# ECM and IRF of Vegetable Oils

FRE530 Assignment 2 (12.5 points)
Due in Canvas *before* midnight (11:59pm) on March 30, 2022

## Background

This assignment has three main objectives: (1) reinforce the time series topics covered in class, (2) build your intuition about time series in economics within the FRE sector, and (3) build your R toolkit.

You may find loading the following libraries helpful in completing the assignment: `pacman::p_load(here, readr, dplyr, tidyr, janitor, xts, lubridate, urca, forecast, vars, modelsummary)`

## Johansen Test (1 point)

Recall that in Assignment 1, we used the Engle-Granger test to determine whether the three pairs of prices were cointegrated or not. To determine if all three price series are simultaneously cointegrated we must use the Johansen test (code is in the Appendix). To confirm the presence of cointegration, this test requires us to establish that the rank of the cointegration matrix is greater than zero (i.e., reject the rank = 0 null)

- Using `readRDS()`, read the `vegoils.RDS` data you created from Assignment 1. Call this object as `vegoils`.
- Use `xts()` to convert `lnpalm`, `lnsoy`, `lnrapeseed` into a time series object and call it `vegoils_ts`.
- Determine the number of lags to include in the Johansen Test. *Hint: Find the lowest AIC for all three variables simultaneously.*
- Conduct a Johansen Trace Test and interpret the results. (1 point)

```
# lag length
VARselect(vegoils_ts, lag.max = 5)$select
```

```
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      3      2      2      3
```

```
# based on trace statistic
summary(ca.jo(vegoils_ts, type = c("trace"), K = 3, ecdet = "none"))
```

```
##
## ######################
## # Johansen-Procedure #
## ######################
##
## Test type: trace statistic , with linear trend
##
## Eigenvalues (lambda):
## [1] 0.13533993 0.04695382 0.02728540
##
## Values of teststatistic and critical values of test:
##
##           test 10pct  5pct  1pct
## r <= 2 |  5.89  6.50  8.18 11.65
## r <= 1 | 16.14 15.66 17.95 23.52
## r = 0  | 47.11 28.71 31.52 37.22
##
```

```
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##                 lnpalm.l3  lnsoy.l3 lnrapeseed.l3
## lnpalm.l3      1.0000000  1.000000     1.0000000
## lnsoy.l3      -1.2418919 -9.467084     1.0660843
## lnrapeseed.l3  0.1635218 10.568061    -0.7574165
##
## Weights W:
## (This is the loading matrix)
##
##                lnpalm.l3     lnsoy.l3 lnrapeseed.l3
## lnpalm.d     0.01195873  0.001794735   -0.02234251
## lnsoy.d      0.16801118  0.001242742   -0.01200869
## lnrapeseed.d 0.11121952 -0.008954803   -0.01046823
```

### Suggested Answers

- Lag length: Based on the lowest AIC for all three variables simultaneously, we should include 3 lags in the model.
- Johansen Test: We begin with the bottom row and test the hypothesis r = 0. The trace test statistic is 47.11, which is bigger than any of the critical values provided. We reject the null hypothesis of rank = 0. Next, we test the hypothesis that r <=1. The trace test statistic of 16.14 is smaller than the 95% critical value of 17.95. We fail to reject the null hypothesis that rank <=1, implying rank = 1. We conclude that the system of three prices is cointegrated.

# Error Correction Model (5 points)

With more than two variables we must use the vector error correction model (VECM) and the Johansen method of estimation. This method is beyond the scope of this class and so we will instead estimate a regular two-variable error correction model (ECM) using one pair of prices. Let's work with the palm oil-rapeseed oil pair.

- Estimate the long run relationship for soybean oil, which is specified as $p^{palm} = \alpha + \beta_1 p^{rapeseed} + \epsilon$.
- Use `resid()` to save the residuals from the long run relationship. Then create a new dataframe called `vegoils_r` that merges `vegoils` with the residuals from the equation you just estimated using the `merge.xts()` function. *Hint: You will have to convert the* `vegoils` *and residuals into an xts object first before doing the merge.* (1 point)
- Determine the number of lags to include in the error correction model for each variable. (1 point)
- Estimate the general ECM model and interpret your results (1 point)
- Does the speed of adjustment make economic sense? You may refer to FRE501 or other sources (2 points)

### Suggested Answers

- Long run dynamics: The positive coefficient $\beta_1 = 1.16$ suggests that palm oil and rapeseed oil are substitutes. Remember that the prices are in logs and so the units of measure are not important (i.e., $\beta_1 = 1.16$ implies the percentage changes are similar when the pair of prices are shocked). As we have discussed in FRE501, rapeseed is considered as a substitute for palm oil in the production of biodiesel.
- Based on the AIC, we will use 5 lags for palm oil, and 1 lag for rapeseed oil.
- The ECM model shows that:
  - $\Delta Palm_{t-1}$, $\Delta Palm_{t-2}$, $\Delta Palm_{t-4}$, and $\Delta Palm_{t-5}$ are significant and play a role in influencing $\Delta Palm_t$
  - $\Delta Rapeseed_t$ and $\Delta Rapeseed_{t-1}$ are also statistically signfiicant
  - $\lambda = 0.048$ suggests a relatively slow speed of adjustment
- In FRE 501 you learned that shipping palm oil from Malaysia and other countries to the EU is a slow process. As well, it may take considerable time to switch feed stocks in the production of biodiesel (i.e., palm oil and rapeseed oil may be close substitutes but they are not perfect substitutes). For these reasons, it is not unusual to find that price adjustments back to the long term equilibrium are relatively slow.

We now analyze the dynamic relationship between these three vegetable oil prices using VAR and IRF. To run a VAR model, the price series must be stationary, and we must use levels instead of first differences. From the first assignment, we showed that each of the price series is not stationary. One way to address this issue is to deflate prices by an index.

# Data download and data cleaning (1.5 points)

- Download the FAO price index for vegetable oils (CSV) here
- Using `read_excel()`, load the FAO price index data and call it `fao`. *Hint: You can add* `skip = 2` *to skip the first two lines. You can also use the* `clean_names()` *function right away to fix the variable names.*
  - If you used `clean_names()`, rename `x1` to `year`, `x2` to `month`, and `oils` to `oil_ppi`
  - You will notice that the year only appears for January of that year. You can use `fill(year)` for R to fill missing values based on the previous entry. Read here for info.
  - Using `mutate()` and `as.Date()`, format the date into a format that R recognizes.
  - Use `filter()` to keep only observations from January 1, 2003 to December 1, 2020.
- Merge `vegoils` and `fao` the two dataframes together by the `date` column. Call this dataframe as `vegoils_fao`.
  - Using `mutate()`, deflate the commodity prices (`palm_oil`, `soybean_oil`, `rapeseed_oil`) by `oil_ppi` and call these columns `palmr`, `soyr`, and `rapeseedr`, respectively.
  - Using `mutate()` take the natural log of each deflated price series. Call these columns `lnpalmr`, `lnsoyr`, and `lnrapeseedr`, respectively.
  - Using `select(-)`, remove the `palm_oil`, `soybean_oil`, and `rapeseed_oil` columns.
- Using `xts()`, convert the dataframe into an xts object. Call this new object `vegoils_real`.
- The first 5 rows of your `vegoils_real` dataframe should match the output below.

```
## [1] "xts" "zoo"
```

```
##             lnpalmr   lnsoyr lnrapeseedr
## 2003-01-01 6.627734 6.734201    6.877001
## 2003-02-01 6.639818 6.730898    6.845317
## 2003-03-01 6.640848 6.782405    6.837145
## 2003-04-01 6.611633 6.811253    6.840978
## 2003-05-01 6.610523 6.801358    6.901631
```

# VAR and Impulse Response Functions (5 points)

- Perform a test to confirm that the natural log deflated prices of each commodity is stationary. Follow the steps in the Stationary testing flowchart. Make sure you control for autocorrelation and test for the optimal lag length (for all 3 variables simultaneously) to use (0.5 point)
- Estimate a basic VAR model. Make sure you include the optimal number of lags. (0.5 point)
- Perform at least 2 diagnostic tests of your VAR model (0.5 point)
- Perform impulse responses functions and plot palm-soybean, palm-rapeseed, and palm-palm IRF plots nicely (0.5 point)
- Explain the IRF plots for palm-soybean, palm-rapeseed, and palm-palm, first while focusing only on the IRF schedules (i.e., ignoring the confidence intervals) and then accounting for the confidence intervals. (3 points)

### Suggested Answers

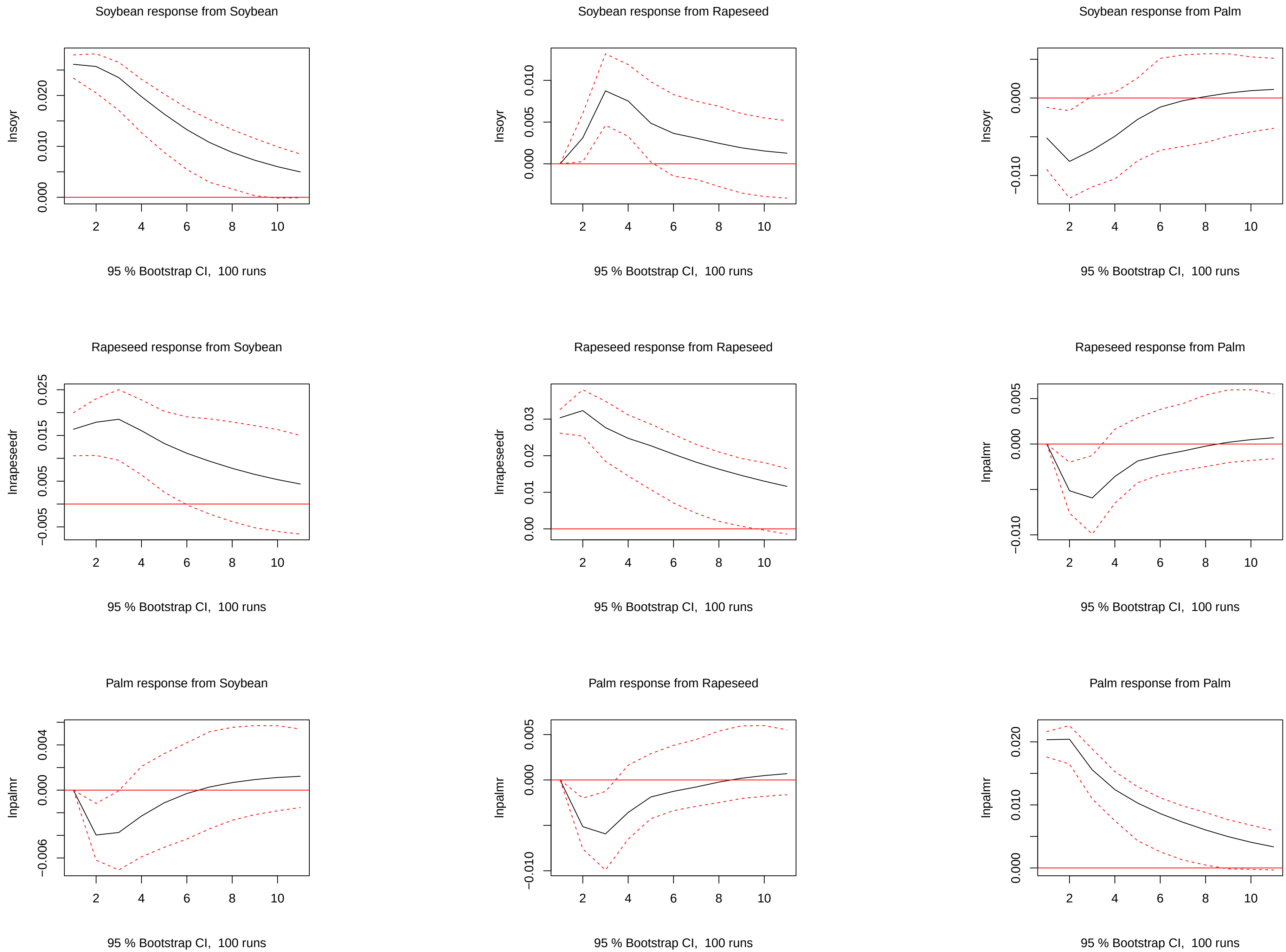
- ADF tests:
  - Palm Oil - The flowchart begins with specifying `type = c("trend")`. The $\tau_3$ test statistic is -4.0288 and the critical values are -3.99, -3.43, -3.13, so the null hypothesis of a unit root can be rejected. We conclude that the natural log deflated palm oil price is stationary.
  - Soybean Oil - The flowchart begins with specifying `type = c("trend")`. The $\tau_3$ test statistic is -4.0834 and the critical values are -3.99, -3.43, -3.13, so the null hypothesis of a unit root can be rejected at the 1% level. We conclude that the natural log deflated soybean oil price is stationary.
  - Rapeseed Oil - The flowchart begins with specifying `type = c("trend")`. The $\tau_3$ test statistic is -3.291 and the critical values are -3.99, -3.43, -3.13, so the null hypothesis of a unit root can be rejected at the 5% level. We conclude that the natural log deflated rapeseed oil price is stationary.

  - Since each of the log vegetable prices are stationary, we can proceed with the VAR model.

- Using R's `VARselect()` function, we compared various information criteria of up to 10 lags. The AIC is smallest when lags = 3, hence we conclude to use 3 lags in our VAR model.

- Tests
  - Autocorrelation: Portmanteau Test - p-value of 0.644, so fail to reject the null hypothesis of no autocorrelation
  - Normality: JB test - p-value of 0.525 so we fail to reject the null hypothesis; errors are normally distributed

- IRFS interpretation (soybean response from shocks from soybean, rapeseed, and palm oils)
  - Palm response from Soybean - a) Initial decrease in price up to period 2 and then gradually returning to LR price at around period 7. b) Pretty much no response because the confidence intervals (red lines) all cross 0
  - Palm response from Rapeseed - a) Initial decrease in price up to period 3 and then gradual return to LR price at around period 9. Slightly sower return to LR price compared to soybean. b) Decrease for about 3 periods then the effect goes to zero because the confidence intervals cross 0 already.
  - Palm response from Palm - a) Slight positive effect initially that gradually fades to 0. b) Similar interpretation because confidence intervals do not cross 0.
  - In FRE501, we discussed that in Europe palm oil and rapeseed oil are used in biodiesel production, but soybean oil is not. Thus, we would expect a stronger pricing dynamics between these two vegetable oils compared to soybean oil and palm oil.


Soybean response from Soybean
lnsoyr
95 % Bootstrap CI, 100 runs
Soybean response from Rapeseed
lnsoyr
95 % Bootstrap CI, 100 runs
Soybean response from Palm
lnsoyr
95 % Bootstrap CI, 100 runs
Rapeseed response from Soybean
lnrapeseedr
95 % Bootstrap CI, 100 runs
Rapeseed response from Rapeseed
lnrapeseedr
95 % Bootstrap CI, 100 runs
Rapeseed response from Palm
lnpalmr
95 % Bootstrap CI, 100 runs
Palm response from Soybean
lnpalmr
95 % Bootstrap CI, 100 runs
Palm response from Rapeseed
lnpalmr
95 % Bootstrap CI, 100 runs
Palm response from Palm
lnpalmr
95 % Bootstrap CI, 100 runs

# Code and Output

## Error Correction Model

```
# estimate palm oil's long run relationship
lr_palmrapeseed <- lm(lnpalm ~ lnrapeseed, data = vegoils)
summary(lr_palmrapeseed)
```

```
##
## Call:
## lm(formula = lnpalm ~ lnrapeseed, data = vegoils)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.32025 -0.08803  0.02476  0.09170  0.21653
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.31935    0.24105  -5.473 1.23e-07 ***
## lnrapeseed   1.16191    0.03544  32.788  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.122 on 214 degrees of freedom
## Multiple R-squared:  0.834,  Adjusted R-squared:  0.8332
## F-statistic:  1075 on 1 and 214 DF,  p-value: < 2.2e-16
```

```
# save the residuals
resid_pr <- resid(lr_palmrapeseed)

VARselect(vegoils$lnpalm, lag.max = 5)$select #5 lags
```

```
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      5      3      2      5
```

```
VARselect(vegoils$lnrapeseed, lag.max = 5)$select #2 lags
```

```
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      2      2      2      2
```

```
# convert vegoils and residuals to xts object then merge
vegoils_ts <- xts(vegoils[,c("lnpalm", "lnrapeseed")], order.by = vegoils$date)
resid_pr_ts <- xts(resid_pr, order.by = vegoils$date)
vegoils_r <- merge.xts(vegoils_ts, resid_pr_ts)
head(vegoils_r)
```

```
##              lnpalm lnrapeseed  resid_pr_ts
## 2003-01-01 6.186435   6.435702  0.028100219
## 2003-02-01 6.167412   6.372910  0.082034687
## 2003-03-01 6.117106   6.313403  0.100870163
## 2003-04-01 6.093615   6.322960  0.066275190
## 2003-05-01 6.117701   6.408809 -0.009387231
## 2003-06-01 6.144636   6.417173  0.007829483
```

| | Palm-Rapeseed |
|---|---|
| (Intercept) | 0.001 |
| | (0.003) |
| LD_palm | 0.429*** |
| | (0.070) |
| L2D_palm | −0.162* |
| | (0.075) |
| L3D_palm | 0.018 |
| | (0.064) |
| L4D_palm | 0.110+ |
| | (0.064) |
| L5D_palm | −0.147* |
| | (0.058) |
| D_rapeseed | 0.759*** |
| | (0.075) |
| LD_rapeseed | −0.232* |
| | (0.092) |
| L2D_rapeseed | 0.048 |
| | (0.091) |
| ECT | −0.082** |
| | (0.029) |
| Num.Obs. | 210 |
| R2 | 0.495 |
| R2 Adj. | 0.472 |
| AIC | −689.1 |
| BIC | −652.2 |
| Log.Lik. | 355.532 |
| F | 21.753 |

```r
# estimate ecm for palm oil
model <- list("Palm-Rapeseed" = lm(diff.xts(lnpalm) ~
               lag.xts(diff.xts(lnpalm)) +
               lag.xts(diff.xts(lnpalm), k = 2) +
               lag.xts(diff.xts(lnpalm), k = 3) +
               lag.xts(diff.xts(lnpalm), k = 4) +
               lag.xts(diff.xts(lnpalm), k = 5) +
               diff.xts(lnrapeseed) +
               lag.xts(diff.xts(lnrapeseed), k = 1) +
               lag.xts(diff.xts(lnrapeseed), k = 2) +
               lag.xts(resid_pr_ts), data = vegoils_r))

modelsummary(model, estimate = "{estimate}{stars}",
             coef_rename = c("lag.xts(diff.xts(lnpalm))" = "LD_palm", "lag.xts(diff.xts(lnpalm), k = 2)" = "L2D_
                             "lag.xts(diff.xts(lnpalm), k = 3)" = "L3D_palm", "lag.xts(diff.xts(lnpalm), k = 4)'
```

## Data Cleaning

```r
fao <- read_csv(here("data", "fao_price_index.csv"), skip = 2) %>%
  clean_names()

fao <- fao %>%
  rename(year = x1,
         month = x2,
         oil_ppi = oils) %>%
  fill(year) %>%
  mutate(date = as.Date(paste0(year, "-", month, "-01"), format("%Y-%B-%d"))) %>%
```

```r
  filter(date >= "2003-01-01" & date <= "2020-12-01") %>%
  dplyr::select(date, oil_ppi)

vegoils_fao <- merge(vegoils, fao, by = c("date"))

vegoils_fao <- vegoils_fao %>%
  mutate(palmr = as.numeric(palm_oil) / oil_ppi * 100,
         soyr = as.numeric(soybean_oil) / oil_ppi * 100,
         rapeseedr = as.numeric(rapeseed_oil) / oil_ppi * 100,
         lnpalmr = log(palmr),
         lnsoyr = log(soyr),
         lnrapeseedr = log(rapeseedr)) %>%
  dplyr::select(-palm_oil, -soybean_oil, -rapeseed_oil)

vegoils_real <- xts(vegoils_fao[, 9:11], order.by = vegoils_fao$date)
```

### VAR and Impulse Response Functions

```r
# ADF test
adf_palm <- ur.df(vegoils_real$lnpalmr, type = c("drift"), lags = 8, selectlags = c("AIC"))
adf_soy <- ur.df(vegoils_real$lnsoyr, type = c("drift"), lags = 2, selectlags = c("AIC"))
adf_rapeseed <- ur.df(vegoils_real$lnrapeseedr, type = c("drift"), lags = 2, selectlags = c("AIC"))

# VAR lag selection
VARselect(vegoils_real, type = c("none"))
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      3      2      1      3
##
## $criteria
##                    1             2             3             4             5
## AIC(n) -2.173286e+01 -2.184616e+01 -2.187986e+01 -2.180934e+01 -2.176348e+01
## HQ(n)  -2.167406e+01 -2.172855e+01 -2.170345e+01 -2.157413e+01 -2.146947e+01
## SC(n)  -2.158746e+01 -2.155537e+01 -2.144368e+01 -2.122777e+01 -2.103652e+01
## FPE(n)  3.643702e-10  3.253542e-10  3.146090e-10  3.376729e-10  3.536529e-10
##                    6             7             8             9            10
## AIC(n) -2.171794e+01 -2.170442e+01 -2.168696e+01 -2.164589e+01 -2.160239e+01
## HQ(n)  -2.136513e+01 -2.129281e+01 -2.121655e+01 -2.111667e+01 -2.101437e+01
## SC(n)  -2.084558e+01 -2.068667e+01 -2.052382e+01 -2.033735e+01 -2.014846e+01
## FPE(n)  3.703416e-10  3.756779e-10  3.826980e-10  3.992892e-10  4.177524e-10
```

```r
# VAR model
model1 <- VAR(vegoils_real, p = 3, type = c("const"))
coef(model1)
```

```
## $lnpalmr
##                   Estimate Std. Error    t value     Pr(>|t|)
## lnpalmr.l1      0.89136878 0.07375349 12.0857840 1.095593e-25
## lnsoyr.l1      -0.04597485 0.05948051 -0.7729397 4.404571e-01
## lnrapeseedr.l1 -0.16924894 0.04548915 -3.7206439 2.572072e-04
## lnpalmr.l2     -0.22835580 0.09470516 -2.4112286 1.678943e-02
## lnsoyr.l2       0.06551938 0.08079175  0.8109663 4.183348e-01
## lnrapeseedr.l2  0.14015138 0.06272668  2.2343185 2.655250e-02
## lnpalmr.l3      0.16876426 0.07340351  2.2991307 2.251499e-02
## lnsoyr.l3      -0.01199392 0.05972205 -0.2008290 8.410334e-01
## lnrapeseedr.l3  0.02941037 0.04545287  0.6470520 5.183293e-01
## const           1.05857899 0.41063437  2.5779113 1.064810e-02
```

```
##
## $lnsoyr
##                   Estimate Std. Error    t value     Pr(>|t|)
## lnpalmr.l1     -0.10790043 0.09655537 -1.1174980 2.651023e-01
## lnsoyr.l1       0.91898446 0.07786970 11.8015663 8.154265e-25
## lnrapeseedr.l1  0.10258773 0.05955273  1.7226369 8.647715e-02
## lnpalmr.l2      0.26197275 0.12398453  2.1129471 3.582710e-02
## lnsoyr.l2      -0.13187330 0.10576959 -1.2467978 2.139083e-01
## lnrapeseedr.l2  0.06620491 0.08211947  0.8062024 4.210696e-01
## lnpalmr.l3     -0.07246884 0.09609718 -0.7541203 4.516508e-01
## lnsoyr.l3       0.05576492 0.07818593  0.7132348 4.765196e-01
## lnrapeseedr.l3 -0.14382896 0.05950523 -2.4170810 1.652950e-02
## const           0.35181664 0.53758748  0.6544361 5.135717e-01
##
## $lnrapeseedr
##                   Estimate Std. Error    t value     Pr(>|t|)
## lnpalmr.l1     -0.43043978 0.13255332 -3.2472954 1.363151e-03
## lnsoyr.l1       0.01908283 0.10690123  0.1785090 8.585013e-01
## lnrapeseedr.l1  1.06401006 0.08175529 13.0145715 1.486142e-28
## lnpalmr.l2      0.44904493 0.17020867  2.6382025 8.980677e-03
## lnsoyr.l2       0.08176260 0.14520281  0.5630924 5.739934e-01
## lnrapeseedr.l2 -0.29581142 0.11273540 -2.6239444 9.352259e-03
## lnpalmr.l3     -0.14072361 0.13192431 -1.0666996 2.873745e-01
## lnsoyr.l3      -0.11581616 0.10733534 -1.0790124 2.818626e-01
## lnrapeseedr.l3  0.13832895 0.08169008  1.6933384 9.192498e-02
## const           1.54596903 0.73801182  2.0947754 3.743153e-02
```

```
# VAR tests
autocorrelation <- serial.test(model1) # Portmanteau Test
normality <- normality.test(model1)

# IRF
irf_ss <- irf(model1, impulse = c("lnsoyr"), response = c("lnsoyr"), n.ahead = 10, boot = TRUE, runs = 100, ci =
irf_sr <- irf(model1, impulse = c("lnrapeseedr"), response = c("lnsoyr"), n.ahead = 10, boot = TRUE, runs = 100,
irf_sp <- irf(model1, impulse = c("lnpalmr"), response = c("lnsoyr"), n.ahead = 10, boot = TRUE, runs = 100, ci

irf_rs <- irf(model1, impulse = c("lnsoyr"), response = c("lnrapeseedr"), n.ahead = 10, boot = TRUE, runs = 100,
irf_rr <- irf(model1, impulse = c("lnrapeseedr"), response = c("lnrapeseedr"), n.ahead = 10, boot = TRUE, runs =
irf_pr <- irf(model1, impulse = c("lnpalmr"), response = c("lnrapeseedr"), n.ahead = 10, boot = TRUE, runs = 100

irf_ps <- irf(model1, impulse = c("lnsoyr"), response = c("lnpalmr"), n.ahead = 10, boot = TRUE, runs = 100, ci
irf_pr <- irf(model1, impulse = c("lnrapeseedr"), response = c("lnpalmr"), n.ahead = 10, boot = TRUE, runs = 100
irf_pp <- irf(model1, impulse = c("lnpalmr"), response = c("lnpalmr"), n.ahead = 10, boot = TRUE, runs = 100, ci
```

```
par(mfrow=c(3,3), mar = c(1, 1, 1 ,1), oma = c(0, 0,0,0))
plot(irf_ss, main = "Soybean response from Soybean")
plot(irf_sr, main = "Soybean response from Rapeseed")
plot(irf_sp, main = "Soybean response from Palm")

plot(irf_rs, main = "Rapeseed response from Soybean")
plot(irf_rr, main = "Rapeseed response from Rapeseed")
plot(irf_pr, main = "Rapeseed response from Palm")

plot(irf_ps, main = "Palm response from Soybean")
plot(irf_pr, main = "Palm response from Rapeseed")
plot(irf_pp, main = "Palm response from Palm")
```