# trabfinal.R

nicholaslepetit

2020-05-29

```r
# Limpar tudo
rm(list=ls())

# Selecionar diretorio
setwd("/Users/nicholaslepetit/Library/Mobile Documents/com~apple~CloudDocs/Documents/ECONOMIA/02.2019/ECONOM
ETRIA II/trab.final/")
dir()
```

```
##  [1] "~$abfinal.docx"            "cambio_08.2004_11.2019.csv"
##  [3] "CEPEA_20191208181754.xls"  "db2.csv"
##  [5] "db3_03.2011_10.2019.csv"   "db4.csv"
##  [7] "ipeadata[08-12-2019-06-13].xls" "milho.csv"
##  [9] "milho2.csv"                "milho3.csv"
## [11] "Trab Final Econometria II"  "trabfinal.docx"
## [13] "trabfinal.html"            "trabfinal.log"
## [15] "trabfinal.R"               "trabfinal.spin.R"
## [17] "trabfinal.spin.Rmd"        "trabfinal.tex"
```

```r
dbmilho <- read.csv2("milho3.csv")
head(dbmilho)
```

```
##         date preco_milho
## 1 02/08/2004       18.24
## 2 03/08/2004       18.04
## 3 04/08/2004       18.02
## 4 05/08/2004       18.06
## 5 06/08/2004       18.13
## 6 09/08/2004       17.97
```

```r
library(anytime)
```

```
## Warning: package 'anytime' was built under R version 3.5.2
```

```r
dbmilho$date <- as.Date(dbmilho$date, tryFormats = c("%d/%m/%Y"))
head(dbmilho,20)
```

```
##          date preco_milho
## 1  2004-08-02       18.24
## 2  2004-08-03       18.04
## 3  2004-08-04       18.02
## 4  2004-08-05       18.06
## 5  2004-08-06       18.13
## 6  2004-08-09       17.97
## 7  2004-08-10       17.98
## 8  2004-08-11       18.08
## 9  2004-08-12       18.07
## 10 2004-08-13       18.04
## 11 2004-08-16       18.20
## 12 2004-08-17       18.22
## 13 2004-08-18       18.26
## 14 2004-08-19       18.35
## 15 2004-08-20       18.58
## 16 2004-08-23       18.65
## 17 2004-08-24       18.81
## 18 2004-08-25       18.77
## 19 2004-08-26       18.73
## 20 2004-08-27       18.97
```

```r
summary(dbmilho)
```

```
##       date           preco_milho
## Min.    :2004-08-02   Min.    :13.32
## 1st Qu.:2008-06-06   1st Qu.:20.74
## Median :2012-04-03   Median :27.06
## Mean    :2012-04-03   Mean    :27.90
## 3rd Qu.:2016-02-03   3rd Qu.:32.78
## Max.    :2019-11-29   Max.    :53.91
```

```
library(zoo)
```

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

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

```
dbmilho$date <- as.yearmon(dbmilho$date, "%d/%m/%Y")
summary(dbmilho$date)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    2005    2008    2012    2012    2016    2020
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 3.5.2
```

```
## ── Attaching packages ──────────────────────────────────────────────────────
────────────────────────── tidyverse 1.3.0 ──
```

```
## ✔ ggplot2 3.3.0      ✔ purrr   0.3.4
## ✔ tibble  3.0.1      ✔ dplyr   0.8.5
## ✔ tidyr   1.0.3      ✔ stringr 1.4.0
## ✔ readr   1.3.1      ✔ forcats 0.4.0
```

```
## Warning: package 'ggplot2' was built under R version 3.5.2
```

```
## Warning: package 'dplyr' was built under R version 3.5.2
```

```
## Warning: package 'stringr' was built under R version 3.5.2
```

```
## Warning: package 'forcats' was built under R version 3.5.2
```

```
## ── Conflicts ──────────────────────────────────────────────────────────────
────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
dbmilho <- dbmilho %>%
  group_by(date) %>%
  summarise(preco_milho = mean(preco_milho))


dbcambio <- read.csv2("cambio_08.2004_11.2019.csv", header = FALSE)
head(dbcambio)
```

```
##        V1
## 1 3.0029
## 2 2.8911
## 3 2.8529
## 4 2.7860
## 5 2.7182
## 6 2.6930
```

```
names(dbcambio)[1] <- "cambio"

db1 <- data.frame(dbmilho$preco_milho, dbcambio)
names(db1)[1] <- "pmilho"
head(db1)
```

```
##      pmilho cambio
## 1 18.36727 3.0029
## 2 19.04900 2.8911
## 3 18.26600 2.8529
## 4 17.70150 2.7860
## 5 17.14429 2.7182
## 6 17.81619 2.6930
```

```
db1.ts <- ts(db1, start = c(2004,8), end = c(2019,11), frequency = 12)
head(db1.ts)
```

```
##        pmilho cambio
## [1,] 18.36727 3.0029
## [2,] 19.04900 2.8911
## [3,] 18.26600 2.8529
## [4,] 17.70150 2.7860
## [5,] 17.14429 2.7182
## [6,] 17.81619 2.6930
```

```
library(dynlm)
```

```
## Warning: package 'dynlm' was built under R version 3.5.2
```

```
mod1 <- dynlm(pmilho ~ cambio, data = db1.ts)
summary(mod1)
```

```
##
## Time series regression with "ts" data:
## Start = 2004(8), End = 2019(11)
##
## Call:
## dynlm(formula = pmilho ~ cambio, data = db1.ts)
##
## Residuals:
##     Min       1Q   Median       3Q      Max
## -12.6818  -4.5476  -0.6069   5.1091  16.8971
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  11.2707     1.6410   6.868 9.95e-11 ***
## cambio        6.5864     0.6221  10.587  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.379 on 182 degrees of freedom
## Multiple R-squared:  0.3811, Adjusted R-squared:  0.3777
## F-statistic: 112.1 on 1 and 182 DF,  p-value: < 2.2e-16
```

```
dbmilho.ts <- ts(db1$pmilho, start = c(2004,8), end = c(2019,11), frequency = 12)
dbcambio.ts <- ts(db1$cambio, start = c(2004,8), end = c(2019,11), frequency = 12)


plot.ts(dbmilho.ts, type = "l", ylab="Preço da Saca de 60kg de Milho", xlab = "Tempo")
abline(h = mean(dbmilho$preco_milho))
```
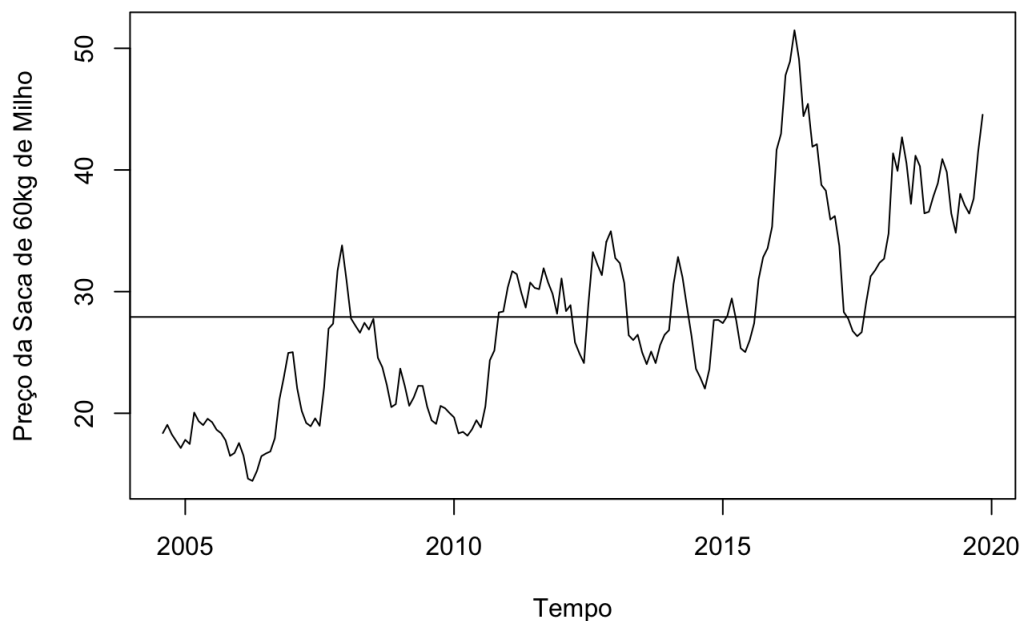


```
cambio.ts <- ts(dbcambio, start = c(2004, 8), end = c(2019, 11), frequency = 12)
plot.ts(cambio.ts, ylab="Taxa de câmbio R$/US$ - comercial – venda", xlab="Tempo")
abline(h = mean(cambio.ts))
```
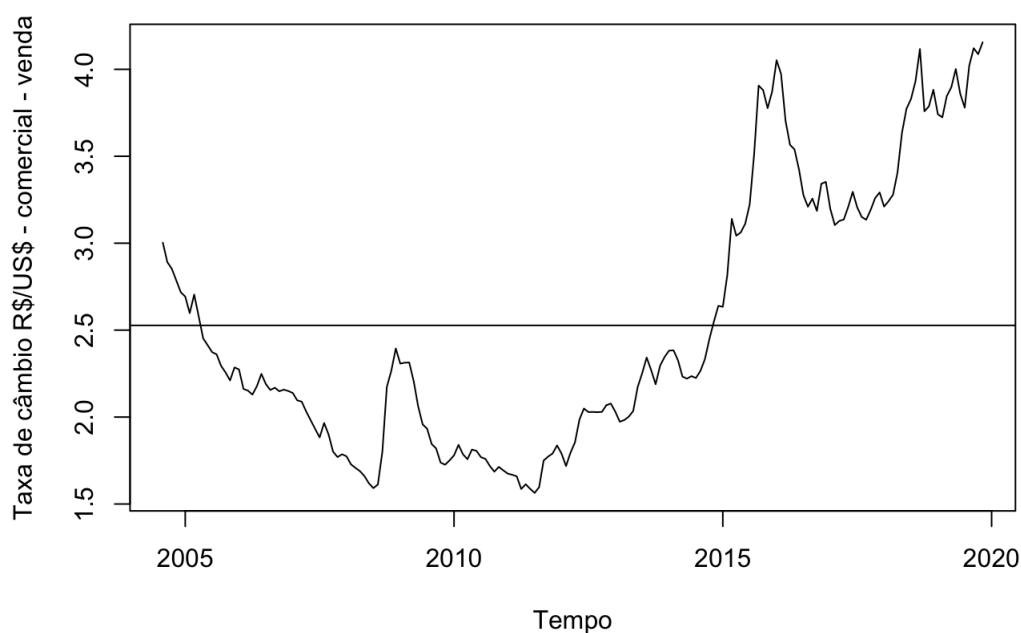


```
## Testes AR(1)
milho.ar <- ar(dbmilho.ts, aic=TRUE, method="ols", order.max = 1)
milho.ar
```

```
## 
## Call:
## ar(x = dbmilho.ts, aic = TRUE, order.max = 1, method = "ols")
## 
## Coefficients:
##      1
## 0.9751
## 
## Intercept: 0.1408 (0.1507)
## 
## Order selected 1  sigma^2 estimated as   4.158
```

```r
cambio.ar <- ar(dbcambio.ts, aic=TRUE, method="ols", order.max = 1)
cambio.ar
```
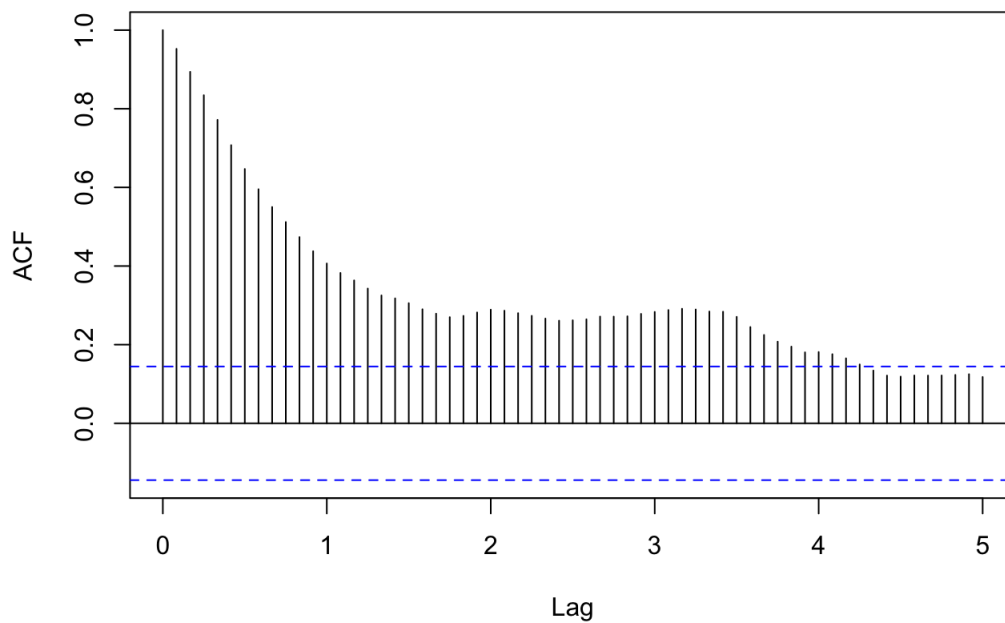
```
## 
## Call:
## ar(x = dbcambio.ts, aic = TRUE, order.max = 1, method = "ols")
## 
## Coefficients:
##      1
## 1.0033
## 
## Intercept: 0.006327 (0.007233)
## 
## Order selected 1  sigma^2 estimated as   0.009571
```

```r
acf(db1.ts) #pmilho = 20 ; cambio = 20
```
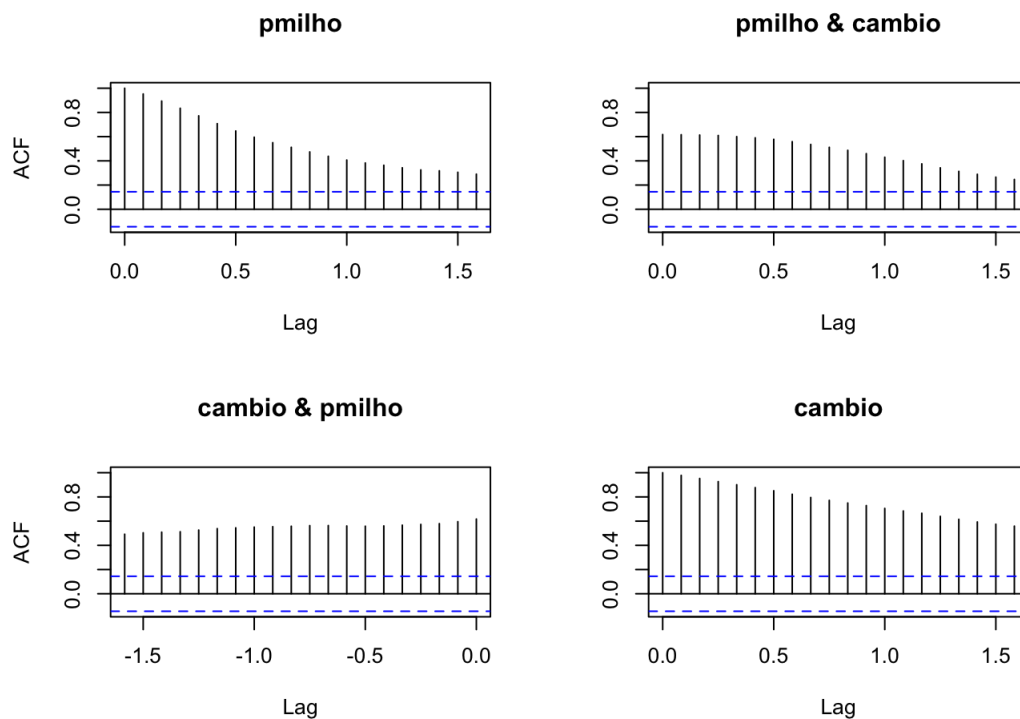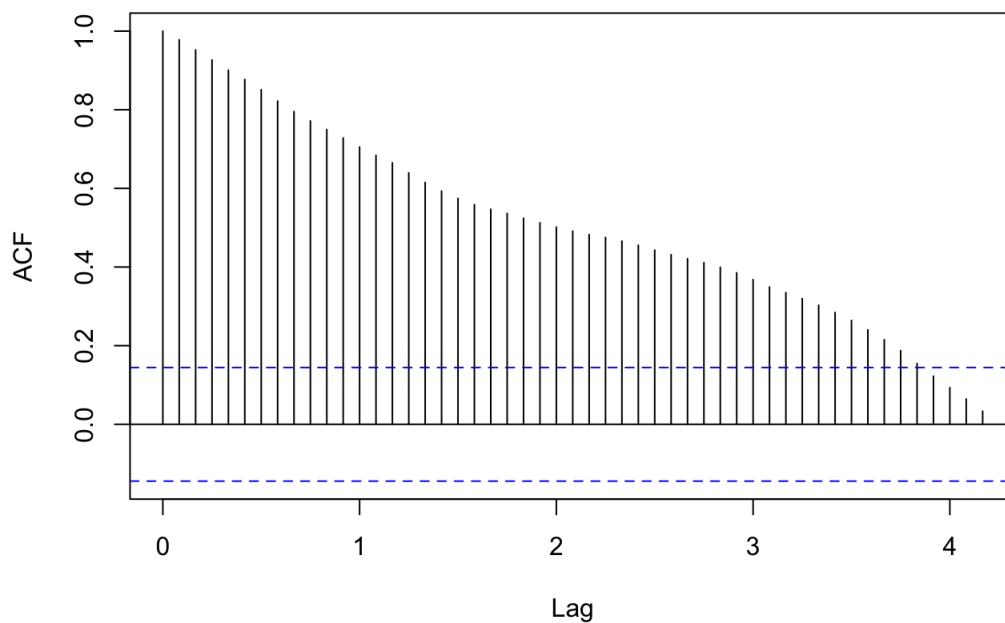


```r
acf(dbmilho.ts, lag.max = 60) # 50 lags para o teste
```

## Series dbmilho.ts



```r
acf(dbcambio.ts, lag.max = 50) # 46 lags para o teste
```

## Series dbcambio.ts



```r
library(tseries)
adf.test(dbmilho.ts, k=50)
```

```
## 
##  Augmented Dickey-Fuller Test
## 
## data:  dbmilho.ts
## Dickey-Fuller = -1.9549, Lag order = 50, p-value = 0.595
## alternative hypothesis: stationary
```

```r
adf.test(dbcambio.ts, k=46)
```

```
## 
##   Augmented Dickey-Fuller Test
## 
## data:  dbcambio.ts
## Dickey-Fuller = -1.1027, Lag order = 46, p-value = 0.9195
## alternative hypothesis: stationary
```

```r
# Identificar a ordem da integracao: funcao ndiff
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 3.5.2
```

```r
ndiffs(dbmilho.ts)
```

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

```r
ndiffs(dbcambio.ts)
```

```
## [1] 2
```

```r
# Obtendo as ordens integradas de cada db
milho.i1.ts <- diff(dbmilho.ts)
cambio.i2.ts <- diff(dbcambio.ts, differences = 2)
cambio.i1.ts <- diff(dbcambio.ts)
# Plotando antes e depois do dbmilho
split.screen(figs = c(2,1))
```

```
## [1] 1 2
```

```r
screen(1)
plot.ts(dbmilho.ts, main="BANCO DE DADOS ORIGINAL DO PREÇO DO MILHO", xlab="Tempo")
screen(2)
plot.ts(milho.i1.ts,  main="DIFERENÇAS DE PRIMEIRA ORDEM DO PREÇO DO MILHO", xlab="Tempo")
abline(h=0)

# Plotando antes e depois do dbcambio
split.screen(figs = c(2,1))
```

```
## [1] 3 4
```

```r
screen(1)
plot.ts(dbcambio.ts, main="BANCO DE DADOS ORIGINAL DO CÂMBIO", xlab="Tempo")
screen(2)
plot.ts(cambio.i2.ts,  main="DIFERENÇAS DE SEGUNDA ORDEM DO CÂMBIO", xlab="Tempo")
abline(h=0)

# Testando estacionariedade para ambas as séries
split.screen(figs = c(2,1))
```
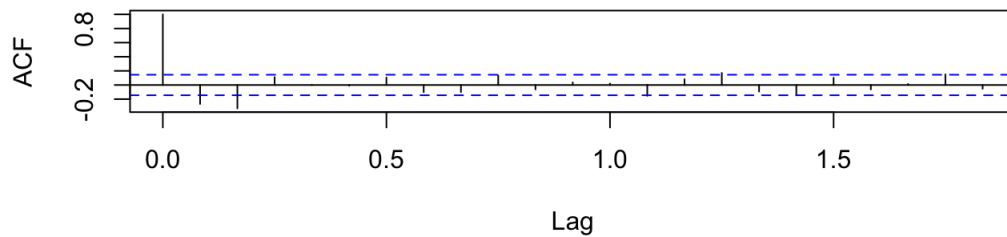
```
## [1] 5 6
```

```r
screen(1)
acf(milho.i1.ts, main="CORRELOGRAMA DAS DIFERENÇAS DE PRIMEIRA ORDEM DO PREÇO DO MILHO") # lag p/ teste = 1
screen(2)
acf(cambio.i2.ts, main="CORRELOGRAMA DAS DIFERENÇAS DE SEGUNDA ORDEM DO CÂMBIO") # lag p/ teste = 2
```

## RRELOGRAMA DAS DIFERENÇAS DE PRIMEIRA ORDEM DO PREÇO DO
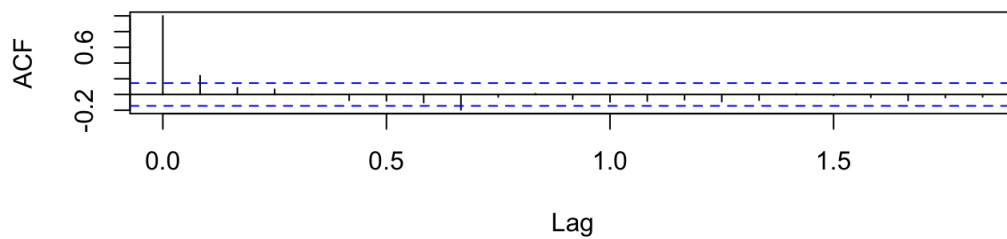


## CORRELOGRAMA DAS DIFERENÇAS DE SEGUNDA ORDEM DO CÂMB



```
adf.test(milho.i1.ts, k=1)
```

```
## Warning in adf.test(milho.i1.ts, k = 1): p-value smaller than printed p-
## value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  milho.i1.ts
## Dickey-Fuller = -7.8035, Lag order = 1, p-value = 0.01
## alternative hypothesis: stationary
```

```
adf.test(cambio.i2.ts, k=2)
```

```
## Warning in adf.test(cambio.i2.ts, k = 2): p-value smaller than printed p-
## value
```
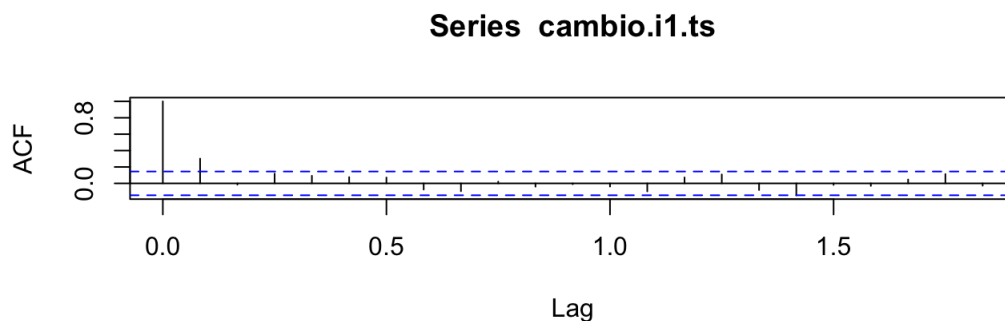
```
##
##  Augmented Dickey-Fuller Test
##
## data:  cambio.i2.ts
## Dickey-Fuller = -12.352, Lag order = 2, p-value = 0.01
## alternative hypothesis: stationary
```

```
mod2 <- dynlm(milho.i1.ts ~ cambio.i2.ts)
summary(mod2)
```

```
## 
## Time series regression with "ts" data:
## Start = 2004(10), End = 2019(11)
## 
## Call:
## dynlm(formula = milho.i1.ts ~ cambio.i2.ts)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.5767 -1.1475 -0.3063  1.1314  6.4602
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.1393     0.1530   0.911    0.364
## cambio.i2.ts   0.7540     1.3237   0.570    0.570
## 
## Residual standard error: 2.064 on 180 degrees of freedom
## Multiple R-squared:  0.0018, Adjusted R-squared:  -0.003746
## F-statistic: 0.3245 on 1 and 180 DF,  p-value: 0.5696
```

```
acf(cambio.i1.ts)
```

### Series  cambio.i1.ts



```
adf.test(cambio.i1.ts, k=1)
```

```
## Warning in adf.test(cambio.i1.ts, k = 1): p-value smaller than printed p-
## value
```

```
## 
##  Augmented Dickey-Fuller Test
## 
## data:  cambio.i1.ts
## Dickey-Fuller = -9.2732, Lag order = 1, p-value = 0.01
## alternative hypothesis: stationary
```

```
mod3 <- dynlm(milho.i1.ts ~ cambio.i1.ts)
summary(mod3)
```

```
##
## Time series regression with "ts" data:
## Start = 2004(9), End = 2019(11)
##
## Call:
## dynlm(formula = milho.i1.ts ~ cambio.i1.ts)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.5959 -1.2963 -0.2683  1.0828  6.3995
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.1306     0.1519   0.860    0.391
## cambio.i1.ts   1.9765     1.5491   1.276    0.204
##
## Residual standard error: 2.051 on 181 degrees of freedom
## Multiple R-squared:  0.008913,   Adjusted R-squared:  0.003438
## F-statistic: 1.628 on 1 and 181 DF,  p-value: 0.2036
```

```
dbx <- as.matrix(cbind(db1$pmilho,db1$cambio))
head(dbx,20)
```

```
##            [,1]   [,2]
##  [1,] 18.36727 3.0029
##  [2,] 19.04900 2.8911
##  [3,] 18.26600 2.8529
##  [4,] 17.70150 2.7860
##  [5,] 17.14429 2.7182
##  [6,] 17.81619 2.6930
##  [7,] 17.47333 2.5978
##  [8,] 20.05682 2.7047
##  [9,] 19.34700 2.5792
## [10,] 19.02429 2.4528
## [11,] 19.54909 2.4135
## [12,] 19.26429 2.3735
## [13,] 18.64348 2.3606
## [14,] 18.36095 2.2944
## [15,] 17.77650 2.2565
## [16,] 16.49700 2.2108
## [17,] 16.73524 2.2855
## [18,] 17.55409 2.2739
## [19,] 16.52278 2.1619
## [20,] 14.62261 2.1520
```

```
po.test(dbx) # H0: Nao haá cointegracao entre as series.
```

```
## Warning in po.test(dbx): p-value greater than printed p-value
```

```
##
##  Phillips-Ouliaris Cointegration Test
##
## data:  dbx
## Phillips-Ouliaris demeaned = -14.218, Truncation lag parameter =
## 1, p-value = 0.15
```

```
# Resultado: p-valor > 0,15 ----- Portanto, nao ha evidencias estatisticas para rejeitarmos H0 e assu-
# mirmos a HA de cointegracao.
```

```
db2 <- read.csv("db3_03.2011_10.2019.csv")
db2 <- read.csv2("db4.csv")
head(db2)
```

```
##       m1pp icred
## 1 108690.8 27.59
## 2 108508.9 28.18
## 3 107710.2 28.11
## 4 109017.8 27.85
## 5 110230.4 28.04
## 6 110625.9 27.46
```

```
db2.ts <- ts(db2, start = c(2011,3), end = c(2019,11), frequency = 12)
head(db2.ts)
```

```
##              m1pp icred
## Mar 2011 108690.8 27.59
## Apr 2011 108508.9 28.18
## May 2011 107710.2 28.11
## Jun 2011 109017.8 27.85
## Jul 2011 110230.4 28.04
## Aug 2011 110625.9 27.46
```

```
mod2 <- dynlm(icred ~ m1pp, data = db2.ts)
summary(mod2)
```

```
##
## Time series regression with "ts" data:
## Start = 2011(3), End = 2019(11)
##
## Call:
## dynlm(formula = icred ~ m1pp, data = db2.ts)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.3304 -2.6150 -0.8357  2.5367  6.8401
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.259e+01  1.900e+00  11.887   <2e-16 ***
## m1pp        2.270e-05  1.162e-05   1.953   0.0535 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.433 on 103 degrees of freedom
## Multiple R-squared:  0.03571,    Adjusted R-squared:  0.02635
## F-statistic: 3.815 on 1 and 103 DF,  p-value: 0.05352
```