# 4423 Advanced Macroeconometrics 1 - Assignment 1

Unterweger L., Pirich G., Tijani S.

2023-04-18

## Excercise 1: FRED-MD

*Download the current version of the FRED-MD database and load it into R (or another statistical software of your choice). Note that the second line in the CSV-file denotes the suggested transformation, you have to remove it.*

```
fredmd <- read.csv("FRED_MD_data.csv")[-1,]
head(fredmd[,1:6])
```

```
##    sasdate      RPI W875RX1 DPCERA3M086SBEA CMRMTSPLx  RETAILx
## 2 1/1/1959 2442.158  2293.2          17.272  292266.4 18235.77
## 3 2/1/1959 2451.778  2301.5          17.452  294424.7 18369.56
## 4 3/1/1959 2467.594  2318.5          17.617  293418.7 18523.06
## 5 4/1/1959 2483.671  2334.9          17.553  299322.8 18534.47
## 6 5/1/1959 2498.026  2350.4          17.765  301364.3 18679.66
## 7 6/1/1959 2505.788  2357.4          17.831  301348.8 18849.75
```

### Subquestion (a)

*Create a function that takes a vector containing observations of a time series as input and returns a dataframe with the following transformed series in its columns as output: - the original time series in its raw form; -the log-transformed time series; - month-on-month growth rates in percent; - year-on-year growth rates in percent; - the first lag of the year-on-year growth rates of the time series*

```
ts_transform <- function(x) {
  output <- data.frame(matrix(NA,    # Create empty data frame
                              nrow = NROW(x),
                              ncol = 0))
  output$raw <- x
  output$log <- log(x)
  output$mom <- ((x-lag(x))/lag(x))*100
  output$yoy <- ((x-lag(x, 12))/lag(x, 12))*100
  output$yoy_1stlag <- lag((x-lag(x, 12))/lag(x, 12))*100
  return(output)
}
```

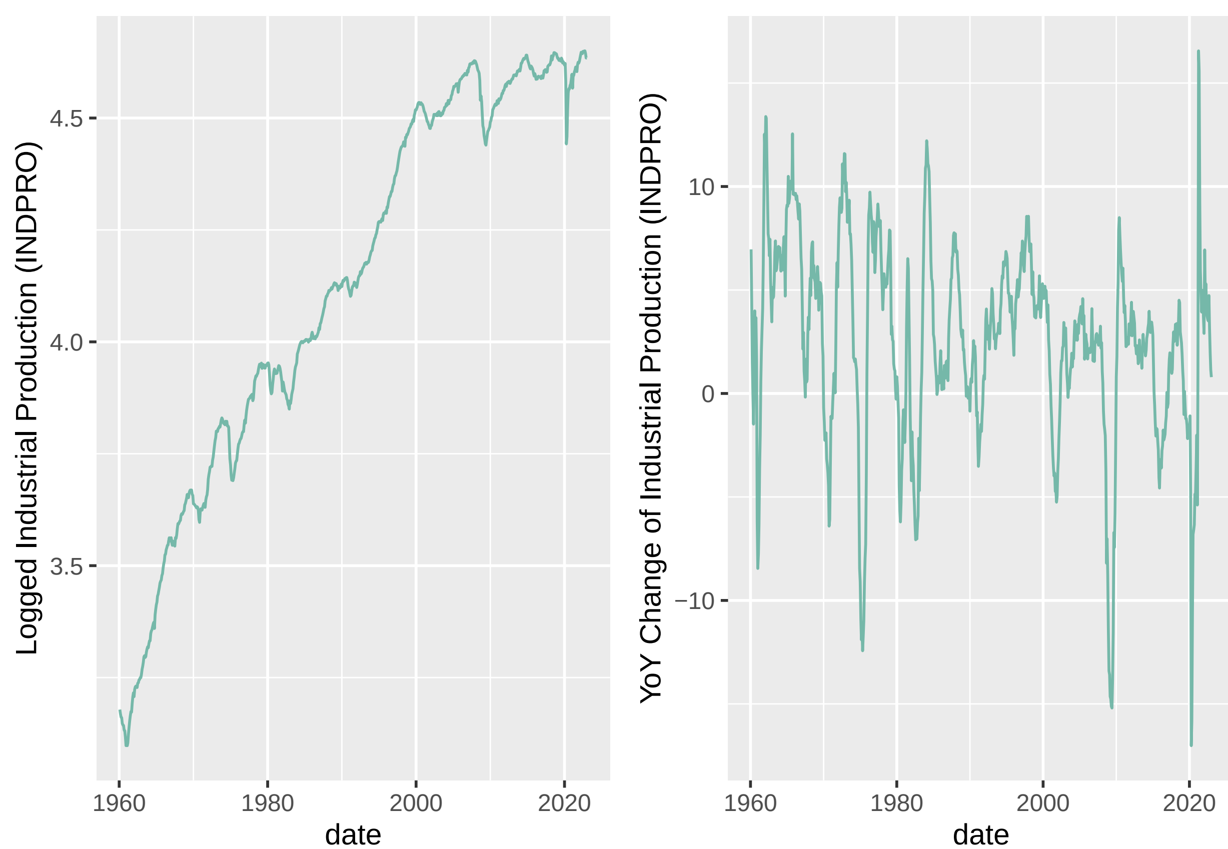
### Subquestion (b)

*Use the created function to create a dataframe with the various transformation for US industrial production (mnemonic INDPRO), plot the logged time series and the yearly changes produced by the function. Briefly describe the properties of the time series.*

```
industrial_prod <- cbind(date=as.Date(fredmd$sasdate, "%m/%d/%Y"), ts_transform(fredmd$INDPRO))
head(industrial_prod)
```

```
##         date     raw      log       mom yoy yoy_1stlag
## 1 1959-01-01 22.0151 3.091729        NA  NA         NA
## 2 1959-02-01 22.4463 3.111126 1.9586556  NA         NA
## 3 1959-03-01 22.7696 3.125426 1.4403265  NA         NA
## 4 1959-04-01 23.2547 3.146507 2.1304722  NA         NA
## 5 1959-05-01 23.6050 3.161459 1.5063622  NA         NA
## 6 1959-06-01 23.6319 3.162597 0.1139589  NA         NA
```

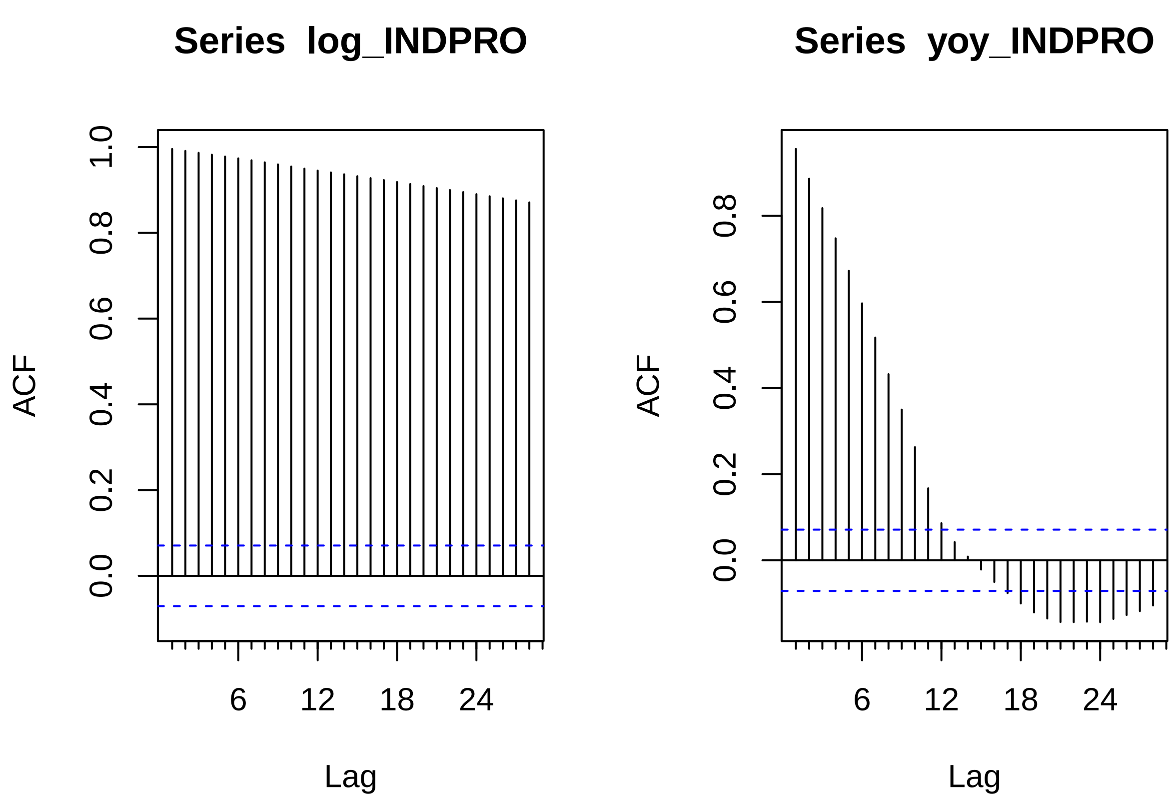
```
p1 <- ggplot(na.omit(industrial_prod), aes(x=date,y=log))+
  geom_line( color="#69b3a2", alpha=0.9, linetype=1) +
  ylab("Logged Industrial Production (INDPRO)")

p2 <- ggplot(na.omit(industrial_prod), aes(x=date,y=yoy))+
  geom_line( color="#69b3a2", alpha=0.9, linetype=1) +
  ylab("YoY Change of Industrial Production (INDPRO)")

grid.arrange(p1, p2, ncol=2, nrow = 1)
```



### Subquestion (c)

*Using suitable functions from the stats and urca package, assess the properties of both logged industrial production and its yearly growth rate. Plot the autocorrelation function and perform Dickey-Fuller tests to test for a unit root (note the different specifications, i.e. including a drift or a trend), interpret the results.*

```
log_INDPRO <- ts(industrial_prod$log, frequency = 12, start = c(1959,1,1))
yoy_INDPRO <- ts(industrial_prod$yoy, frequency = 12, start = c(1959,1,1))

par(mfrow=c(1,2))
Acf(log_INDPRO)
Acf(yoy_INDPRO)
```




The ACF plots show us that. . . .

```
summary(ur.df(na.omit(industrial_prod$yoy), lags = 0, type="none")) # Process is stationary
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression none
##
##
```

```
## Call:
## lm(formula = z.diff ~ z.lag.1 - 1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.3097  -0.5880   0.0529   0.7056  15.5830
##
## Coefficients:
##          Estimate Std. Error t value Pr(>|t|)
## z.lag.1 -0.035909   0.009356  -3.838 0.000134 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.405 on 755 degrees of freedom
## Multiple R-squared:  0.01914,    Adjusted R-squared:  0.01784
## F-statistic: 14.73 on 1 and 755 DF,  p-value: 0.0001344
##
##
## Value of test-statistic is: -3.838
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau1 -2.58 -1.95 -1.62
```

```
summary(ur.df(na.omit(industrial_prod$log), lags = 0, type="none")) # Cannot reject H0
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression none
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 - 1)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.143587 -0.003563  0.000648  0.004666  0.058851
##
## Coefficients:
##          Estimate Std. Error t value Pr(>|t|)
## z.lag.1 4.514e-04  8.747e-05   5.161 3.13e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01002 on 767 degrees of freedom
## Multiple R-squared:  0.03356,    Adjusted R-squared:  0.0323
## F-statistic: 26.64 on 1 and 767 DF,  p-value: 3.132e-07
##
##
## Value of test-statistic is: 5.161
##
```

```
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau1 -2.58 -1.95 -1.62
```

```r
summary(ur.df(na.omit(industrial_prod$log), lags = 0, type="drift"))
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression drift
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.142265 -0.003909  0.000693  0.004596  0.059784
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0128789  0.0032689   3.940  8.9e-05 ***
## z.lag.1     -0.0026467  0.0007911  -3.345 0.000861 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.009922 on 766 degrees of freedom
## Multiple R-squared:  0.0144, Adjusted R-squared:  0.01311
## F-statistic: 11.19 on 1 and 766 DF,  p-value: 0.0008614
##
##
## Value of test-statistic is: -3.3455 21.3309
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau2 -3.43 -2.86 -2.57
## phi1  6.43  4.59  3.78
```

```r
summary(ur.df(na.omit(industrial_prod$log), lags = 0, type="trend"))
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
```

```
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.142730 -0.003882  0.000696  0.004660  0.059091
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.870e-02  1.047e-02   1.786   0.0746 .
## z.lag.1     -4.412e-03  3.120e-03  -1.414   0.1578
## tt           3.726e-06  6.370e-06   0.585   0.5587
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.009927 on 765 degrees of freedom
## Multiple R-squared:  0.01484,    Adjusted R-squared:  0.01227
## F-statistic: 5.762 on 2 and 765 DF,  p-value: 0.003282
##
##
## Value of test-statistic is: -1.4141 14.3224 5.7624
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -3.96 -3.41 -3.12
## phi2  6.09  4.68  4.03
## phi3  8.27  6.25  5.34
```

The augmented dickey fuller tests…

## Subquestion (d)

*Estimate a suitable AR model (e.g. using the ar.ols() function) for the stationary time series (as determined in the previous point). How is the lag order determined by default? Use the estimated model to produceforecasts for the next year and plot them. Interpret their behaviour (i.e. are they converging towards a certain value? What could that be?). Use the produced forecasts to also forecast the change in the original time series.*

```
mod_log <- ar.ols(industrial_prod$log, na.action = na.omit,
                  demean = TRUE, intercept = TRUE)
mod_log
```

```
##
## Call:
## ar.ols(x = industrial_prod$log, na.action = na.omit, demean = TRUE,      intercept = TRUE)
##
## Coefficients:
##       1        2        3        4        5        6        7        8
##  1.2637  -0.3294   0.0921   0.0518  -0.0941   0.0529  -0.0165  -0.0789
##       9       10       11       12       13       14       15       16
##  0.0961   0.0492  -0.1650   0.0850  -0.0609   0.0292   0.0281  -0.0108
##      17       18       19       20       21       22       23       24
##  0.0137   0.0036  -0.0460   0.0232  -0.0230   0.0326   0.0277  -0.1539
##      25
##  0.1268
```

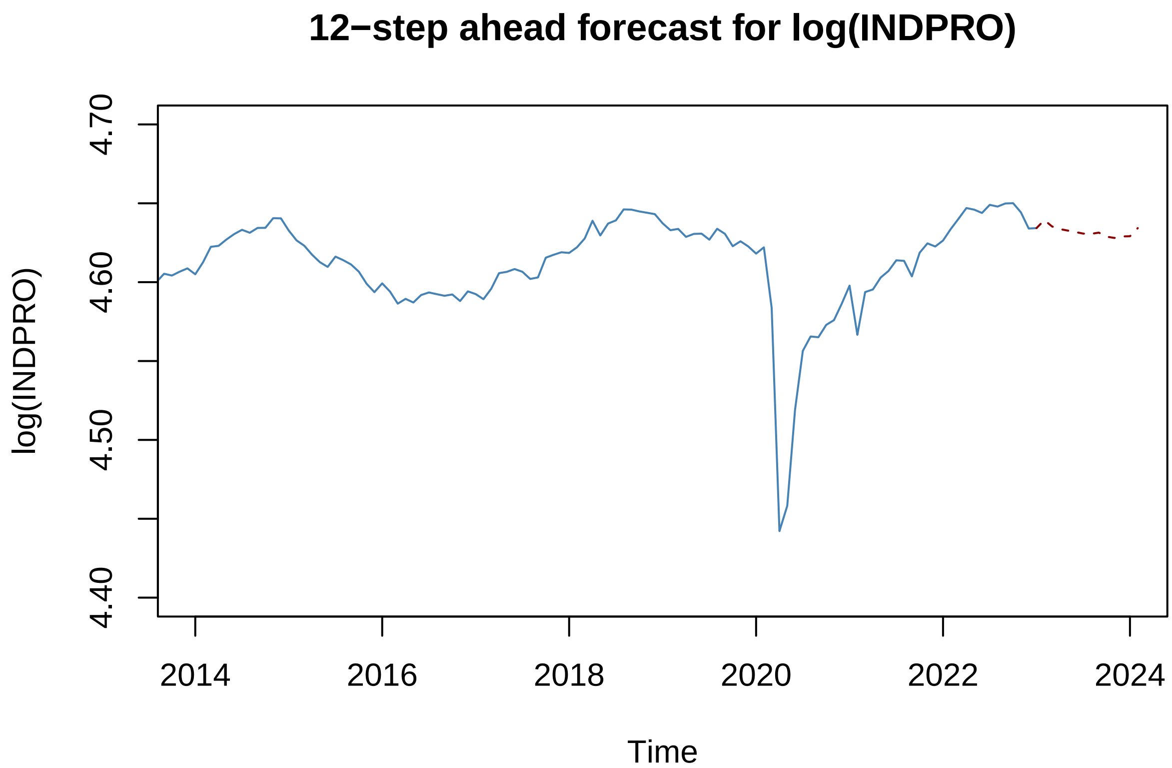
```
##
## Intercept: 0.001837 (0.0004059)
##
## Order selected 25  sigma^2 estimated as  7.871e-05
```

```
mod_yoy <- ar.ols(industrial_prod$yoy, na.action = na.omit,
                  demean = FALSE, intercept = FALSE)
mod_yoy
```

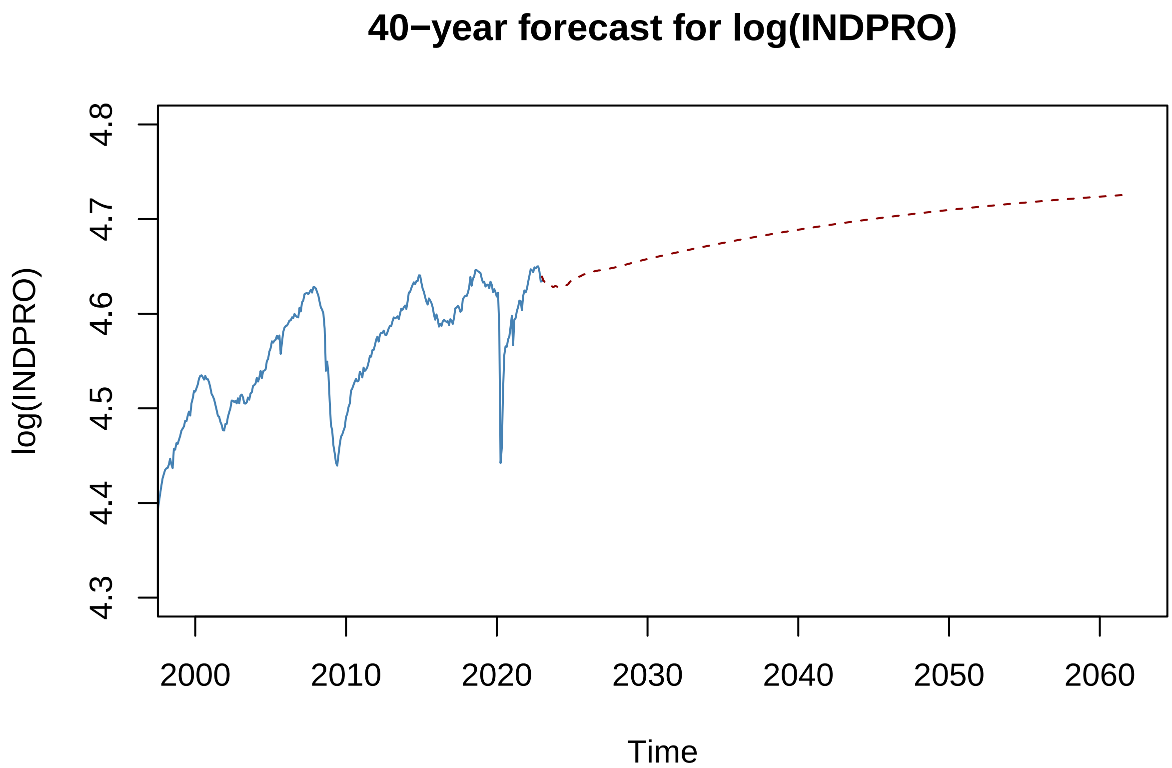
```
##
## Call:
## ar.ols(x = industrial_prod$yoy, na.action = na.omit, demean = FALSE,      intercept = FALSE)
##
## Coefficients:
##       1        2        3        4        5        6        7        8
##  1.2925  -0.3567   0.0895   0.0171  -0.0735   0.0812  -0.0743  -0.0481
##       9       10       11       12       13       14       15       16
##  0.1293   0.0448  -0.1444  -0.6125   0.8058  -0.2258   0.1347  -0.0914
##      17       18       19       20       21       22       23       24
##  0.0290   0.0253  -0.1213   0.0522   0.0187   0.1260  -0.0878  -0.5378
##      25       26
##  0.6397  -0.1440
##
## Order selected 26  sigma^2 estimated as  1.042
```

By default the lag order is determined by the AIC information criterion. For the log of INDPRO a lag order of 25 is estimated. The model for the year on year changes is estimated with lag order 26.

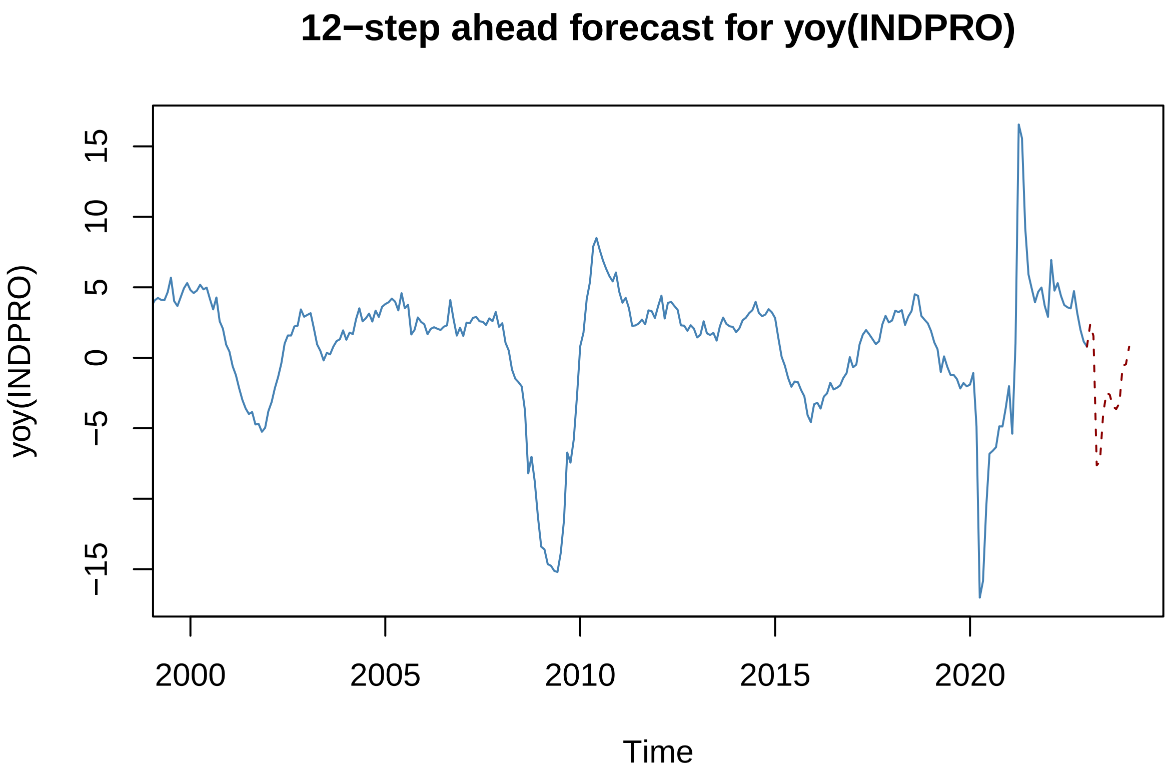
```
predict_mod_log <- predict(mod_log, na.omit(industrial_prod$log), n.ahead = 12)

ts_log <- ts(industrial_prod$log, start=c(1959, 1), end=c(2023,1), frequency=12)
ts_log_pred <- ts(c(tail(ts_log,n=1), predict_mod_log$pred),
                  start = c(2023, 1), end=c(2024,2), frequency=12)

plot.ts(ts_log,
        xlim = c(2014, 2024),
        ylim = c(4.4,4.7),
        xlab = "Time",
        ylab = "log(INDPRO)",
        main = "12-step ahead forecast for log(INDPRO)",
        col = "steelblue") #plotting th predicted values) #plotting th predicted values
points(ts_log_pred, type = "l",lty=2, col = "darkred")
```



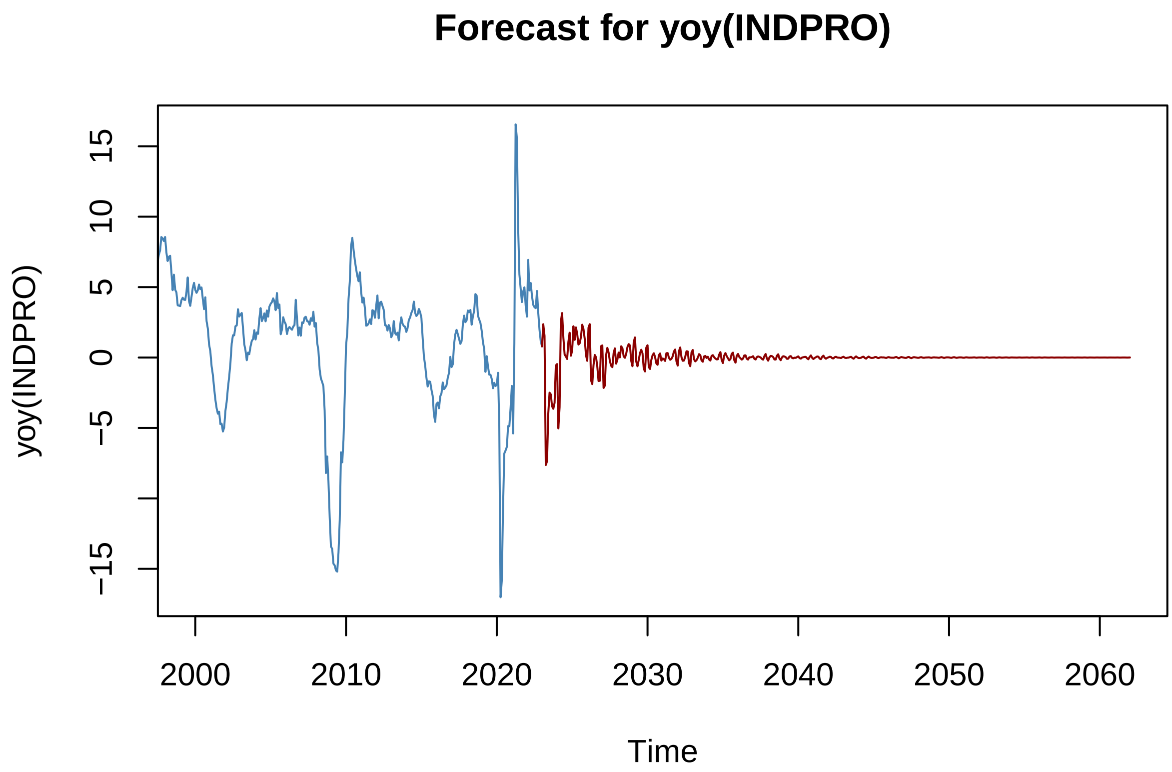

```
predict_mod_log <- predict(mod_log, na.omit(industrial_prod$log), n.ahead = 480)

ts_log <- ts(industrial_prod$log, start=c(1959, 1), end=c(2023,1), frequency=12)
ts_log_pred <- ts(predict_mod_log$pred, start = c(2023, 1), end=c(2062,1), frequency=12)

plot.ts(ts_log,
        xlim = c(2000, 2062),
        ylim = c(4.3,4.8),
        xlab = "Time",
        ylab = "log(INDPRO)",
        main = "40-year forecast for log(INDPRO)",
        col = "steelblue") #plotting the predicted values
points(ts_log_pred, type = "l", lty=2, col = "darkred")
```




```
predict_mod_yoy <- predict(mod_yoy, na.omit(industrial_prod$yoy), n.ahead = 12)

ts_yoy <- ts(industrial_prod$yoy, start=c(1959, 1), end=c(2023,1), frequency=12)
ts_yoy_pred <- ts(c(tail(na.omit(industrial_prod$yoy),n=1),predict_mod_yoy$pred),
                  start = c(2023, 1), end=c(2024,2), frequency=12)

plot.ts(ts_yoy,
        xlim = c(2000, 2024),
        xlab = "Time",
        ylab = "yoy(INDPRO)",
        main = "12-step ahead forecast for yoy(INDPRO)",
        col = "steelblue") #plotting the predicted values
points(ts_yoy_pred, type = "l", lty=2, col = "darkred")
```




```
predict_mod_yoy <- predict(mod_yoy, na.omit(industrial_prod$yoy), n.ahead = 480)

ts_yoy <- ts(industrial_prod$yoy, start=c(1959, 1), end=c(2023,1), frequency=12)
ts_yoy_pred <- ts(c(tail(na.omit(industrial_prod$yoy),n=1),predict_mod_yoy$pred),
                  start = c(2023, 1), end=c(2062,1), frequency=12)

plot.ts(ts_yoy,
        xlim = c(2000, 2062),
        xlab = "Time",
        ylab = "yoy(INDPRO)",
        main = "Forecast for yoy(INDPRO)",
        col = "steelblue") #plotting th predicted values
points(ts_yoy_pred, type = "l", col = "darkred")
```




```
last_origvalues <- tail(na.omit(industrial_prod$raw),12)
pred_yoygrowth <- predict_mod_yoy$pred
orig_forecast <- vector()
for(i in 1:12){
  orig_forecast <- c(orig_forecast, last_origvalues[i]*(100+pred_yoygrowth[i])/100)
}
orig_forecast
```

```
##  [1] 105.33552 105.16440  96.31155  96.49389  99.83880 101.87643 101.63979
##  [8] 100.95871 100.78447 100.64529 102.34614 102.47634
```

### Subquestion (e) - Bonus question

We start by defining the function.

```
rmse_ar <- function(data, lag, hold_period) {
  #Remove holdout period from the end of the sample
  data_train <- data[1:(length(data) - hold_period)]

  # Estimate the AR model
  ar_model <- ar.ols(data_train, order.max = lag, demean = TRUE, intercept = TRUE)

  # Forecast for the holdout period
  data_test <- data[(length(data) - hold_period + 1):length(data)]
```

```r
  ar_forecasts <- predict(ar_model, n.ahead = hold_period)

  # Compute the RMSE
  rmse <- sqrt(mean((data_test - ar_forecasts$pred)^2))
  return(rmse)
}


#RMSE for 50 different lag orders and returning the minimal
rmse <- list()
for (x in 1:50) {
  rmse[[x]] <- rmse_ar(na.omit(industrial_prod$yoy), x, 6)
}
which.min((rmse))
```

```
## [1] 16
```

```r
#RMSE for 50 different lag orders and returning the minimal
for (x in 1:50) {
  rmse[[x]] <- rmse_ar(na.omit(industrial_prod$yoy), x, 12)
}
which.min((rmse))
```

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

```r
#RMSE for 50 different lag orders and returning the minimal
rmse <- list()
for (x in 1:50) {
  rmse[[x]] <- rmse_ar(na.omit(industrial_prod$log), x, 6)
}
which.min((rmse))
```

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

```r
#RMSE for 50 different lag orders and returning the minimal
for (x in 1:50) {
  rmse[[x]] <- rmse_ar(na.omit(industrial_prod$log), x, 12)
}
which.min((rmse))
```

```
## [1] 11
```

We run the function for the two time series and compare the RMSE for AR models up to order 50. For a holdout period of 6 months we find that the year on year growth rates are best predicted with an AR(16) model. For a holdout period of 12 months, an AR(1) model produces the lowest RMSE.

For the log(industrial production) the lowest RMSE is produced by an AR(1) model for a 6 month holdout period. Over 12 months an AR(11) model serves as the best predictor.

Remarkably, based on the forecast performance, the optimal lag order is significantly lower than the lag order chosen by the AIC or BIC criterion. The AIC would have selected a model of order 25 (for the logged time series) and 26 (for the year on year growth rates).

## Excercise 2 - Killian and Park (2009)

*Read Kilian & Park (2009), who discuss the effects of oil price shocks on the US stock market, focus on Sections 2 and 3.1-3.3. Load the provided data by Kilian & Park (2009), which contains a measure of change in oil production, a measure of real economic activity, the real price of oil, and changes in real US dividend growth from 1973M1 to 2016M12.*

```
data <- read.table("data_kilian_park_2009.txt")
#("Global Oil Production Change","Global Real Activity", "Real Price of Oil", "US Stock Returns")
colnames(data) <- c("GOPC","GRA","RPO","USSR")
head(data)
```

```
##       GOPC     GRA      RPO    USSR
## 1  11.8773 34.5887 -46.3143 -1.3498
## 2   1.4191 40.0667 -46.6013 -0.3862
## 3   1.1777 42.5462 -45.3973  1.2771
## 4  27.4551 46.6761 -42.1724 -2.4366
## 5 -13.1104 50.6190 -39.8859 -0.2239
## 6  36.2581 51.5436 -39.3027  0.6786
```

### Subquestion (a)

*Using the the packages vars in R (or an equivalent one in another language), estimate the VAR described in section 2.2 using the variables in the same order as specified by Kilian and Park (2009).*

```
mod1 <- VAR(data, p=24, type="const")
```

Due to the enormity of the VAR consisting of 24 lags and 4 time series, we decided to exclude the output at this point of the assignment. The estimated coefficients are instead given in the appendix.

### Subquestion (b)

*Using the estimated VAR, compute impulse response functions (take a look at the irf() function in vars, it uses the same identification scheme as Kilian & Park (2009) propose (recursive ordering based on a Cholesky decomposition of the vcov-matrix of the errors) by default. Replicate Figure 1 and the lower panel in figure 3 of Kilian & Park (2009).3 Interpret the results.*

#### Figure 1

```
# TO DO
```

#### Figure 3

```
# TO TO
```

## Subquestion (c)

*Calculate forecast error variance decompositions for the included variables (take a look at the fevd() function in vars). Replicate Table 2 of Kilian and Park (2009). Interpret the results.*

```
fevd_mod1 <- vars::fevd(mod1, n.ahead=150)
table2 <- round(fevd_mod1$USSR[c(1,2,3,12,150),]*100,2) # We decided to use 150 as a proxy for infinity
rownames(table2) <- c("h=1","h=2","h=3","h=12","h=Inf")
table2
```

```
##       GOPC  GRA  RPO  USSR
## h=1   0.20 0.16 1.69 97.94
## h=2   0.55 0.36 2.09 97.00
## h=3   0.76 0.48 2.12 96.64
## h=12  2.80 6.83 4.53 85.84
## h=Inf 6.63 8.38 7.93 77.06
```

## Subquestion (d)

*Note that the dataset provided misses US stock market returns (due to the licensing of the underlying time series). Look for alternative data on the US stock market, create a variable similar to the one used by Kilian & Park (2009). Re-estimate the model and replicate Figure 1 again as well as the top panel of Figure 3 and Table 1.5 Interpret the results.*

In this exercise we decided to use the S&P 500 returns as well as the CPI data from the FRED database to create monthly real stock returns.

```
fred <- read.csv("FRED_MD_data.csv")[,c("sasdate","S.P.500","CPIAUCSL")][-1,]
str(fred)
```

```
## 'data.frame':    770 obs. of  3 variables:
##  $ sasdate : chr  "1/1/1959" "2/1/1959" "3/1/1959" "4/1/1959" ...
##  $ S.P.500 : num  55.6 54.8 56.2 57.1 58 ...
##  $ CPIAUCSL: num  29 29 29 29 29 ...
```

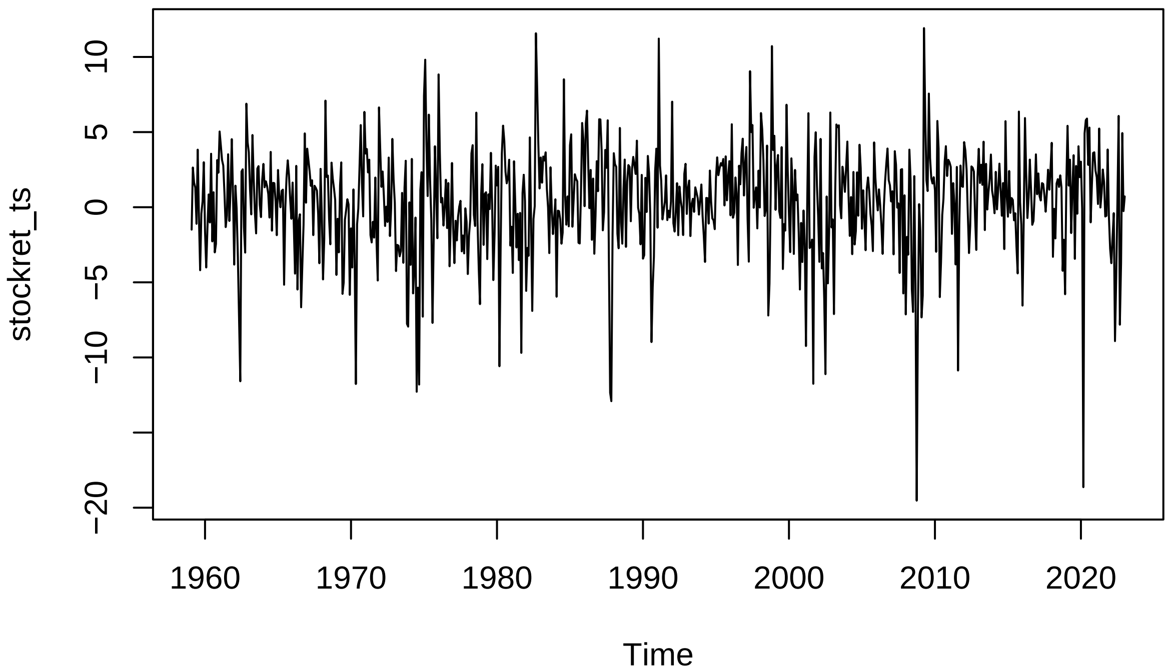
```
# Extract SP500 Growth and Inflation
ret <- ts_transform(fred$S.P.500)$mom
inf <- ts_transform(fred$CPIAUCSL)$mom

# Compute deflated stock returns
stockret <- ret - inf
head(stockret, n=10)
```

```
##  [1]         NA -1.4937564  2.6413340  1.6392707  1.2990903 -1.1037107
##  [7]  3.8305679 -0.6720489 -4.1961193 -0.4295228
```

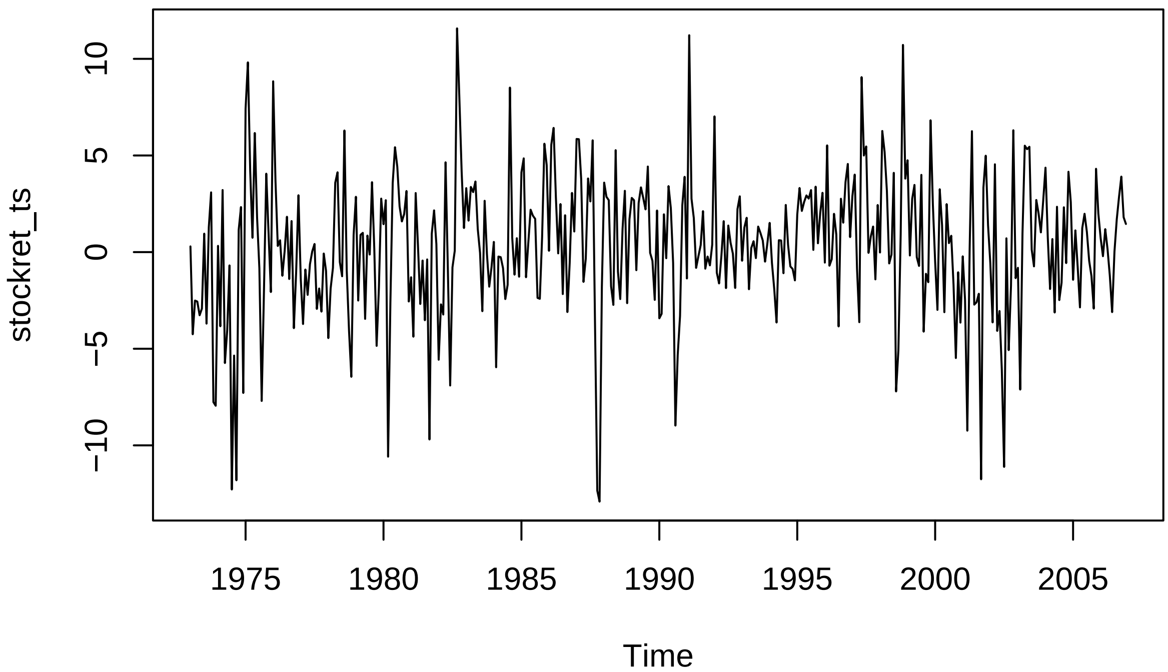
```
# Create time series
stockret_ts <- ts(stockret, frequency=12, start=c(1959,1))
head(stockret_ts)
```

```
##            Jan       Feb       Mar       Apr       May       Jun
## 1959        NA -1.493756  2.641334  1.639271  1.299090 -1.103711
```

```
plot(stockret_ts)
```




```
stockret_ts <- window(stockret_ts, start=c(1973,1), end=c(2006,12))
plot(stockret_ts)
```




Using this data, we replicate the VAR given above. Again, the estimated coefficients are given in the appendix.

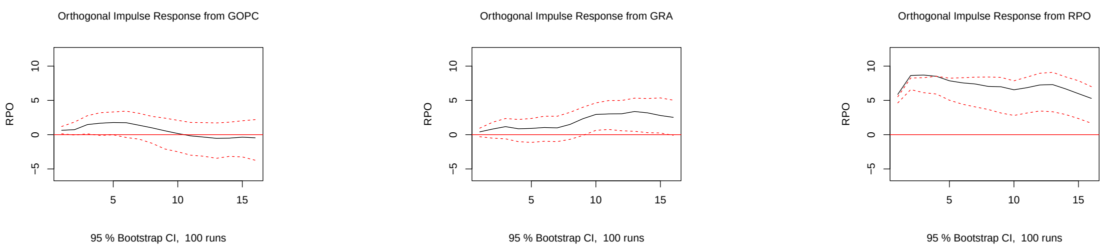
```
data$adj_stock_returns <- stockret_ts[-1]

df_2 <- data[,c("GOPC", "GRA", "RPO", "adj_stock_returns")]

var_rep <- VAR(df_2, p = 24, type = "const")
```

**Figure 1**

```
# Supply Shock
irf_supply_specific_shocks <- irf(var_rep, impulse = "GOPC", response = "RPO", boot = TRUE, cumulative =

# Inverting Signs
for(x in 1:length(irf_supply_specific_shocks$irf$GOPC)){
  irf_supply_specific_shocks$irf$GOPC[x,] <- irf_supply_specific_shocks$irf$GOPC[x,]*(-1)
  irf_supply_specific_shocks$Upper$GOPC[x,] <- irf_supply_specific_shocks$Upper$GOPC[x,]*(-1)
  irf_supply_specific_shocks$Lower$GOPC[x,] <- irf_supply_specific_shocks$Lower$GOPC[x,]*(-1)
}

# Aggregate-demand shock
irf_agg_demand_shocks <- irf(var_rep, impulse = "GRA", response = "RPO", boot = TRUE, cumulative = FALS
```

```r
# Oil specific-demand shock
irf_oi_specific_demand_shocks <- irf(var_rep, impulse = "RPO", response = "RPO", boot = TRUE, cumulative
```

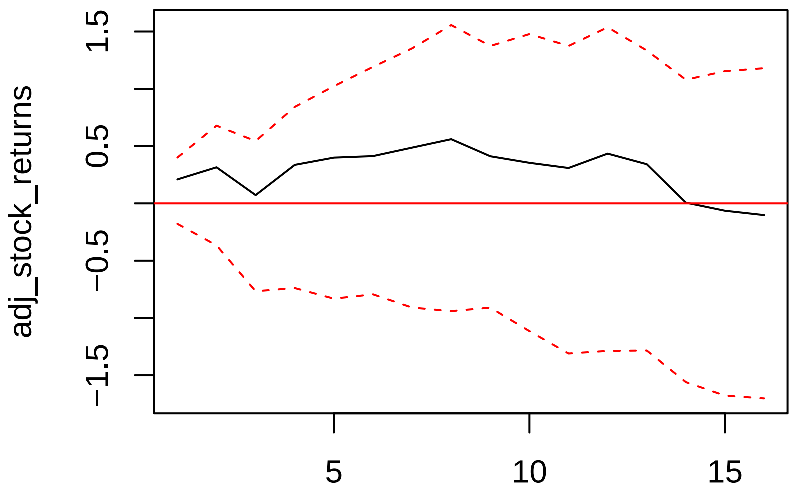
```r
# Figure 1
par(mar = c(4, 4, .1, .1))
plot(irf_supply_specific_shocks, ylim=c(-6,12))

plot(irf_agg_demand_shocks, ylim=c(-6,12))

plot(irf_oi_specific_demand_shocks, ylim=c(-6,12))
```




### Top Panel of Figure 3

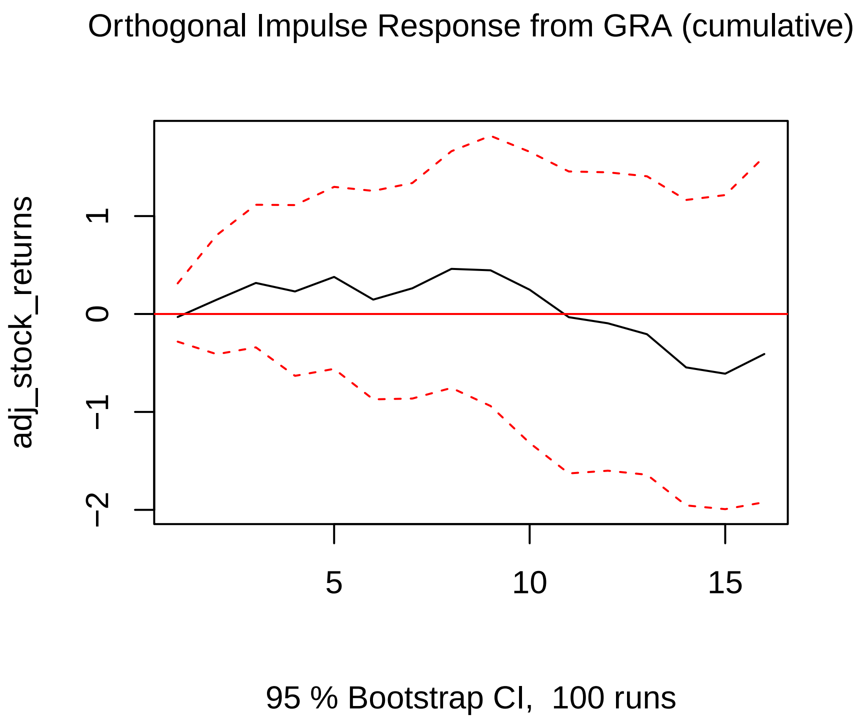
```r
# Supply shock
irf_supply_shock_stock <- irf(var_rep, impulse = "GOPC", response = "adj_stock_returns", boot = TRUE, c

# Inverting signs
for(x in 1:length(irf_supply_shock_stock$irf$GOPC)){
  irf_supply_shock_stock$irf$GOPC[x,] <- irf_supply_shock_stock$irf$GOPC[x,]*(-1)
  irf_supply_shock_stock$Upper$GOPC[x,] <- irf_supply_shock_stock$Upper$GOPC[x,]*(-1)
  irf_supply_shock_stock$Lower$GOPC[x,] <- irf_supply_shock_stock$Lower$GOPC[x,]*(-1)
}

# Aggregate-demand shock
irf_agg_demand_shock_stock <- irf(var_rep, impulse = "GRA", response = "adj_stock_returns", boot = TRUE

# Oil specific-demand shock
irf_oi_specific_demand_shock <- irf(var_rep, impulse = "RPO", response = "adj_stock_returns", boot = TRU


# Figure 3
par(mfrow = c(1, 3))
plot(irf_supply_shock_stock)
```

Orthogonal Impulse Response from GOPC (cumulative)




95 % Bootstrap CI,  100 runs

```
plot(irf_agg_demand_shock_stock)
```




```
plot(irf_oi_specific_demand_shock)
```

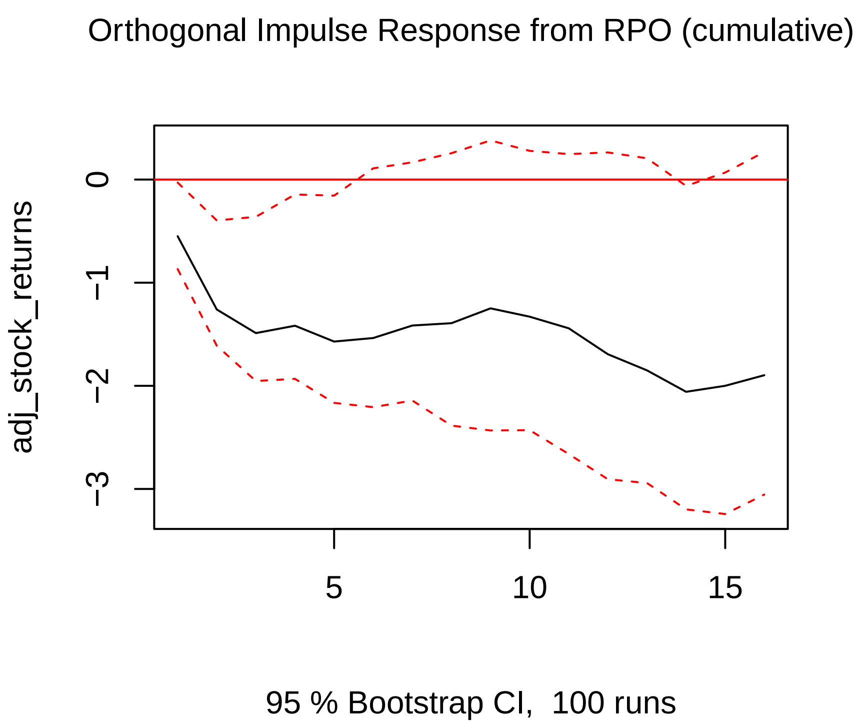



**Table 1**

Similar to before, we replicate table 1 using our data on S&P500 returns.

```
fevd_stock <- fevd(var_rep, n.ahead = 150)
fevd_stock <- round(fevd_stock$adj_stock_returns[c(1,2,3,12,150),]*100,2) # We decided to use 150 as a 
colnames(fevd_stock) <- c("GOPC","GRA","RPO","SP500 Returns")
rownames(fevd_stock) <- c("h=1","h=2","h=3","h=12","h=Inf")
fevd_stock
```

```
##        GOPC  GRA  RPO SP500 Returns
## h=1    0.38 0.01 2.64         96.97
## h=2    0.43 0.25 6.34         92.97
## h=3    0.89 0.48 6.68         91.96
## h=12   1.78 2.35 7.42         88.45
## h=Inf  5.84 6.17 9.04         78.96
```

# Appendix

Here are some additional outputs which are way too long for the main document.

```r
# Summary of model 1
summary(mod1)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: GOPC, GRA, RPO, USSR
## Deterministic variables: const
## Sample size: 383
## Log Likelihood: -4332.408
## Roots of the characteristic polynomial:
## 0.9869 0.9761 0.9761 0.9749 0.9749 0.9745 0.9745 0.9704 0.9704 0.9696 0.9696 0.9693 0.9693 0.9685 0.9
## Call:
## VAR(y = data, p = 24, type = "const")
##
##
## Estimation results for equation GOPC:
## =====================================
## GOPC = GOPC.l1 + GRA.l1 + RPO.l1 + USSR.l1 + GOPC.l2 + GRA.l2 + RPO.l2 + USSR.l2 + GOPC.l3 + GRA.l3 +
##
##           Estimate Std. Error t value Pr(>|t|)
## GOPC.l1  -0.129498   0.058658  -2.208 0.028060 *
## GRA.l1   -0.221827   0.266750  -0.832 0.406334
## RPO.l1   -0.477407   0.190269  -2.509 0.012656 *
## USSR.l1  -0.453988   0.975409  -0.465 0.641975
## GOPC.l2  -0.132503   0.057952  -2.286 0.022961 *
## GRA.l2    0.460494   0.409188   1.125 0.261369
## RPO.l2    0.854052   0.329495   2.592 0.010032 *
## USSR.l2   1.170992   0.977994   1.197 0.232165
## GOPC.l3  -0.285100   0.058954  -4.836 2.17e-06 ***
## GRA.l3   -0.411880   0.402039  -1.024 0.306475
## RPO.l3   -0.574777   0.350788  -1.639 0.102410
## USSR.l3   0.467702   0.986845   0.474 0.635907
## GOPC.l4  -0.103021   0.061469  -1.676 0.094836 .
## GRA.l4    0.032507   0.399724   0.081 0.935241
## RPO.l4    0.276691   0.351854   0.786 0.432294
## USSR.l4   1.384134   1.021778   1.355 0.176604
## GOPC.l5  -0.256445   0.059789  -4.289 2.46e-05 ***
## GRA.l5    0.246985   0.401258   0.616 0.538697
## RPO.l5   -0.269018   0.347805  -0.773 0.439881
## USSR.l5   1.252854   1.031197   1.215 0.225387
## GOPC.l6  -0.080966   0.061146  -1.324 0.186510
## GRA.l6   -0.289052   0.401867  -0.719 0.472560
## RPO.l6    0.535320   0.348479   1.536 0.125604
## USSR.l6   0.671933   1.031420   0.651 0.515270
## GOPC.l7  -0.119813   0.060060  -1.995 0.047005 *
## GRA.l7    0.575549   0.400876   1.436 0.152172
## RPO.l7   -0.243980   0.350685  -0.696 0.487166
## USSR.l7   0.093727   1.033083   0.091 0.927774
```

```
## GOPC.l8  -0.072798   0.059515  -1.223 0.222263
## GRA.l8   -0.273770   0.400626  -0.683 0.494935
## RPO.l8   -0.233840   0.351327  -0.666 0.506210
## USSR.l8  -0.069432   1.052360  -0.066 0.947442
## GOPC.l9   0.096222   0.059203   1.625 0.105204
## GRA.l9   -0.215352   0.398761  -0.540 0.589581
## RPO.l9    0.126854   0.351486   0.361 0.718436
## USSR.l9   1.658440   1.048687   1.581 0.114882
## GOPC.l10 -0.035560   0.058108  -0.612 0.541045
## GRA.l10   0.067917   0.391630   0.173 0.862443
## RPO.l10   0.182622   0.348680   0.524 0.600856
## USSR.l10 -0.061630   1.040533  -0.059 0.952811
## GOPC.l11 -0.114046   0.058097  -1.963 0.050611 .
## GRA.l11   0.062443   0.385908   0.162 0.871571
## RPO.l11  -0.315856   0.351091  -0.900 0.369068
## USSR.l11 -0.145423   1.031966  -0.141 0.888034
## GOPC.l12  0.209483   0.058398   3.587 0.000393 ***
## GRA.l12  -0.166625   0.382032  -0.436 0.663053
## RPO.l12   0.406722   0.354808   1.146 0.252623
## USSR.l12 -0.108427   1.022965  -0.106 0.915662
## GOPC.l13 -0.022224   0.058745  -0.378 0.705474
## GRA.l13   0.333735   0.379553   0.879 0.379986
## RPO.l13  -0.225130   0.350101  -0.643 0.520710
## USSR.l13  0.154391   1.005844   0.153 0.878117
## GOPC.l14 -0.095299   0.058918  -1.617 0.106878
## GRA.l14  -0.899450   0.382447  -2.352 0.019360 *
## RPO.l14  -0.177207   0.344246  -0.515 0.607113
## USSR.l14 -0.487767   1.116605  -0.437 0.662564
## GOPC.l15  0.170775   0.059296   2.880 0.004277 **
## GRA.l15   0.233980   0.385337   0.607 0.544194
## RPO.l15   0.061130   0.350925   0.174 0.861835
## USSR.l15  1.635650   1.124519   1.455 0.146895
## GOPC.l16 -0.084994   0.060090  -1.414 0.158320
## GRA.l16   0.183684   0.384496   0.478 0.633209
## RPO.l16   0.297773   0.353849   0.842 0.400756
## USSR.l16  1.174337   1.160704   1.012 0.312515
## GOPC.l17 -0.010691   0.058759  -0.182 0.855754
## GRA.l17  -0.462543   0.385539  -1.200 0.231236
## RPO.l17  -0.416688   0.356992  -1.167 0.244094
## USSR.l17  1.789003   1.166427   1.534 0.126198
## GOPC.l18 -0.031792   0.057471  -0.553 0.580570
## GRA.l18   0.993036   0.385989   2.573 0.010595 *
## RPO.l18   0.654656   0.356477   1.836 0.067328 .
## USSR.l18  0.847356   1.168428   0.725 0.468916
## GOPC.l19 -0.008979   0.057203  -0.157 0.875386
## GRA.l19  -1.500210   0.385719  -3.889 0.000125 ***
## RPO.l19  -0.615006   0.358608  -1.715 0.087432 .
## USSR.l19 -1.275622   1.130342  -1.129 0.260043
## GOPC.l20 -0.029677   0.056751  -0.523 0.601426
## GRA.l20   0.887459   0.393212   2.257 0.024766 *
## RPO.l20   0.319920   0.360016   0.889 0.374951
## USSR.l20  0.141381   1.130968   0.125 0.900604
## GOPC.l21 -0.102112   0.055969  -1.824 0.069129 .
## GRA.l21   0.636615   0.399207   1.595 0.111884
```

```
## RPO.l21  -0.277640   0.358074  -0.775 0.438761
## USSR.l21 -0.438823   1.107180  -0.396 0.692147
## GOPC.l22 -0.042981   0.054458  -0.789 0.430621
## GRA.l22  -1.130261   0.396723  -2.849 0.004704 **
## RPO.l22   0.325394   0.358129   0.909 0.364329
## USSR.l22 -0.173196   1.086725  -0.159 0.873487
## GOPC.l23 -0.020762   0.054437  -0.381 0.703192
## GRA.l23   0.830848   0.398004   2.088 0.037725 *
## RPO.l23  -0.017338   0.341044  -0.051 0.959489
## USSR.l23 -0.071287   1.078487  -0.066 0.947345
## GOPC.l24 -0.039888   0.054093  -0.737 0.461487
## GRA.l24  -0.126947   0.261165  -0.486 0.627282
## RPO.l24  -0.244923   0.193333  -1.267 0.206241
## USSR.l24 -1.610127   1.054845  -1.526 0.128013
## const     0.433333   1.149328   0.377 0.706430
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 18.91 on 286 degrees of freedom
## Multiple R-Squared: 0.3704,  Adjusted R-squared: 0.159
## F-statistic: 1.752 on 96 and 286 DF,  p-value: 0.0002078
##
##
## Estimation results for equation GRA:
## ====================================
## GRA = GOPC.l1 + GRA.l1 + RPO.l1 + USSR.l1 + GOPC.l2 + GRA.l2 + RPO.l2 + USSR.l2 + GOPC.l3 + GRA.l3 +
##
##            Estimate Std. Error t value Pr(>|t|)
## GOPC.l1   0.0042092  0.0128159   0.328  0.74282
## GRA.l1    1.1812029  0.0582812  20.267  < 2e-16 ***
## RPO.l1    0.1136618  0.0415712   2.734  0.00664 **
## USSR.l1   0.0604760  0.2131134   0.284  0.77679
## GOPC.l2   0.0125577  0.0126618   0.992  0.32214
## GRA.l2   -0.2114997  0.0894018  -2.366  0.01866 *
## RPO.l2   -0.1690809  0.0719900  -2.349  0.01952 *
## USSR.l2   0.2977945  0.2136781   1.394  0.16450
## GOPC.l3   0.0317651  0.0128806   2.466  0.01425 *
## GRA.l3   -0.0412306  0.0878399  -0.469  0.63915
## RPO.l3    0.1029446  0.0766422   1.343  0.18028
## USSR.l3   0.0388804  0.2156120   0.180  0.85702
## GOPC.l4  -0.0082946  0.0134302  -0.618  0.53733
## GRA.l4    0.0545079  0.0873342   0.624  0.53304
## RPO.l4    0.0124102  0.0768751   0.161  0.87187
## USSR.l4   0.2630997  0.2232442   1.179  0.23957
## GOPC.l5  -0.0185343  0.0130630  -1.419  0.15704
## GRA.l5   -0.0689038  0.0876693  -0.786  0.43255
## RPO.l5   -0.0532697  0.0759906  -0.701  0.48387
## USSR.l5   0.2004175  0.2253021   0.890  0.37446
## GOPC.l6  -0.0069733  0.0133595  -0.522  0.60209
## GRA.l6    0.0342413  0.0878024   0.390  0.69684
## RPO.l6    0.0108841  0.0761378   0.143  0.88643
## USSR.l6  -0.0232977  0.2253509  -0.103  0.91773
## GOPC.l7  -0.0007967  0.0131223  -0.061  0.95163
```

```
## GRA.l7     0.0402783  0.0875857   0.460  0.64596
## RPO.l7     0.0495150  0.0766199   0.646  0.51864
## USSR.l7   -0.4656210  0.2257143  -2.063  0.04003 *
## GOPC.l8    0.0224316  0.0130031   1.725  0.08559 .
## GRA.l8    -0.0716621  0.0875313  -0.819  0.41364
## RPO.l8    -0.0216925  0.0767600  -0.283  0.77769
## USSR.l8    0.0203852  0.2299260   0.089  0.92941
## GOPC.l9    0.0018361  0.0129351   0.142  0.88722
## GRA.l9    -0.0348409  0.0871237  -0.400  0.68953
## RPO.l9    -0.0211776  0.0767949  -0.276  0.78293
## USSR.l9   -0.0403153  0.2291235  -0.176  0.86045
## GOPC.l10 -0.0128459  0.0126958  -1.012  0.31248
## GRA.l10    0.0852814  0.0855656   0.997  0.31976
## RPO.l10   -0.0733980  0.0761818  -0.963  0.33613
## USSR.l10 -0.0467695  0.2273421  -0.206  0.83715
## GOPC.l11   0.0100298  0.0126933   0.790  0.43009
## GRA.l11    0.0775887  0.0843155   0.920  0.35823
## RPO.l11    0.0711597  0.0767085   0.928  0.35436
## USSR.l11   0.0436395  0.2254701   0.194  0.84667
## GOPC.l12   0.0102755  0.0127592   0.805  0.42129
## GRA.l12    0.0225023  0.0834686   0.270  0.78767
## RPO.l12    0.0158919  0.0775205   0.205  0.83772
## USSR.l12 -0.2018535  0.2235037  -0.903  0.36722
## GOPC.l13 -0.0275151  0.0128350  -2.144  0.03290 *
## GRA.l13   -0.1360462  0.0829271  -1.641  0.10199
## RPO.l13    0.0076027  0.0764922   0.099  0.92090
## USSR.l13 -0.0542324  0.2197629  -0.247  0.80526
## GOPC.l14 -0.0068676  0.0128728  -0.533  0.59411
## GRA.l14    0.0206803  0.0835594   0.247  0.80470
## RPO.l14   -0.1325978  0.0752129  -1.763  0.07897 .
## USSR.l14 -0.0654896  0.2439626  -0.268  0.78855
## GOPC.l15   0.0126834  0.0129554   0.979  0.32840
## GRA.l15   -0.1668676  0.0841908  -1.982  0.04843 *
## RPO.l15    0.0552477  0.0766722   0.721  0.47176
## USSR.l15   0.2288820  0.2456918   0.932  0.35234
## GOPC.l16 -0.0237519  0.0131289  -1.809  0.07148 .
## GRA.l16    0.1398646  0.0840069   1.665  0.09702 .
## RPO.l16    0.0907591  0.0773111   1.174  0.24139
## USSR.l16   0.2065186  0.2535976   0.814  0.41612
## GOPC.l17   0.0015981  0.0128380   0.124  0.90102
## GRA.l17    0.0492946  0.0842348   0.585  0.55887
## RPO.l17   -0.0586692  0.0779978  -0.752  0.45256
## USSR.l17   0.2564306  0.2548481   1.006  0.31517
## GOPC.l18 -0.0010453  0.0125567  -0.083  0.93372
## GRA.l18   -0.0272927  0.0843332  -0.324  0.74645
## RPO.l18   -0.0661724  0.0778854  -0.850  0.39625
## USSR.l18   0.1154986  0.2552852   0.452  0.65130
## GOPC.l19 -0.0103724  0.0124981  -0.830  0.40728
## GRA.l19    0.0258444  0.0842743   0.307  0.75932
## RPO.l19    0.1199415  0.0783508   1.531  0.12692
## USSR.l19 -0.1330073  0.2469641  -0.539  0.59060
## GOPC.l20 -0.0067499  0.0123993  -0.544  0.58661
## GRA.l20   -0.0174428  0.0859114  -0.203  0.83925
## RPO.l20   -0.0903018  0.0786586  -1.148  0.25192
```

```
## USSR.l20  0.0124956  0.2471007   0.051  0.95970
## GOPC.l21  0.0061919  0.0122285   0.506  0.61300
## GRA.l21  -0.1203291  0.0872211  -1.380  0.16879
## RPO.l21  -0.0068561  0.0782342  -0.088  0.93023
## USSR.l21  0.0099157  0.2419033   0.041  0.96733
## GOPC.l22 -0.0158568  0.0118984  -1.333  0.18370
## GRA.l22   0.1483100  0.0866785   1.711  0.08816 .
## RPO.l22   0.1418314  0.0782461   1.813  0.07094 .
## USSR.l22  0.2264569  0.2374343   0.954  0.34101
## GOPC.l23 -0.0042543  0.0118937  -0.358  0.72084
## GRA.l23   0.0844620  0.0869584   0.971  0.33222
## RPO.l23  -0.1826180  0.0745134  -2.451  0.01485 *
## USSR.l23  0.0561154  0.2356343   0.238  0.81194
## GOPC.l24 -0.0118408  0.0118187  -1.002  0.31725
## GRA.l24  -0.1276611  0.0570609  -2.237  0.02604 *
## RPO.l24   0.0849188  0.0422406   2.010  0.04533 *
## USSR.l24  0.1707592  0.2304690   0.741  0.45935
## const    -0.2237883  0.2511122  -0.891  0.37358
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 4.132 on 286 degrees of freedom
## Multiple R-Squared: 0.9648,  Adjusted R-squared: 0.9531
## F-statistic: 81.78 on 96 and 286 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation RPO:
## ====================================
## RPO = GOPC.l1 + GRA.l1 + RPO.l1 + USSR.l1 + GOPC.l2 + GRA.l2 + RPO.l2 + USSR.l2 + GOPC.l3 + GRA.l3 +
##
##           Estimate Std. Error t value Pr(>|t|)
## GOPC.l1   0.013168   0.018496   0.712   0.4771
## GRA.l1    0.025183   0.084110   0.299   0.7648
## RPO.l1    1.446905   0.059995  24.117  < 2e-16 ***
## USSR.l1  -0.164671   0.307562  -0.535   0.5928
## GOPC.l2  -0.035818   0.018273  -1.960   0.0510 .
## GRA.l2    0.004776   0.129023   0.037   0.9705
## RPO.l2   -0.614867   0.103895  -5.918 9.29e-09 ***
## USSR.l2   0.794311   0.308377   2.576   0.0105 *
## GOPC.l3   0.014033   0.018589   0.755   0.4509
## GRA.l3   -0.099391   0.126769  -0.784   0.4337
## RPO.l3    0.191247   0.110609   1.729   0.0849 .
## USSR.l3   0.280047   0.311168   0.900   0.3689
## GOPC.l4  -0.021326   0.019382  -1.100   0.2721
## GRA.l4    0.178454   0.126039   1.416   0.1579
## RPO.l4   -0.121700   0.110945  -1.097   0.2736
## USSR.l4   0.321842   0.322182   0.999   0.3187
## GOPC.l5  -0.006124   0.018852  -0.325   0.7455
## GRA.l5   -0.135607   0.126523  -1.072   0.2847
## RPO.l5    0.145922   0.109668   1.331   0.1844
## USSR.l5  -0.283874   0.325152  -0.873   0.3834
## GOPC.l6   0.013809   0.019280   0.716   0.4744
## GRA.l6    0.043401   0.126715   0.343   0.7322
```

```
## RPO.l6   -0.081927   0.109881  -0.746   0.4565
## USSR.l6  -0.003877   0.325223  -0.012   0.9905
## GOPC.l7  -0.006468   0.018938  -0.342   0.7330
## GRA.l7    0.084087   0.126402   0.665   0.5064
## RPO.l7    0.042328   0.110577   0.383   0.7022
## USSR.l7   0.216880   0.325747   0.666   0.5061
## GOPC.l8   0.007206   0.018766   0.384   0.7012
## GRA.l8   -0.001024   0.126324  -0.008   0.9935
## RPO.l8   -0.041952   0.110779  -0.379   0.7052
## USSR.l8  -0.577344   0.331825  -1.740   0.0830 .
## GOPC.l9   0.004317   0.018668   0.231   0.8173
## GRA.l9    0.044031   0.125735   0.350   0.7265
## RPO.l9   -0.046550   0.110829  -0.420   0.6748
## USSR.l9  -0.242536   0.330667  -0.733   0.4639
## GOPC.l10  0.004438   0.018322   0.242   0.8088
## GRA.l10  -0.191926   0.123487  -1.554   0.1212
## RPO.l10   0.150344   0.109944   1.367   0.1726
## USSR.l10 -0.374854   0.328096  -1.143   0.2542
## GOPC.l11  0.002111   0.018319   0.115   0.9084
## GRA.l11   0.047066   0.121683   0.387   0.6992
## RPO.l11  -0.077647   0.110704  -0.701   0.4836
## USSR.l11 -0.006662   0.325395  -0.020   0.9837
## GOPC.l12  0.018877   0.018414   1.025   0.3061
## GRA.l12   0.123805   0.120460   1.028   0.3049
## RPO.l12   0.005891   0.111876   0.053   0.9580
## USSR.l12  0.178940   0.322557   0.555   0.5795
## GOPC.l13 -0.002043   0.018523  -0.110   0.9122
## GRA.l13  -0.163906   0.119679  -1.370   0.1719
## RPO.l13  -0.188998   0.110392  -1.712   0.0880 .
## USSR.l13  0.086661   0.317158   0.273   0.7849
## GOPC.l14  0.015065   0.018578   0.811   0.4181
## GRA.l14   0.008779   0.120592   0.073   0.9420
## RPO.l14   0.256337   0.108546   2.362   0.0189 *
## USSR.l14  0.652423   0.352083   1.853   0.0649 .
## GOPC.l15 -0.023287   0.018697  -1.246   0.2140
## GRA.l15   0.025691   0.121503   0.211   0.8327
## RPO.l15  -0.213459   0.110652  -1.929   0.0547 .
## USSR.l15  0.343139   0.354578   0.968   0.3340
## GOPC.l16 -0.006323   0.018947  -0.334   0.7389
## GRA.l16   0.017935   0.121237   0.148   0.8825
## RPO.l16   0.120134   0.111574   1.077   0.2825
## USSR.l16  0.706772   0.365988   1.931   0.0545 .
## GOPC.l17  0.014386   0.018528   0.776   0.4381
## GRA.l17   0.133044   0.121566   1.094   0.2747
## RPO.l17   0.027161   0.112565   0.241   0.8095
## USSR.l17  0.113747   0.367793   0.309   0.7573
## GOPC.l18  0.008680   0.018122   0.479   0.6323
## GRA.l18  -0.162792   0.121708  -1.338   0.1821
## RPO.l18  -0.044219   0.112403  -0.393   0.6943
## USSR.l18 -0.468301   0.368423  -1.271   0.2047
## GOPC.l19 -0.009808   0.018037  -0.544   0.5870
## GRA.l19  -0.018852   0.121623  -0.155   0.8769
## RPO.l19   0.045769   0.113075   0.405   0.6860
## USSR.l19 -0.033714   0.356414  -0.095   0.9247
```

```
## GOPC.l20 -0.013552   0.017894  -0.757   0.4495
## GRA.l20   0.233269   0.123986   1.881   0.0609 .
## RPO.l20   0.058540   0.113519   0.516   0.6065
## USSR.l20 -0.127005   0.356612  -0.356   0.7220
## GOPC.l21  0.017444   0.017648   0.988   0.3238
## GRA.l21  -0.217676   0.125876  -1.729   0.0848 .
## RPO.l21  -0.122334   0.112906  -1.083   0.2795
## USSR.l21 -0.461896   0.349111  -1.323   0.1869
## GOPC.l22  0.014065   0.017172   0.819   0.4134
## GRA.l22   0.163648   0.125093   1.308   0.1919
## RPO.l22   0.130040   0.112924   1.152   0.2505
## USSR.l22 -0.246169   0.342661  -0.718   0.4731
## GOPC.l23  0.011246   0.017165   0.655   0.5129
## GRA.l23  -0.074353   0.125497  -0.592   0.5540
## RPO.l23  -0.173853   0.107537  -1.617   0.1070
## USSR.l23 -0.380674   0.340064  -1.119   0.2639
## GOPC.l24 -0.007196   0.017057  -0.422   0.6734
## GRA.l24  -0.015966   0.082349  -0.194   0.8464
## RPO.l24   0.093254   0.060961   1.530   0.1272
## USSR.l24  0.265824   0.332609   0.799   0.4248
## const     0.046228   0.362401   0.128   0.8986
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 5.963 on 286 degrees of freedom
## Multiple R-Squared: 0.9872,  Adjusted R-squared: 0.9829
## F-statistic: 229.7 on 96 and 286 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation USSR:
## =====================================
## USSR = GOPC.l1 + GRA.l1 + RPO.l1 + USSR.l1 + GOPC.l2 + GRA.l2 + RPO.l2 + USSR.l2 + GOPC.l3 + GRA.l3 +
##
##             Estimate Std. Error t value Pr(>|t|)
## GOPC.l1   0.0029064  0.0034096   0.852 0.394708
## GRA.l1    0.0140124  0.0155055   0.904 0.366913
## RPO.l1   -0.0135327  0.0110599  -1.224 0.222116
## USSR.l1  -0.0581940  0.0566981  -1.026 0.305578
## GOPC.l2   0.0028392  0.0033686   0.843 0.400029
## GRA.l2   -0.0025951  0.0237851  -0.109 0.913195
## RPO.l2    0.0101092  0.0191527   0.528 0.598034
## USSR.l2  -0.1416585  0.0568484  -2.492 0.013274 *
## GOPC.l3   0.0040800  0.0034269   1.191 0.234801
## GRA.l3   -0.0185202  0.0233695  -0.792 0.428730
## RPO.l3   -0.0053587  0.0203904  -0.263 0.792890
## USSR.l3   0.2581135  0.0573629   4.500 9.91e-06 ***
## GOPC.l4  -0.0010391  0.0035731  -0.291 0.771411
## GRA.l4    0.0593446  0.0232350   2.554 0.011165 *
## RPO.l4    0.0086409  0.0204524   0.422 0.672986
## USSR.l4   0.0500138  0.0593934   0.842 0.400449
## GOPC.l5   0.0052622  0.0034754   1.514 0.131097
## GRA.l5   -0.0521731  0.0233241  -2.237 0.026065 *
## RPO.l5   -0.0198651  0.0202171  -0.983 0.326639
```

```
## USSR.l5   0.0796988  0.0599409   1.330 0.184702
## GOPC.l6   0.0015711  0.0035542   0.442 0.658792
## GRA.l6    0.0023891  0.0233595   0.102 0.918610
## RPO.l6    0.0372354  0.0202562   1.838 0.067067 .
## USSR.l6   0.1814172  0.0599539   3.026 0.002704 **
## GOPC.l7   0.0002589  0.0034912   0.074 0.940932
## GRA.l7   -0.0122479  0.0233019  -0.526 0.599562
## RPO.l7   -0.0368360  0.0203845  -1.807 0.071803 .
## USSR.l7   0.0196915  0.0600506   0.328 0.743216
## GOPC.l8   0.0048125  0.0034594   1.391 0.165273
## GRA.l8    0.0437877  0.0232874   1.880 0.061081 .
## RPO.l8    0.0291006  0.0204217   1.425 0.155253
## USSR.l8   0.1149466  0.0611711   1.879 0.061248 .
## GOPC.l9   0.0039472  0.0034413   1.147 0.252344
## GRA.l9   -0.0254723  0.0231790  -1.099 0.272719
## RPO.l9   -0.0336992  0.0204310  -1.649 0.100162
## USSR.l9   0.0336916  0.0609576   0.553 0.580897
## GOPC.l10  0.0016361  0.0033777   0.484 0.628489
## GRA.l10   0.0148370  0.0227645   0.652 0.515077
## RPO.l10   0.0158265  0.0202679   0.781 0.435529
## USSR.l10 -0.0321164  0.0604836  -0.531 0.595836
## GOPC.l11  0.0039906  0.0033770   1.182 0.238306
## GRA.l11   0.0109818  0.0224319   0.490 0.624818
## RPO.l11   0.0104069  0.0204081   0.510 0.610487
## USSR.l11  0.0187873  0.0599856   0.313 0.754359
## GOPC.l12  0.0004390  0.0033945   0.129 0.897202
## GRA.l12  -0.0452638  0.0222065  -2.038 0.042439 *
## RPO.l12  -0.0356935  0.0206241  -1.731 0.084589 .
## USSR.l12 -0.5208241  0.0594625  -8.759  < 2e-16 ***
## GOPC.l13  0.0027315  0.0034147   0.800 0.424422
## GRA.l13   0.0548078  0.0220625   2.484 0.013556 *
## RPO.l13   0.0614710  0.0203505   3.021 0.002751 **
## USSR.l13 -0.0529849  0.0584672  -0.906 0.365576
## GOPC.l14  0.0059040  0.0034248   1.724 0.085805 .
## GRA.l14  -0.0411000  0.0222307  -1.849 0.065520 .
## RPO.l14  -0.0759633  0.0200102  -3.796 0.000179 ***
## USSR.l14 -0.0423042  0.0649055  -0.652 0.515066
## GOPC.l15  0.0095867  0.0034467   2.781 0.005773 **
## GRA.l15   0.0035112  0.0223987   0.157 0.875545
## RPO.l15   0.0813139  0.0203984   3.986 8.53e-05 ***
## USSR.l15  0.2292846  0.0653655   3.508 0.000525 ***
## GOPC.l16 -0.0006768  0.0034929  -0.194 0.846496
## GRA.l16  -0.0076873  0.0223498  -0.344 0.731133
## RPO.l16  -0.0672092  0.0205684  -3.268 0.001217 **
## USSR.l16  0.1216073  0.0674688   1.802 0.072532 .
## GOPC.l17  0.0009340  0.0034155   0.273 0.784707
## GRA.l17   0.0317834  0.0224104   1.418 0.157208
## RPO.l17   0.0407566  0.0207511   1.964 0.050490 .
## USSR.l17  0.1183042  0.0678015   1.745 0.082083 .
## GOPC.l18  0.0049041  0.0033407   1.468 0.143201
## GRA.l18  -0.0118184  0.0224366  -0.527 0.598778
## RPO.l18  -0.0147280  0.0207212  -0.711 0.477806
## USSR.l18  0.1852271  0.0679178   2.727 0.006782 **
## GOPC.l19 -0.0026958  0.0033251  -0.811 0.418193
```

```
## GRA.l19   0.0052265  0.0224209   0.233 0.815845
## RPO.l19   0.0025540  0.0208450   0.123 0.902569
## USSR.l19  0.0110614  0.0657040   0.168 0.866426
## GOPC.l20  0.0012596  0.0032988   0.382 0.702873
## GRA.l20  -0.0046347  0.0228565  -0.203 0.839455
## RPO.l20  -0.0090478  0.0209269  -0.432 0.665810
## USSR.l20  0.0314814  0.0657404   0.479 0.632394
## GOPC.l21  0.0038619  0.0032534   1.187 0.236193
## GRA.l21  -0.0148647  0.0232049  -0.641 0.522307
## RPO.l21  -0.0006461  0.0208140  -0.031 0.975258
## USSR.l21  0.0172383  0.0643576   0.268 0.789007
## GOPC.l22  0.0008933  0.0031655   0.282 0.777995
## GRA.l22   0.0013801  0.0230605   0.060 0.952320
## RPO.l22   0.0353943  0.0208171   1.700 0.090172 .
## USSR.l22 -0.1132701  0.0631686  -1.793 0.074007 .
## GOPC.l23 -0.0044274  0.0031643  -1.399 0.162849
## GRA.l23  -0.0029853  0.0231350  -0.129 0.897418
## RPO.l23  -0.0285322  0.0198241  -1.439 0.151167
## USSR.l23 -0.0127512  0.0626898  -0.203 0.838965
## GOPC.l24  0.0045899  0.0031443   1.460 0.145457
## GRA.l24   0.0138337  0.0151809   0.911 0.362924
## RPO.l24   0.0084847  0.0112380   0.755 0.450870
## USSR.l24 -0.3295174  0.0613155  -5.374 1.60e-07 ***
## const     0.1363407  0.0668076   2.041 0.042189 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 1.099 on 286 degrees of freedom
## Multiple R-Squared: 0.4757,  Adjusted R-squared: 0.2997
## F-statistic: 2.703 on 96 and 286 DF,  p-value: 8.175e-11
##
##
##
## Covariance matrix of residuals:
##          GOPC     GRA      RPO    USSR
## GOPC 357.6018 -0.4482 -11.6081 -0.9295
## GRA   -0.4482 17.0705   1.8063  0.1848
## RPO  -11.6081  1.8063  35.5542 -0.7962
## USSR  -0.9295  0.1848  -0.7962  1.2083
##
## Correlation matrix of residuals:
##           GOPC       GRA      RPO     USSR
## GOPC  1.000000 -0.005737 -0.10295 -0.04472
## GRA  -0.005737  1.000000  0.07332  0.04070
## RPO  -0.102947  0.073319  1.00000 -0.12148
## USSR -0.044716  0.040701 -0.12148  1.00000
```

```
# Summary of model 2 using the SP500 stock returns
summary(var_rep)
```

```
##
## VAR Estimation Results:
## =========================
```

```
## Endogenous variables: GOPC, GRA, RPO, adj_stock_returns
## Deterministic variables: const
## Sample size: 383
## Log Likelihood: -4746.627
## Roots of the characteristic polynomial:
## 0.9857 0.9765 0.9765 0.9739 0.9739 0.9705 0.9705 0.965 0.965 0.9638 0.9638 0.9632 0.9632 0.9618 0.96
## Call:
## VAR(y = df_2, p = 24, type = "const")
##
##
## Estimation results for equation GOPC:
## =====================================
## GOPC = GOPC.l1 + GRA.l1 + RPO.l1 + adj_stock_returns.l1 + GOPC.l2 + GRA.l2 + RPO.l2 + adj_stock_retu
##
##                        Estimate Std. Error t value Pr(>|t|)
## GOPC.l1               -0.130833   0.059202  -2.210 0.027901 *
## GRA.l1                -0.164136   0.261178  -0.628 0.530213
## RPO.l1                -0.360287   0.184469  -1.953 0.051783 .
## adj_stock_returns.l1  -0.238034   0.323366  -0.736 0.462265
## GOPC.l2               -0.084580   0.059002  -1.434 0.152801
## GRA.l2                 0.440406   0.403890   1.090 0.276450
## RPO.l2                 0.755874   0.322398   2.345 0.019735 *
## adj_stock_returns.l2   0.836274   0.330647   2.529 0.011970 *
## GOPC.l3               -0.211987   0.059560  -3.559 0.000435 ***
## GRA.l3                -0.518081   0.396211  -1.308 0.192063
## RPO.l3                -0.420766   0.347228  -1.212 0.226595
## adj_stock_returns.l3  -0.190631   0.329261  -0.579 0.563069
## GOPC.l4               -0.156281   0.061902  -2.525 0.012122 *
## GRA.l4                 0.200560   0.396312   0.506 0.613201
## RPO.l4                 0.025712   0.346710   0.074 0.940935
## adj_stock_returns.l4  -0.475750   0.328244  -1.449 0.148328
## GOPC.l5               -0.194776   0.060929  -3.197 0.001546 **
## GRA.l5                 0.156280   0.396761   0.394 0.693957
## RPO.l5                -0.329380   0.340477  -0.967 0.334158
## adj_stock_returns.l5  -0.094935   0.321118  -0.296 0.767720
## GOPC.l6               -0.012130   0.061414  -0.198 0.843565
## GRA.l6                -0.257942   0.395615  -0.652 0.514923
## RPO.l6                 0.864881   0.342519   2.525 0.012109 *
## adj_stock_returns.l6   1.106103   0.321190   3.444 0.000660 ***
## GOPC.l7               -0.091988   0.060021  -1.533 0.126481
## GRA.l7                 0.701983   0.395156   1.776 0.076718 .
## RPO.l7                -0.477360   0.344799  -1.384 0.167297
## adj_stock_returns.l7   0.364942   0.325489   1.121 0.263138
## GOPC.l8               -0.025386   0.059772  -0.425 0.671364
## GRA.l8                -0.536310   0.394530  -1.359 0.175102
## RPO.l8                -0.176865   0.341664  -0.518 0.605098
## adj_stock_returns.l8  -0.038928   0.324505  -0.120 0.904599
## GOPC.l9                0.132523   0.059817   2.215 0.027515 *
## GRA.l9                -0.133424   0.389563  -0.342 0.732229
## RPO.l9                 0.099478   0.340092   0.293 0.770114
## adj_stock_returns.l9   0.362469   0.321837   1.126 0.261003
## GOPC.l10               0.007628   0.059160   0.129 0.897501
## GRA.l10               -0.018140   0.380201  -0.048 0.961979
## RPO.l10                0.263181   0.338036   0.779 0.436883
```

```
## adj_stock_returns.l10 -0.252015   0.320538  -0.786 0.432388
## GOPC.l11              -0.093177   0.058925  -1.581 0.114920
## GRA.l11                0.301504   0.374902   0.804 0.421937
## RPO.l11               -0.404678   0.340876  -1.187 0.236146
## adj_stock_returns.l11 -0.114407   0.319520  -0.358 0.720565
## GOPC.l12               0.249493   0.059138   4.219  3.3e-05 ***
## GRA.l12               -0.153887   0.371643  -0.414 0.679132
## RPO.l12                0.267401   0.344552   0.776 0.438341
## adj_stock_returns.l12 -0.461782   0.319985  -1.443 0.150077
## GOPC.l13               0.054893   0.060125   0.913 0.362028
## GRA.l13                0.187968   0.368568   0.510 0.610449
## RPO.l13                0.028224   0.340847   0.083 0.934063
## adj_stock_returns.l13  0.499797   0.323679   1.544 0.123666
## GOPC.l14              -0.075380   0.060333  -1.249 0.212542
## GRA.l14               -0.897478   0.369723  -2.427 0.015824 *
## RPO.l14               -0.279476   0.330110  -0.847 0.397917
## adj_stock_returns.l14 -0.418299   0.321690  -1.300 0.194540
## GOPC.l15               0.111743   0.060505   1.847 0.065803 .
## GRA.l15                0.456163   0.369773   1.234 0.218353
## RPO.l15               -0.011225   0.330283  -0.034 0.972913
## adj_stock_returns.l15 -0.124614   0.322395  -0.387 0.699395
## GOPC.l16               0.001001   0.060708   0.016 0.986857
## GRA.l16                0.129118   0.368419   0.350 0.726247
## RPO.l16                0.331237   0.328795   1.007 0.314581
## adj_stock_returns.l16 -0.078641   0.325692  -0.241 0.809372
## GOPC.l17               0.061138   0.059445   1.028 0.304596
## GRA.l17               -0.415380   0.370337  -1.122 0.262963
## RPO.l17               -0.348619   0.328084  -1.063 0.288865
## adj_stock_returns.l17  0.552032   0.325746   1.695 0.091227 .
## GOPC.l18              -0.012726   0.058417  -0.218 0.827705
## GRA.l18                0.784598   0.370546   2.117 0.035088 *
## RPO.l18                0.540420   0.327250   1.651 0.099754 .
## adj_stock_returns.l18 -0.527377   0.327828  -1.609 0.108785
## GOPC.l19              -0.008046   0.056253  -0.143 0.886367
## GRA.l19               -1.157037   0.373180  -3.100 0.002125 **
## RPO.l19               -0.561427   0.330694  -1.698 0.090648 .
## adj_stock_returns.l19 -0.198829   0.329013  -0.604 0.546111
## GOPC.l20               0.017873   0.055772   0.320 0.748853
## GRA.l20                0.717550   0.380062   1.888 0.060040 .
## RPO.l20                0.258955   0.336348   0.770 0.441994
## adj_stock_returns.l20 -0.033587   0.326620  -0.103 0.918169
## GOPC.l21              -0.063920   0.055487  -1.152 0.250287
## GRA.l21                0.504829   0.387861   1.302 0.194110
## RPO.l21               -0.147289   0.336776  -0.437 0.662188
## adj_stock_returns.l21  0.581906   0.323352   1.800 0.072977 .
## GOPC.l22              -0.056641   0.054309  -1.043 0.297851
## GRA.l22               -0.934453   0.385410  -2.425 0.015946 *
## RPO.l22                0.290046   0.337767   0.859 0.391217
## adj_stock_returns.l22 -0.436784   0.323680  -1.349 0.178266
## GOPC.l23               0.005719   0.054385   0.105 0.916330
## GRA.l23                0.469900   0.386359   1.216 0.224901
## RPO.l23               -0.323945   0.323601  -1.001 0.317643
## adj_stock_returns.l23 -0.148825   0.322760  -0.461 0.645075
## GOPC.l24              -0.006142   0.054007  -0.114 0.909528
```

```
## GRA.l24                0.056117   0.250634   0.224 0.822994
## RPO.l24                0.082425   0.183666   0.449 0.653933
## adj_stock_returns.l24 -0.381630   0.308811  -1.236 0.217546
## const                  1.121480   1.121446   1.000 0.318141
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 18.45 on 286 degrees of freedom
## Multiple R-Squared: 0.4005,  Adjusted R-squared: 0.1993
## F-statistic: 1.991 on 96 and 286 DF,  p-value: 6.499e-06
##
##
## Estimation results for equation GRA:
## ====================================
## GRA = GOPC.l1 + GRA.l1 + RPO.l1 + adj_stock_returns.l1 + GOPC.l2 + GRA.l2 + RPO.l2 + adj_stock_returr
##
##                         Estimate Std. Error t value Pr(>|t|)
## GOPC.l1               -6.171e-04  1.318e-02  -0.047  0.96269
## GRA.l1                 1.193e+00  5.814e-02  20.514  < 2e-16 ***
## RPO.l1                 1.161e-01  4.107e-02   2.828  0.00502 **
## adj_stock_returns.l1   1.311e-02  7.199e-02   0.182  0.85565
## GOPC.l2                2.348e-02  1.313e-02   1.788  0.07486 .
## GRA.l2                -1.986e-01  8.991e-02  -2.209  0.02799 *
## RPO.l2                -1.716e-01  7.177e-02  -2.391  0.01743 *
## adj_stock_returns.l2  -5.375e-02  7.361e-02  -0.730  0.46584
## GOPC.l3                3.437e-02  1.326e-02   2.592  0.01003 *
## GRA.l3                -7.236e-02  8.820e-02  -0.820  0.41269
## RPO.l3                 8.601e-02  7.730e-02   1.113  0.26677
## adj_stock_returns.l3   9.699e-03  7.330e-02   0.132  0.89482
## GOPC.l4               -9.245e-03  1.378e-02  -0.671  0.50281
## GRA.l4                 7.122e-02  8.822e-02   0.807  0.42022
## RPO.l4                 3.841e-02  7.718e-02   0.498  0.61913
## adj_stock_returns.l4   1.301e-02  7.307e-02   0.178  0.85880
## GOPC.l5               -1.844e-02  1.356e-02  -1.360  0.17504
## GRA.l5                -5.523e-02  8.832e-02  -0.625  0.53226
## RPO.l5                -8.898e-02  7.579e-02  -1.174  0.24140
## adj_stock_returns.l5  -7.757e-02  7.149e-02  -1.085  0.27880
## GOPC.l6                4.685e-03  1.367e-02   0.343  0.73207
## GRA.l6                 3.927e-02  8.807e-02   0.446  0.65600
## RPO.l6                 3.818e-02  7.625e-02   0.501  0.61695
## adj_stock_returns.l6   3.508e-02  7.150e-02   0.491  0.62403
## GOPC.l7                1.120e-02  1.336e-02   0.838  0.40260
## GRA.l7                 1.941e-02  8.797e-02   0.221  0.82551
## RPO.l7                 7.025e-02  7.676e-02   0.915  0.36084
## adj_stock_returns.l7   1.259e-01  7.246e-02   1.737  0.08341 .
## GOPC.l8                2.491e-02  1.331e-02   1.872  0.06219 .
## GRA.l8                -7.335e-02  8.783e-02  -0.835  0.40431
## RPO.l8                -3.682e-02  7.606e-02  -0.484  0.62866
## adj_stock_returns.l8  -9.356e-02  7.224e-02  -1.295  0.19631
## GOPC.l9                1.126e-02  1.332e-02   0.846  0.39828
## GRA.l9                -2.086e-02  8.672e-02  -0.241  0.81008
## RPO.l9                -4.141e-02  7.571e-02  -0.547  0.58483
## adj_stock_returns.l9  -5.869e-02  7.165e-02  -0.819  0.41335
```

```
## GOPC.l10              1.918e-03  1.317e-02   0.146  0.88431
## GRA.l10               8.670e-02  8.464e-02   1.024  0.30652
## RPO.l10              -5.627e-02  7.525e-02  -0.748  0.45522
## adj_stock_returns.l10  6.942e-02  7.136e-02   0.973  0.33147
## GOPC.l11              1.668e-02  1.312e-02   1.271  0.20465
## GRA.l11               5.554e-02  8.346e-02   0.666  0.50625
## RPO.l11               7.259e-02  7.588e-02   0.957  0.33959
## adj_stock_returns.l11 -6.441e-02  7.113e-02  -0.906  0.36594
## GOPC.l12              1.864e-02  1.316e-02   1.416  0.15786
## GRA.l12               4.821e-02  8.273e-02   0.583  0.56051
## RPO.l12              -1.313e-06  7.670e-02   0.000  0.99999
## adj_stock_returns.l12 -6.351e-02  7.123e-02  -0.892  0.37335
## GOPC.l13             -1.091e-02  1.338e-02  -0.815  0.41567
## GRA.l13              -1.327e-01  8.205e-02  -1.617  0.10700
## RPO.l13               5.784e-03  7.588e-02   0.076  0.93929
## adj_stock_returns.l13  4.300e-02  7.206e-02   0.597  0.55109
## GOPC.l14             -3.094e-03  1.343e-02  -0.230  0.81797
## GRA.l14              -1.951e-02  8.231e-02  -0.237  0.81282
## RPO.l14              -1.168e-01  7.349e-02  -1.590  0.11298
## adj_stock_returns.l14 -3.821e-02  7.161e-02  -0.534  0.59406
## GOPC.l15              1.131e-02  1.347e-02   0.840  0.40158
## GRA.l15              -1.338e-01  8.232e-02  -1.625  0.10525
## RPO.l15               4.061e-02  7.353e-02   0.552  0.58118
## adj_stock_returns.l15 -1.499e-01  7.177e-02  -2.088  0.03766 *
## GOPC.l16             -1.635e-02  1.351e-02  -1.210  0.22743
## GRA.l16               1.442e-01  8.201e-02   1.758  0.07975 .
## RPO.l16               7.934e-02  7.319e-02   1.084  0.27929
## adj_stock_returns.l16  4.936e-02  7.250e-02   0.681  0.49659
## GOPC.l17              1.051e-02  1.323e-02   0.794  0.42785
## GRA.l17               1.263e-02  8.244e-02   0.153  0.87835
## RPO.l17              -1.004e-02  7.304e-02  -0.137  0.89079
## adj_stock_returns.l17 -8.495e-03  7.252e-02  -0.117  0.90683
## GOPC.l18             -4.314e-03  1.300e-02  -0.332  0.74031
## GRA.l18              -1.108e-02  8.249e-02  -0.134  0.89327
## RPO.l18              -1.221e-01  7.285e-02  -1.676  0.09478 .
## adj_stock_returns.l18 -1.173e-01  7.298e-02  -1.607  0.10912
## GOPC.l19             -1.188e-02  1.252e-02  -0.949  0.34354
## GRA.l19               6.831e-02  8.307e-02   0.822  0.41160
## RPO.l19               1.475e-01  7.362e-02   2.004  0.04598 *
## adj_stock_returns.l19 -3.174e-02  7.324e-02  -0.433  0.66512
## GOPC.l20             -1.905e-03  1.242e-02  -0.153  0.87815
## GRA.l20              -6.581e-02  8.461e-02  -0.778  0.43731
## RPO.l20              -1.041e-01  7.488e-02  -1.390  0.16562
## adj_stock_returns.l20  2.508e-02  7.271e-02   0.345  0.73035
## GOPC.l21              2.292e-03  1.235e-02   0.186  0.85290
## GRA.l21              -1.010e-01  8.634e-02  -1.170  0.24296
## RPO.l21               3.692e-02  7.497e-02   0.492  0.62282
## adj_stock_returns.l21  4.044e-03  7.198e-02   0.056  0.95524
## GOPC.l22             -2.146e-02  1.209e-02  -1.775  0.07701 .
## GRA.l22               1.630e-01  8.580e-02   1.900  0.05841 .
## RPO.l22               8.646e-02  7.519e-02   1.150  0.25115
## adj_stock_returns.l22  6.609e-02  7.206e-02   0.917  0.35980
## GOPC.l23              5.639e-04  1.211e-02   0.047  0.96289
## GRA.l23               7.001e-02  8.601e-02   0.814  0.41631
```

```
## RPO.l23               -1.454e-01  7.204e-02  -2.018  0.04453 *
## adj_stock_returns.l23 -7.855e-02  7.185e-02  -1.093  0.27520
## GOPC.l24              -8.006e-03  1.202e-02  -0.666  0.50602
## GRA.l24               -1.283e-01  5.579e-02  -2.299  0.02224 *
## RPO.l24                7.633e-02  4.089e-02   1.867  0.06294 .
## adj_stock_returns.l24  8.820e-02  6.875e-02   1.283  0.20054
## const                  2.528e-02  2.496e-01   0.101  0.91941
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 4.108 on 286 degrees of freedom
## Multiple R-Squared: 0.9653,  Adjusted R-squared: 0.9536
## F-statistic: 82.77 on 96 and 286 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation RPO:
## ====================================
## RPO = GOPC.l1 + GRA.l1 + RPO.l1 + adj_stock_returns.l1 + GOPC.l2 + GRA.l2 + RPO.l2 + adj_stock_return
##
##                         Estimate Std. Error t value Pr(>|t|)
## GOPC.l1                1.090e-02  1.902e-02   0.573  0.56716
## GRA.l1                 5.409e-02  8.390e-02   0.645  0.51970
## RPO.l1                 1.464e+00  5.926e-02  24.704  < 2e-16 ***
## adj_stock_returns.l1  -4.745e-02  1.039e-01  -0.457  0.64816
## GOPC.l2               -4.185e-02  1.895e-02  -2.208  0.02804 *
## GRA.l2                -7.869e-04  1.298e-01  -0.006  0.99517
## RPO.l2                -6.628e-01  1.036e-01  -6.399 6.38e-10 ***
## adj_stock_returns.l2   1.713e-01  1.062e-01   1.612  0.10798
## GOPC.l3                3.804e-03  1.913e-02   0.199  0.84254
## GRA.l3                -1.829e-01  1.273e-01  -1.437  0.15186
## RPO.l3                 2.405e-01  1.115e-01   2.156  0.03193 *
## adj_stock_returns.l3  -1.039e-01  1.058e-01  -0.983  0.32666
## GOPC.l4               -1.806e-02  1.989e-02  -0.908  0.36459
## GRA.l4                 2.371e-01  1.273e-01   1.862  0.06358 .
## RPO.l4                -1.272e-01  1.114e-01  -1.142  0.25453
## adj_stock_returns.l4   1.257e-01  1.054e-01   1.192  0.23425
## GOPC.l5                1.776e-04  1.957e-02   0.009  0.99277
## GRA.l5                -1.125e-01  1.275e-01  -0.882  0.37836
## RPO.l5                 9.740e-02  1.094e-01   0.890  0.37398
## adj_stock_returns.l5  -1.496e-01  1.032e-01  -1.451  0.14799
## GOPC.l6                1.057e-02  1.973e-02   0.536  0.59241
## GRA.l6                -1.671e-02  1.271e-01  -0.131  0.89551
## RPO.l6                -5.021e-02  1.100e-01  -0.456  0.64848
## adj_stock_returns.l6  -3.343e-02  1.032e-01  -0.324  0.74617
## GOPC.l7                2.929e-03  1.928e-02   0.152  0.87935
## GRA.l7                 1.923e-01  1.269e-01   1.515  0.13095
## RPO.l7                -9.967e-03  1.108e-01  -0.090  0.92837
## adj_stock_returns.l7   4.102e-02  1.046e-01   0.392  0.69514
## GOPC.l8                1.718e-02  1.920e-02   0.895  0.37165
## GRA.l8                -7.987e-02  1.267e-01  -0.630  0.52907
## RPO.l8                 6.716e-02  1.098e-01   0.612  0.54111
## adj_stock_returns.l8  -2.905e-02  1.042e-01  -0.279  0.78068
## GOPC.l9                6.544e-03  1.922e-02   0.341  0.73368
```

```
## GRA.l9                 1.234e-02  1.251e-01   0.099  0.92154
## RPO.l9                -1.216e-01  1.093e-01  -1.113  0.26683
## adj_stock_returns.l9   1.649e-01  1.034e-01   1.595  0.11187
## GOPC.l10               1.574e-02  1.901e-02   0.828  0.40818
## GRA.l10               -1.435e-01  1.221e-01  -1.175  0.24099
## RPO.l10                2.263e-01  1.086e-01   2.084  0.03808 *
## adj_stock_returns.l10 -2.678e-02  1.030e-01  -0.260  0.79499
## GOPC.l11               1.029e-02  1.893e-02   0.544  0.58718
## GRA.l11                9.379e-02  1.204e-01   0.779  0.43677
## RPO.l11               -1.499e-01  1.095e-01  -1.369  0.17210
## adj_stock_returns.l11  9.737e-02  1.026e-01   0.949  0.34361
## GOPC.l12               2.346e-02  1.900e-02   1.235  0.21799
## GRA.l12                3.616e-02  1.194e-01   0.303  0.76221
## RPO.l12               -7.046e-04  1.107e-01  -0.006  0.99493
## adj_stock_returns.l12 -2.220e-01  1.028e-01  -2.159  0.03166 *
## GOPC.l13               6.296e-05  1.932e-02   0.003  0.99740
## GRA.l13               -1.412e-01  1.184e-01  -1.193  0.23395
## RPO.l13               -1.242e-01  1.095e-01  -1.134  0.25770
## adj_stock_returns.l13 -4.567e-04  1.040e-01  -0.004  0.99650
## GOPC.l14               1.573e-02  1.938e-02   0.811  0.41779
## GRA.l14                4.786e-02  1.188e-01   0.403  0.68729
## RPO.l14                1.449e-01  1.060e-01   1.366  0.17301
## adj_stock_returns.l14  2.870e-02  1.033e-01   0.278  0.78140
## GOPC.l15              -2.426e-03  1.944e-02  -0.125  0.90075
## GRA.l15                2.672e-02  1.188e-01   0.225  0.82221
## RPO.l15               -8.527e-02  1.061e-01  -0.804  0.42230
## adj_stock_returns.l15 -1.726e-01  1.036e-01  -1.667  0.09670 .
## GOPC.l16              -4.859e-03  1.950e-02  -0.249  0.80344
## GRA.l16                2.306e-02  1.184e-01   0.195  0.84568
## RPO.l16               -3.227e-02  1.056e-01  -0.305  0.76021
## adj_stock_returns.l16  1.038e-01  1.046e-01   0.992  0.32211
## GOPC.l17               2.176e-02  1.910e-02   1.139  0.25547
## GRA.l17                1.013e-01  1.190e-01   0.851  0.39545
## RPO.l17                1.694e-01  1.054e-01   1.608  0.10901
## adj_stock_returns.l17 -1.604e-01  1.046e-01  -1.533  0.12645
## GOPC.l18               1.885e-02  1.877e-02   1.004  0.31601
## GRA.l18               -1.992e-01  1.190e-01  -1.674  0.09531 .
## RPO.l18               -1.414e-01  1.051e-01  -1.345  0.17981
## adj_stock_returns.l18 -5.571e-03  1.053e-01  -0.053  0.95785
## GOPC.l19              -1.425e-02  1.807e-02  -0.788  0.43111
## GRA.l19                2.365e-02  1.199e-01   0.197  0.84377
## RPO.l19                1.260e-01  1.062e-01   1.186  0.23648
## adj_stock_returns.l19  9.849e-02  1.057e-01   0.932  0.35222
## GOPC.l20              -1.150e-02  1.792e-02  -0.642  0.52156
## GRA.l20                2.595e-01  1.221e-01   2.126  0.03439 *
## RPO.l20                9.726e-03  1.081e-01   0.090  0.92834
## adj_stock_returns.l20  1.418e-02  1.049e-01   0.135  0.89256
## GOPC.l21               2.416e-02  1.783e-02   1.355  0.17635
## GRA.l21               -2.331e-01  1.246e-01  -1.870  0.06244 .
## RPO.l21               -1.210e-01  1.082e-01  -1.118  0.26439
## adj_stock_returns.l21 -7.436e-02  1.039e-01  -0.716  0.47466
## GOPC.l22               1.118e-02  1.745e-02   0.641  0.52222
## GRA.l22                1.342e-01  1.238e-01   1.084  0.27925
## RPO.l22                1.663e-01  1.085e-01   1.533  0.12650
```

```
## adj_stock_returns.l22 -2.406e-02  1.040e-01  -0.231  0.81716
## GOPC.l23              6.572e-03  1.747e-02   0.376  0.70706
## GRA.l23              -5.873e-02  1.241e-01  -0.473  0.63648
## RPO.l23              -2.626e-01  1.040e-01  -2.526  0.01209 *
## adj_stock_returns.l23  5.080e-03  1.037e-01   0.049  0.96096
## GOPC.l24             -1.466e-02  1.735e-02  -0.845  0.39869
## GRA.l24              -1.669e-02  8.052e-02  -0.207  0.83595
## RPO.l24               1.627e-01  5.900e-02   2.758  0.00619 **
## adj_stock_returns.l24 -1.391e-01  9.921e-02  -1.402  0.16209
## const                 2.561e-01  3.603e-01   0.711  0.47777
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 5.928 on 286 degrees of freedom
## Multiple R-Squared: 0.9873,  Adjusted R-squared: 0.9831
## F-statistic: 232.4 on 96 and 286 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation adj_stock_returns:
## ==================================================
## adj_stock_returns = GOPC.l1 + GRA.l1 + RPO.l1 + adj_stock_returns.l1 + GOPC.l2 + GRA.l2 + RPO.l2 + ad
##
##                        Estimate Std. Error t value Pr(>|t|)
## GOPC.l1              -0.0060655  0.0108421  -0.559  0.57630
## GRA.l1                0.0546690  0.0478318   1.143  0.25402
## RPO.l1               -0.0964276  0.0337835  -2.854  0.00463 **
## adj_stock_returns.l1   0.2618979  0.0592209   4.422 1.39e-05 ***
## GOPC.l2               0.0139176  0.0108055   1.288  0.19878
## GRA.l2               -0.0299465  0.0739680  -0.405  0.68588
## RPO.l2                0.1202471  0.0590436   2.037  0.04261 *
## adj_stock_returns.l2  -0.0619621  0.0605543  -1.023  0.30705
## GOPC.l3              -0.0213287  0.0109078  -1.955  0.05151 .
## GRA.l3               -0.0518465  0.0725618  -0.715  0.47549
## RPO.l3               -0.0141059  0.0635911  -0.222  0.82461
## adj_stock_returns.l3   0.0349970  0.0603006   0.580  0.56212
## GOPC.l4              -0.0054486  0.0113366  -0.481  0.63116
## GRA.l4                0.0633875  0.0725802   0.873  0.38321
## RPO.l4               -0.0553021  0.0634962  -0.871  0.38451
## adj_stock_returns.l4  -0.0329524  0.0601143  -0.548  0.58401
## GOPC.l5               0.0009181  0.0111585   0.082  0.93448
## GRA.l5               -0.0934883  0.0726625  -1.287  0.19927
## RPO.l5                0.0719176  0.0623546   1.153  0.24972
## adj_stock_returns.l5   0.1091194  0.0588092   1.855  0.06456 .
## GOPC.l6              -0.0061828  0.0112473  -0.550  0.58294
## GRA.l6                0.1115057  0.0724525   1.539  0.12491
## RPO.l6               -0.0110588  0.0627286  -0.176  0.86019
## adj_stock_returns.l6  -0.0460015  0.0588224  -0.782  0.43484
## GOPC.l7              -0.0035204  0.0109922  -0.320  0.74900
## GRA.l7               -0.0337570  0.0723684  -0.466  0.64124
## RPO.l7               -0.0310062  0.0631462  -0.491  0.62379
## adj_stock_returns.l7  -0.0500895  0.0596097  -0.840  0.40145
## GOPC.l8               0.0084000  0.0109467   0.767  0.44350
## GRA.l8               -0.0144665  0.0722539  -0.200  0.84145
```

```
## RPO.l8                 0.0112209  0.0625721   0.179  0.85781
## adj_stock_returns.l8   0.0057698  0.0594296   0.097  0.92273
## GOPC.l9               -0.0002943  0.0109548  -0.027  0.97859
## GRA.l9                -0.0388540  0.0713441  -0.545  0.58645
## RPO.l9                -0.0010918  0.0622841  -0.018  0.98603
## adj_stock_returns.l9   0.0339309  0.0589409   0.576  0.56529
## GOPC.l10              -0.0006848  0.0108346  -0.063  0.94965
## GRA.l10                0.0084041  0.0696297   0.121  0.90402
## RPO.l10               -0.0074897  0.0619075  -0.121  0.90379
## adj_stock_returns.l10  0.0083087  0.0587031   0.142  0.88754
## GOPC.l11              -0.0051669  0.0107915  -0.479  0.63245
## GRA.l11                0.0463181  0.0686591   0.675  0.50047
## RPO.l11               -0.0081882  0.0624277  -0.131  0.89574
## adj_stock_returns.l11  0.0698314  0.0585166   1.193  0.23372
## GOPC.l12               0.0116714  0.0108304   1.078  0.28210
## GRA.l12               -0.0581981  0.0680624  -0.855  0.39323
## RPO.l12                0.0218335  0.0631009   0.346  0.72959
## adj_stock_returns.l12 -0.0575063  0.0586018  -0.981  0.32727
## GOPC.l13               0.0188778  0.0110113   1.714  0.08754 .
## GRA.l13               -0.0277399  0.0674992  -0.411  0.68140
## RPO.l13               -0.0233724  0.0624223  -0.374  0.70837
## adj_stock_returns.l13  0.0300496  0.0592782   0.507  0.61260
## GOPC.l14              -0.0032292  0.0110493  -0.292  0.77031
## GRA.l14                0.0808361  0.0677108   1.194  0.23353
## RPO.l14                0.0427256  0.0604560   0.707  0.48031
## adj_stock_returns.l14 -0.0577901  0.0589140  -0.981  0.32746
## GOPC.l15               0.0069824  0.0110808   0.630  0.52911
## GRA.l15                0.0119504  0.0677198   0.176  0.86005
## RPO.l15               -0.0414493  0.0604878  -0.685  0.49374
## adj_stock_returns.l15 -0.0661182  0.0590430  -1.120  0.26373
## GOPC.l16               0.0133542  0.0111181   1.201  0.23070
## GRA.l16                0.0475049  0.0674718   0.704  0.48196
## RPO.l16                0.0496877  0.0602152   0.825  0.40996
## adj_stock_returns.l16  0.0461049  0.0596469   0.773  0.44018
## GOPC.l17              -0.0037900  0.0108868  -0.348  0.72800
## GRA.l17               -0.1265182  0.0678232  -1.865  0.06315 .
## RPO.l17               -0.0435599  0.0600850  -0.725  0.46906
## adj_stock_returns.l17  0.0484917  0.0596569   0.813  0.41698
## GOPC.l18              -0.0027102  0.0106984  -0.253  0.80020
## GRA.l18                0.0996158  0.0678615   1.468  0.14322
## RPO.l18                0.0125645  0.0599322   0.210  0.83409
## adj_stock_returns.l18 -0.0074742  0.0600380  -0.124  0.90101
## GOPC.l19              -0.0105064  0.0103021  -1.020  0.30867
## GRA.l19               -0.1219323  0.0683439  -1.784  0.07547 .
## RPO.l19               -0.0404668  0.0605630  -0.668  0.50456
## adj_stock_returns.l19 -0.0999304  0.0602551  -1.658  0.09832 .
## GOPC.l20               0.0122550  0.0102141   1.200  0.23120
## GRA.l20                0.0793530  0.0696041   1.140  0.25521
## RPO.l20                0.0202982  0.0615985   0.330  0.74200
## adj_stock_returns.l20  0.0323539  0.0598168   0.541  0.58901
## GOPC.l21               0.0139887  0.0101618   1.377  0.16971
## GRA.l21                0.0414962  0.0710325   0.584  0.55956
## RPO.l21                0.0907745  0.0616768   1.472  0.14218
## adj_stock_returns.l21 -0.0724002  0.0592184  -1.223  0.22249
```

```
## GOPC.l22              0.0210792  0.0099460   2.119  0.03492 *
## GRA.l22              -0.0603178  0.0705836  -0.855  0.39351
## RPO.l22              -0.1071424  0.0618584  -1.732  0.08434 .
## adj_stock_returns.l22  0.0158837  0.0592784   0.268  0.78893
## GOPC.l23             -0.0132102  0.0099601  -1.326  0.18580
## GRA.l23               0.0069675  0.0707574   0.098  0.92163
## RPO.l23               0.0009223  0.0592639   0.016  0.98759
## adj_stock_returns.l23 -0.0578641  0.0591099  -0.979  0.32845
## GOPC.l24             -0.0093367  0.0098907  -0.944  0.34597
## GRA.l24               0.0180769  0.0459008   0.394  0.69400
## RPO.l24               0.0331659  0.0336363   0.986  0.32496
## adj_stock_returns.l24  0.0527249  0.0565554   0.932  0.35198
## const                 0.3786477  0.2053806   1.844  0.06627 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 3.379 on 286 degrees of freedom
## Multiple R-Squared: 0.2919,  Adjusted R-squared: 0.05423
## F-statistic: 1.228 on 96 and 286 DF,  p-value: 0.1006
##
##
##
## Covariance matrix of residuals:
##                       GOPC     GRA     RPO adj_stock_returns
## GOPC             340.4719 -0.2916 -11.836           -3.8614
## GRA               -0.2916 16.8726   1.685           -0.1231
## RPO              -11.8361  1.6852  35.138           -3.1072
## adj_stock_returns  -3.8614 -0.1231  -3.107           11.4194
##
## Correlation matrix of residuals:
##                        GOPC       GRA      RPO adj_stock_returns
## GOPC              1.000000 -0.003847 -0.10821         -0.061928
## GRA              -0.003847  1.000000  0.06921         -0.008865
## RPO              -0.108213  0.069209  1.00000         -0.155116
## adj_stock_returns -0.061928 -0.008865 -0.15512          1.000000
```