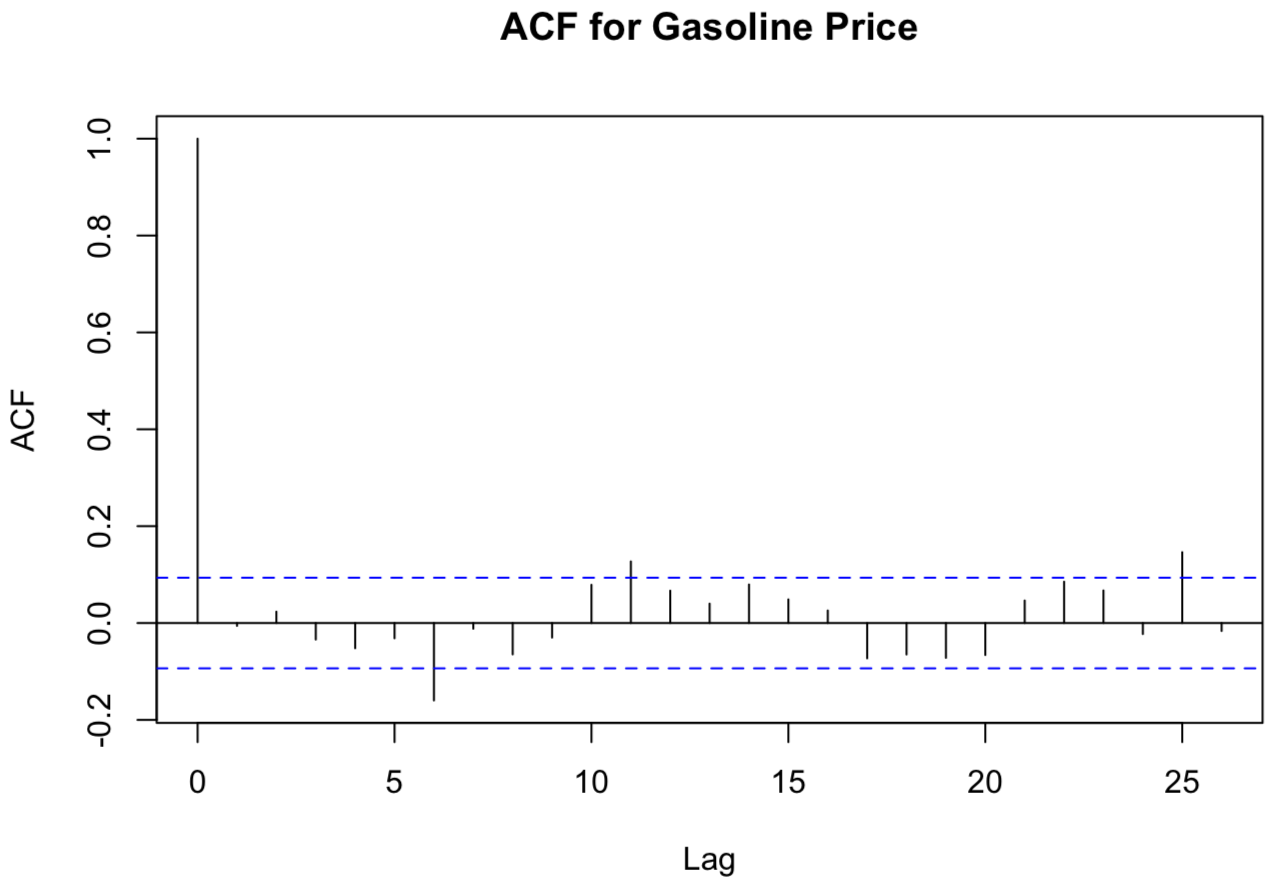

```
# ACF and PACF plots for Crude Oil Price
acf(residuals(y_m1)[,1], main = "ACF for Crude Oil Price")
```



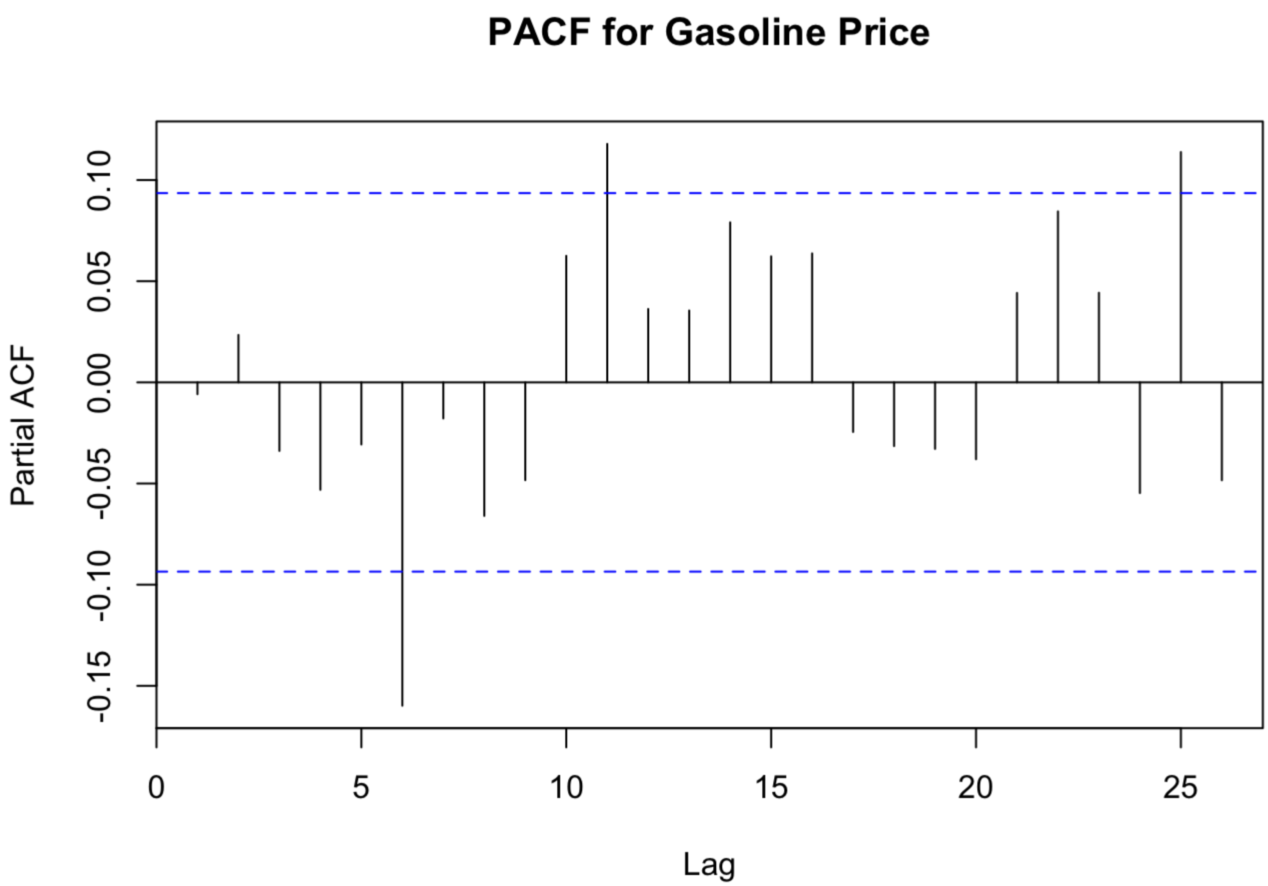

```
pacf(residuals(y_m1)[,1], main = "PACF for Crude Oil Price")
```




```
# ACF and PACF plots for Gasoline Price
acf(residuals(y_m1)[,2], main = "ACF for Gasoline Price")
```




```
pacf(residuals(y_m1)[,2], main = "PACF for Gasoline Price")
```

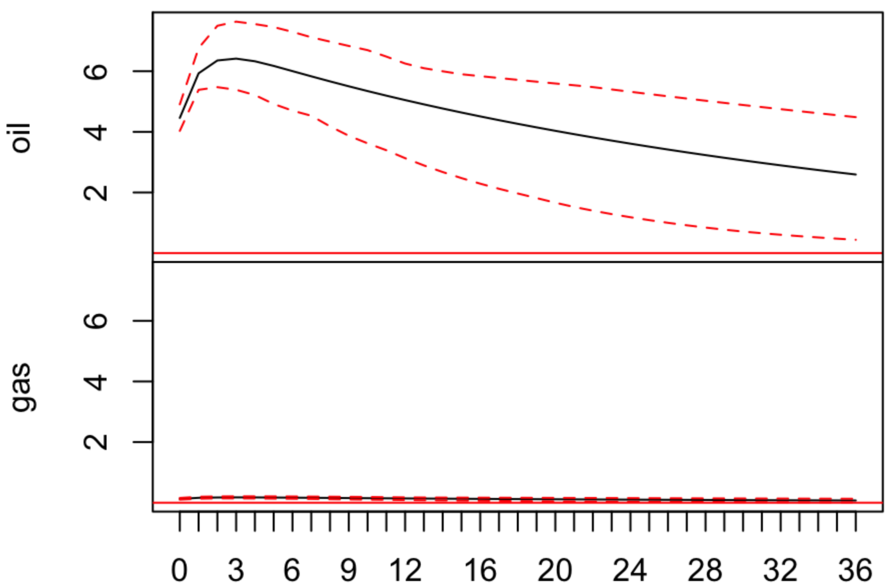


The ACF and PACF plots of the residuals of the VAR model look very similar to white noise, meaning that the VAR model did a great job.

(m)

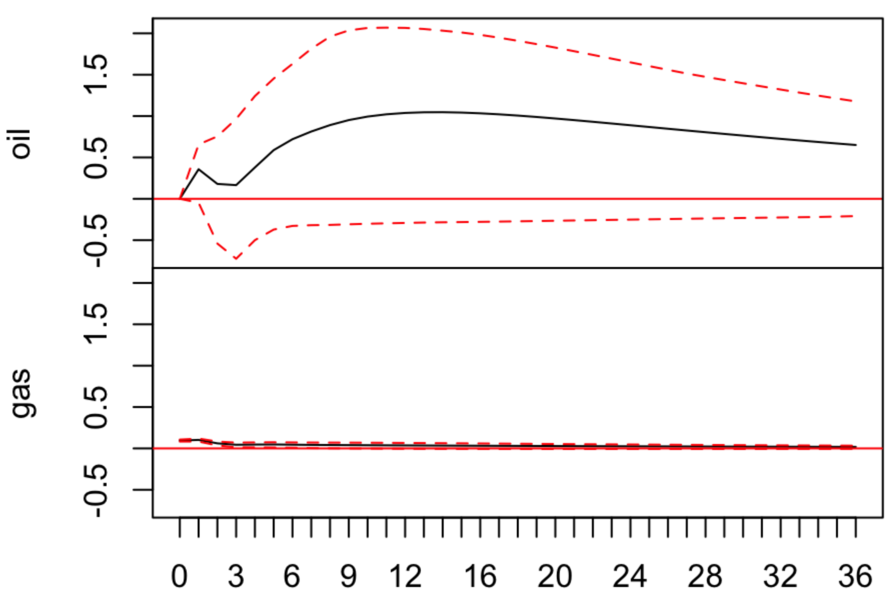
```
# Impulse Response Function
y_irf <- irf(y_m1)
plot(irf(y_m1, n.ahead = 36))
```

Orthogonal Impulse Response from oil



95 % Bootstrap CI, 100 runs

Orthogonal Impulse Response from gas



95 % Bootstrap CI, 100 runs

A “shock” in the crude oil price makes the oil price increase, and then the effect decreases. However, the “shock” in oil price does not influence the gas price significantly.
A “shock” in the gas price does not influence either the oil price or the gas price significantly.

(n)

```
# Granger Test
oil.gas <- grangertest(oil ~ gas, order = 3)
oil.gas
```

```
## Granger causality test
##
## Model 1: oil ~ Lags(oil, 1:3) + Lags(gas, 1:3)
## Model 2: oil ~ Lags(oil, 1:3)
##   Res.Df Df      F  Pr(>F)
## 1    432
## 2    435 -3 2.1451 0.09388 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

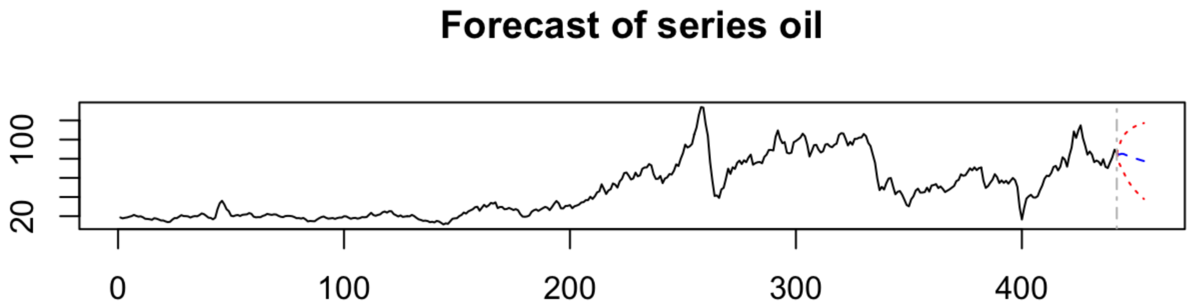
```
gas.oil <- grangertest(gas ~ oil, order = 3)
gas.oil
```

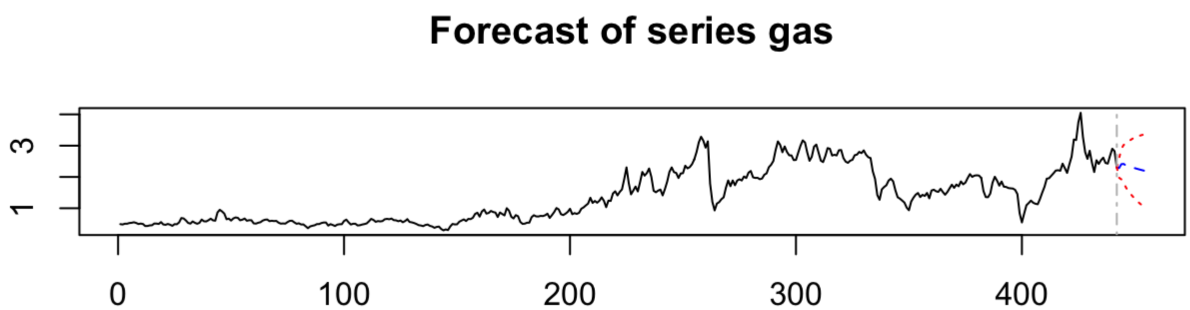
```
## Granger causality test
##
## Model 1: gas ~ Lags(gas, 1:3) + Lags(oil, 1:3)
## Model 2: gas ~ Lags(gas, 1:3)
##   Res.Df Df    F    Pr(>F)
## 1    432
## 2    435 -3 9.46 4.584e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Based on the results, we can conclude that the gasoline price does not significantly help to explain the change in the crude oil price, but the crude oil price is statistically significant in predicting the gasoline price. This is consistent with what we discussed in the introduction, as we hypothesized that there would be a positive relationship between gasoline and crude oil prices. Crude oil is used to produce gasoline, so it is logical that crude oil prices have a significant effect on gasoline prices, and not vice versa.

(o)

```
var_forc <- predict(y_m1, n.ahead = 12)
plot(var_forc)
```





The models we built give forecasts that converge to the mean very quickly. The auto.arima and VAR models do a better job for that problem and show more patterns before converging to the mean values.For gasoline series, forecasts from three models perform more differently compared to the oil series. For the crude oil price data, the model we built and auto.arima show that the forecasts are around the mean value, but VAR gives a decreasing trend. For the gasoline price data, the model we built and auto.arima show a slightly upward trend, but VAR gives a decreasing trend.

III. CONCLUSIONS & FUTURE WORK

Based on the result generated from the VARselect() function, we choose VA(3) as our final model in order to analyze the dynamics of the prices of crude oil and conventional gasoline. In the long run, both prices have been historically increasing, but they experience clear seasonal fluctuations by the month; gas demand is larger in the summer.

Our series decomposition of the gas and oil time series revealed a (1) slight upward trend, (2) multiplicative seasonality, and (3) present cycles. It appears that the variance of the data varies a lot, so we attempted to fit a multiplicative method for seasonality. Although we suspected seasonality, based on the STL decomposition, the linear model fits showed the absence of significant seasonal dummies, so we concluded that we should not represent them in our future models. When we fit auto arima models to our data, we found that the residuals mimicked white noise, better than our linear model. However, one problem we faced was that in the residual diagnostics, we found that the variance of the error term was not constant.

When comparing the Mean Average Percentage Error (MAPE) of the 12-step linear forecast and 12-step auto arima forecast, we found that the MAPE for the arima forecast was smaller than the latter. Thus, we concluded that the auto arima forecast produces more accurate estimates. By combining both forecasts, we found that the

MAPE was actually quite smaller than both of the MAPEs of the forecasts mentioned above. This means that overall, the combined forecast gave us even more accurate results.

Based on the results of our Impulse Response Function, we found that a “shock” in crude oil prices affects gas prices. However, the “shock” in oil price does not influence the gas price significantly, and vice versa. After performing a Granger-Test, we concluded that the results mentioned above were consistent with the results from the test (high p-value). As gas and oil prices are positively related, and crude oil is used to produce gasoline, this finding makes sense.

Additionally, we implemented a VAR forecast to our model. We can see that these 3 forecasts vary in direction only slightly. The linear model and auto arima forecasts seem to converge to the mean rather quickly while the VAR forecast does not.

As gas prices continue to fluctuate, it is imperative to analyze the changes in crude oil prices as a driving factor of gas price changes.

IV. REFERENCES

Our data is sourced from the U.S. Energy Information Administration’s database (EIA):
https://www.eia.gov/dnav/pet/pet_pri_spt_s1_m.htm (https://www.eia.gov/dnav/pet/pet_pri_spt_s1_m.htm)