Cointegration Analysis in Time Series

Kai Chien Chen

- Cointegration
-
- **PPP**

I(1)

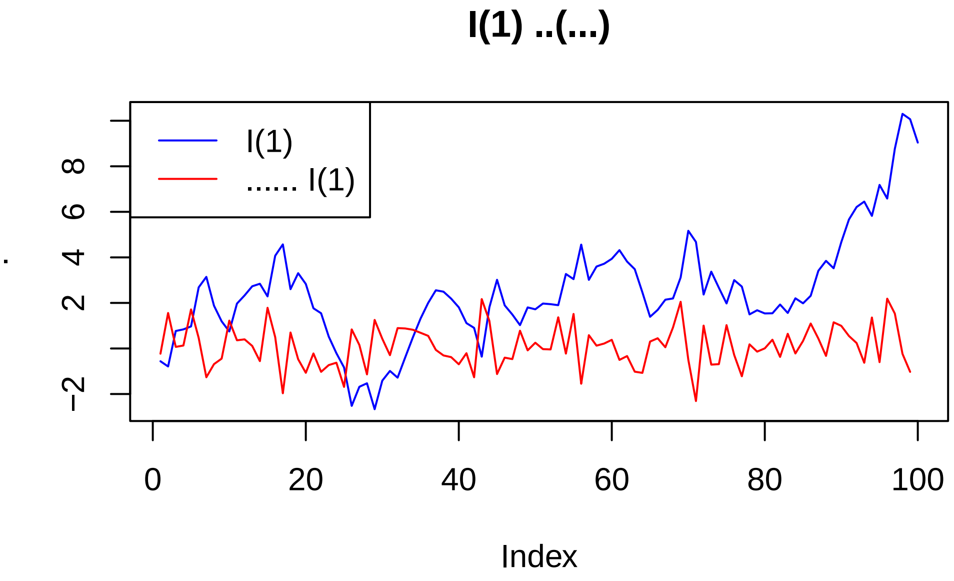
-
- **I(1)**
  - GDP

I(1)

```r
# 1.      I(1)     (  )
set.seed(123)  #
n <- 100
I1_series <- cumsum(rnorm(n))  #      I(1)

# 2.
diff_series <- diff(I1_series)

# 3.      I(1)
plot(I1_series, type = 'l', col = 'blue', main = "I(1)     ", ylab = " ")
lines(diff_series, type = 'l', col = 'red')
legend("topleft",lty = 1,col = c("blue","red"),legend = c("I(1)","    I(1)"))
```



PPP PPP q_t s_t p_t p_t^*

$$q_t = s_t + p_t^* - p_t$$

- $I(1)$
- A $A'y_t \sim I(0)$ y_t I(1)

-
-

- ECM
- $\Delta y_t = \beta_0 \Delta Z_t + (\varphi - 1)[y_{t-1} - (\frac{\beta_0+\beta_1}{1-\varphi})Z_{t-1}] + \epsilon_t$

-

$$y_t = \varphi y_{t-1} + \beta_0 Z_t + \beta_1 Z_{t-1} + \epsilon_t$$

$Z_t = Z_{t-1} + u_t$

$$u_t \sim N(0, \sigma^2)$$

Engle-Granger (1)

1. y_t z_t
2.

$$y_t = \beta_0 + \beta_1 z_t + e_t$$

3. $\hat{e_t}$ ADF

$$\Delta \hat{e} = a_0 + a_1 \widehat{e_{t-1}} + \sum_{i=1}^{n} a_{i+1} \Delta \widehat{e_{t-i}} + \epsilon_t$$

Engle-Granger (2)

1. $H_0 : a_1 = 0$ $H_1 : a_1 < 0$
2. H_0

Johansen

- **Johansen**

VAR(p) VECM

$\mathbf{X}_t$ k

$$\mathbf{X}_t = \mathbf{A}_1 \mathbf{X}_{t-1} + \mathbf{A}_2 \mathbf{X}_{t-2} + \cdots + \mathbf{A}_p \mathbf{X}_{t-p} + {}_t$$

VECM

$$\Delta \mathbf{X}_t = \Pi \mathbf{X}_{t-1} + \sum_{i=1}^{p-1} \Gamma_i \Delta \mathbf{X}_{t-i} + {}_t$$

According to (Beveridge and Nelson JME 1981)

Π

- $\Pi = \alpha\beta'$
  - β:
  - α:

Π

Π

1. **0** $\text{rank}(\Pi) = 0$
   -
2. **0** k $0 < \text{rank}(\Pi) < k$
   - r r
3. k $\text{rank}(\Pi) = k$
   -

Π eigenvalue

1.

$$\begin{aligned} rank(\Pi) = 0 &\Rightarrow \lambda_1 = \lambda_2 = \cdots = \lambda_k = 0 \\ &\Rightarrow log(1-\lambda_i) = 0 \quad \forall i \\ rank(\Pi) = k &\Rightarrow log(1-\lambda_i) \neq 0 \quad \forall i \end{aligned}$$

x_t

---

2. $rank(\Pi) = r$

$$\begin{cases} \lambda_1, \lambda_2, \cdots, \lambda_r \neq 0 \\ \lambda_{r+1} = \lambda_{r+2} = \cdots = \lambda_k = 0 \end{cases}$$

$$\begin{cases} log(1-\lambda_i) \neq 0 \quad for \quad i = 1, 2, \cdots, r \\ log(1-\lambda_i) = 0 \quad for \quad i = r+1, r+2, \cdots, k \end{cases}$$

x_t

Johansen

1.

   - H_0: r ( )
   - H_1: k or $r+1$

2.

   - r
   - $r+1$

(Trace Test)

$$\text{Trace Statistic} = -T\sum_{i=r+1}^{k}\ln(1-\lambda_i)$$

- $>$ H_0 r

(Max Test)

$$\text{Max-Eigen Statistic} = -T\ln(1-\lambda_{r+1})$$

- $>$ H_0 $r+1$
-
- Step1: $H_0: r=0 \quad vs. \quad H_1: r=1$ $H_0 \rightarrow$ step2
- Step2: $H_0: r=1 \quad vs. \quad H_1: r=2$ $H_0 \rightarrow H_0: r=2 \quad vs. \quad H_2: r=3$ H_0

Cointegration in R

- **R package**: `urca` for conducting the Johansen test.

R Code Example

1. Load Necessary Packages and data

```r
#   packages
# install.packages("vars")
library(urca)
```

```
Warning: package 'urca' was built under R version 4.4.2
```

```r
library(tidyverse)
library(tseries)
```

```
Warning: package 'tseries' was built under R version 4.4.2
```

```r
library(quantmod)
```

```r
#
getSymbols("EWA", from="2006-04-26", to="2012-04-09")
```

```
[1] "EWA"
```

```r
getSymbols("EWC", from="2006-04-26", to="2012-04-09")
```

```
[1] "EWC"
```

```r
getSymbols("IGE", from="2006-04-26", to="2012-04-09")
```

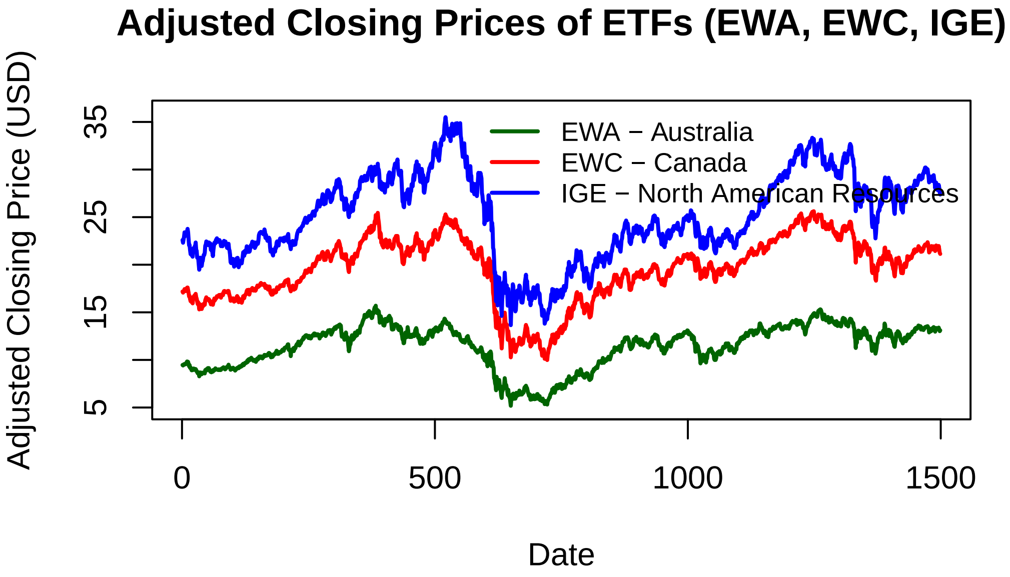
```
[1] "IGE"
```

```r
#
ewaAdj = unclass(EWA$EWA.Adjusted)
ewcAdj = unclass(EWC$EWC.Adjusted)
igeAdj = unclass(IGE$IGE.Adjusted)
```

:

1. EWA - iShares MSCI Australia ETF ETF
2. EWC - iShares MSCI Canada ETF ETF
3. IGE - iShares North American Natural Resources ETF ETF

2. Plot and describe the Data



```
Attaching package: 'table1'


The following objects are masked from 'package:base':

    units, units<-


Get nicer `table1` LaTeX output by simply installing the `kableExtra` package
```

	Overall
	(N=1499)
EWA.Adjusted	
Mean (SD)	11.4 (2.28)
Median [Min, Max]	12.1 [5.18, 15.7]

	Overall
EWC.Adjusted	
Mean (SD)	19.5 (3.41)
Median [Min, Max]	20.1 [9.99, 25.6]
IGE.Adjusted	
Mean (SD)	25.0 (4.57)
Median [Min, Max]	24.8 [13.7, 35.5]

5. Augmented Dickey-Fuller Test (ADF)

```
# ADF
adf.test(ewaAdj)
```

```

    Augmented Dickey-Fuller Test

data:  ewaAdj
Dickey-Fuller = -1.732, Lag order = 11, p-value = 0.6918
alternative hypothesis: stationary
```

```
adf.test(ewcAdj)
```

```

    Augmented Dickey-Fuller Test

data:  ewcAdj
Dickey-Fuller = -1.8042, Lag order = 11, p-value = 0.6612
alternative hypothesis: stationary
```

```
adf.test(igeAdj)
```

```

    Augmented Dickey-Fuller Test

data:  igeAdj
Dickey-Fuller = -1.9476, Lag order = 11, p-value = 0.6005
alternative hypothesis: stationary
```

6. Select the Optimal Lag for VAR

```
#   VARselect
library(vars)
```

```
Warning: package 'vars' was built under R version 4.4.2

Loading required package: MASS


Attaching package: 'MASS'

The following object is masked from 'package:dplyr':

    select

Loading required package: strucchange

Warning: package 'strucchange' was built under R version 4.4.2

Loading required package: sandwich


Attaching package: 'strucchange'

The following object is masked from 'package:stringr':

    boundary

Loading required package: lmtest

Warning: package 'lmtest' was built under R version 4.4.2
```

```
var_select <- VARselect(data.frame(ewaAdj, ewcAdj, igeAdj), lag.max = 10, type = "none")
var_select$selection
```

```
AIC(n)  HQ(n)  SC(n) FPE(n)
     4      2      2      4
```

7. Johansen Cointegration Test

```
# Johansen
jotest.t <- ca.jo(data.frame(ewaAdj, ewcAdj, igeAdj), type="trace", K=3, ecdet="none", spec="
summary(jotest.t)
```

```
######################
# Johansen-Procedure #
######################

Test type: trace statistic , with linear trend

Eigenvalues (lambda):
[1] 0.010707105 0.007278243 0.003526436

Values of teststatistic and critical values of test:

          test 10pct  5pct  1pct
r <= 2 |  5.28  6.50  8.18 11.65
r <= 1 | 16.21 15.66 17.95 23.52
r = 0  | 32.32 28.71 31.52 37.22

Eigenvectors, normalised to first column:
(These are the cointegration relations)

                EWA.Adjusted.l3 EWC.Adjusted.l3 IGE.Adjusted.l3
EWA.Adjusted.l3       1.0000000           1.000        1.000000
EWC.Adjusted.l3      -0.8294124        1959.344        5.869016
IGE.Adjusted.l3       0.1389769       -1621.201       -2.089886

Weights W:
(This is the loading matrix)

               EWA.Adjusted.l3 EWC.Adjusted.l3 IGE.Adjusted.l3
EWA.Adjusted.d    -0.001333744    7.065922e-06   -0.0007292514
EWC.Adjusted.d     0.030525641    7.334925e-06   -0.0009250704
IGE.Adjusted.d     0.042640887    1.711143e-05   -0.0006355286
```

```
jotest.m <- ca.jo(data.frame(ewaAdj, ewcAdj, igeAdj), type="eigen", K=3, ecdet="none", spec="
summary(jotest.m)
```

```
######################
# Johansen-Procedure #
######################

Test type: maximal eigenvalue statistic (lambda max) , with linear trend

Eigenvalues (lambda):
[1] 0.010707105 0.007278243 0.003526436

Values of teststatistic and critical values of test:

          test 10pct  5pct  1pct
r <= 2 |  5.28  6.50  8.18 11.65
r <= 1 | 10.93 12.91 14.90 19.19
r = 0  | 16.10 18.90 21.07 25.75

Eigenvectors, normalised to first column:
(These are the cointegration relations)

                EWA.Adjusted.l3 EWC.Adjusted.l3 IGE.Adjusted.l3
EWA.Adjusted.l3       1.0000000           1.000        1.000000
EWC.Adjusted.l3      -0.8294124        1959.344        5.869016
IGE.Adjusted.l3       0.1389769       -1621.201       -2.089886

Weights W:
(This is the loading matrix)

               EWA.Adjusted.l3 EWC.Adjusted.l3 IGE.Adjusted.l3
EWA.Adjusted.d    -0.001333744    7.065922e-06   -0.0007292514
EWC.Adjusted.d     0.030525641    7.334925e-06   -0.0009250704
IGE.Adjusted.d     0.042640887    1.711143e-05   -0.0006355286
```

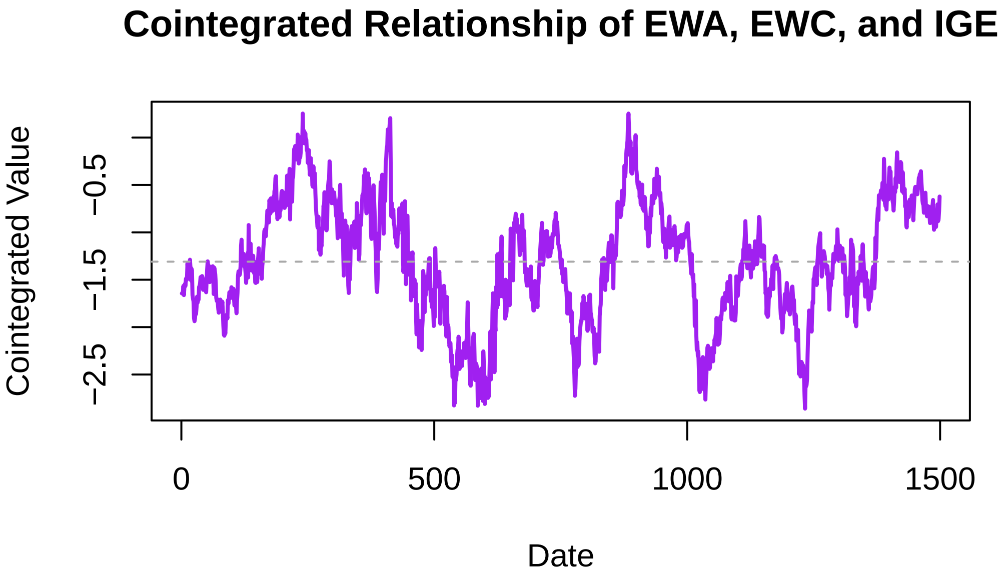
8. Calculate and Plot Cointegrated Relationship

```
#
alpha1 <- -0.8294165
```

```r
alpha2 <-  0.1389800

#      CI
CI <- coredata(ewaAdj) + alpha1 * coredata(ewcAdj) + alpha2 * coredata(igeAdj)

#
plot(CI, type = "l", col = "purple", lwd = 2,
     main = "Cointegrated Relationship of EWA, EWC, and IGE",
     xlab = "Date", ylab = "Cointegrated Value")
abline(h = mean(CI), col = "darkgray", lty = 2) #
```



9. ADF Test for Cointegrated Series

```r
# ADF
adf.test(CI)
```

```

    Augmented Dickey-Fuller Test

data:  CI
Dickey-Fuller = -3.1871, Lag order = 11, p-value = 0.09014
alternative hypothesis: stationary
```

```r
kpss.test(CI)
```

```
Warning in kpss.test(CI): p-value smaller than printed p-value
```

```

    KPSS Test for Level Stationarity

data:  CI
KPSS Level = 0.75332, Truncation lag parameter = 7, p-value = 0.01
```

10. Plot All Series and Cointegration

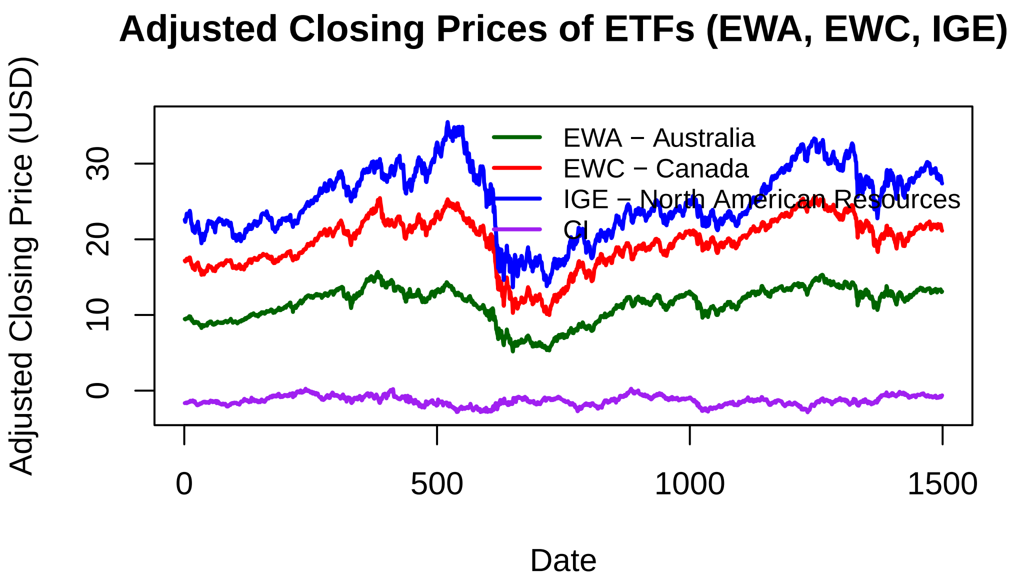
```r
#
plot(ewaAdj, type = "l", col = "darkgreen", lwd = 2, ylim = c(-3, 36),
     main = "Adjusted Closing Prices of ETFs (EWA, EWC, IGE)",
     xlab = "Date", ylab = "Adjusted Closing Price (USD)")

#      ewcAdj
lines(ewcAdj, col = "red", lwd = 2)

#      igeAdj
lines(igeAdj, col = "blue", lwd = 2)

#
lines(CI, col = "purple", lwd = 2)

#
legend("topright", legend = c("EWA - Australia", "EWC - Canada", "IGE - North American Resou
       col = c("darkgreen", "red", "blue", "purple"), lwd = 2, cex = 0.8, box.lty = 0)
```



- trace
-

- trace max
  - trace max
  - trace max
- trace max

	Johansen	Engle-Granger
	(VAR)	

	Trace Test	**Max Test**
	r $-T\sum_{i=r+1}^{k}\ln(1-\lambda_i)$	r $-T\ln(1-\lambda_{r+1})$

markdown

- ( ) :Time Series Analysis (3): Spurious regression and Cointegration
- : /
- https://www.quantstart.com/articles/Johansen-Test-for-Cointegrating-Time-Series-Analysis-in-R/
- https://github.com/KaiChienChen/Statistical-Consulting/blob/main/test_presentation.qmd

END