

Modelo estimado utilizando R

Gabriel Petrini da Silveira

2019

Carregando pacotes

```
library(zoo)
library(xts)
library(tsDyn)
library(urca)
library(vars)
library(dplyr)
library(stargazer)
library(lmtest)
```

Carregando dados

```
df <- read.csv(
  "/dados/Dissertacao/Modelo/SeriesTemporais/Dados_yeojohnson.csv",
  encoding="UTF-8",
  stringsAsFactors=FALSE
)
df <- ts(data = df, start = c(1992,03), frequency = 4)
df <- as.xts(df)
df <- zoo::na.locf(df)
```

Teste de cointegração

```
vars::VARselect(
  y = df[,c("Inflação", "gZ")],
  type = "both",
  exogen = df[,c("Taxa.de.juros")],
  lag.max = 12
)$selection
```

```
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      5      1      1      5
```

Teste de Johansen

```
urca::ca.jo(
  x = df[,c("Inflação", "gZ")],
  type = "trace",
  ecdet = "trend",
  spec = "longrun",
  K = 4
) -> model.coimt
model.coimt %>% summary()
```

```
##
## #####
## # Johansen-Procedure #
## #####
##
## Test type: trace statistic , with linear trend in cointegration
##
## Eigenvalues (lambda):
## [1] 2.037177e-01 5.141294e-02 5.551115e-17
##
## Values of teststatistic and critical values of test:
##
##          test 10pct  5pct  1pct
## r <= 1 |   5.65 10.49 12.25 16.26
## r = 0  |  30.02 22.76 25.32 30.45
##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##          Inflação.l4      gZ.l4      trend.l4
## Inflação.l4  1.000000e+00  1.000000e+00  1.000000000
## gZ.l4        -1.300951e-01 -2.607295e-02  0.134768185
## trend.l4     -2.141512e-05  4.086791e-05 -0.008350887
##
## Weights W:
## (This is the loading matrix)
##
##          Inflação.l4      gZ.l4      trend.l4
## Inflação.d -0.04466473 -0.1321764  1.089563e-18
## gZ.d        2.54907017 -0.9143351  9.977913e-18
```

Seleção do rank

```
tsDyn::rank.select(
  data = df[,c("Inflação", "gZ")],
  lag.max = 12,
  include = "both"
) %>% summary()
```

```
## Best AIC: rank= 2 lag= 5
## Best BIC: rank= 2 lag= 1
```

```
## Best HQ : rank= 2 lag= 4
##
## Best number of lags:
##      r=0 r=1 r=2
## AIC    5   4   5
## BIC    1   1   1
## HQ     1   1   1
```

Engel-Granger

```
egcm::egcm(
  X = df[, c("Inflação")],
  Y = df[, c("gZ")],
  log = FALSE
) %>% egcm::is.cointegrated()
```

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

VECM

Estimação

```
tsDyn::VECM(
  data=df[,c("Inflação", "gZ")],
  include = "trend",
  LRinclude = "both",
  lag = 4,
  r = 1,
  exogen = df[,c("Taxa.de.juros")] %>% coredata(),
  estim = "ML"
) -> model
model %>% summary()

## #####
## ###Model VECM
## #####
## Full sample size: 111      End sample size: 106
## Number of variables: 2    Number of estimated slope parameters 20
## AIC -1738.982    BIC -1683.05    SSR 0.2685122
## Cointegrating vector (estimated by ML):
##      Inflação      gZ      const      trend
## r1          1 -0.1488119 0.2037247 -0.001104518
##
##
##          ECT          Inflação -1
## Equation Inflação 0.0539(0.0362) -0.0612(0.1105)
## Equation gZ      2.0613(0.3754)*** -0.5613(1.1450)
##          gZ -1          Inflação -2          gZ -2
```

```
## Equation Inflação 0.0037(0.0081) -0.2580(0.1087)* 0.0113(0.0080)
## Equation gZ 0.1066(0.0835) 0.2744(1.1266) 0.0927(0.0831)
## Inflação -3 gZ -3
## Equation Inflação -0.0135(0.1059) 0.0092(0.0082)
## Equation gZ 1.1275(1.0971) 0.1951(0.0853)*
## Inflação -4 gZ -4
## Equation Inflação 0.1612(0.1062) -9.9e-05(0.0084)
## Equation gZ -0.4527(1.1011) -0.2675(0.0866)**
## Taxa.de.juros
## Equation Inflação -0.1243(0.0835)
## Equation gZ -4.7855(0.8658)***
```

Pós Estimação

```
vars::serial.test(
  model.coint %>% vars::vec2var(),
  lags.pt=8,
  type='PT.asymptotic'
)
```

```
##
## Portmanteau Test (asymptotic)
##
## data: Residuals of VAR object model.coint %>% vars::vec2var()
## Chi-squared = 22.236, df = 18, p-value = 0.2217
```

```
vars::serial.test(
  model.coint %>% vars::vec2var(),
  lags.pt=8,
  type='PT.adjusted'
)
```

```
##
## Portmanteau Test (adjusted)
##
## data: Residuals of VAR object model.coint %>% vars::vec2var()
## Chi-squared = 23.284, df = 18, p-value = 0.1799
```

```
vars::serial.test(
  model.coint %>% vars::vec2var(),
  lags.pt=8,
  type='BG'
)
```

```
##
## Breusch-Godfrey LM test
##
## data: Residuals of VAR object model.coint %>% vars::vec2var()
## Chi-squared = 35.068, df = 20, p-value = 0.01974
```

```

normality.test(
  model.coint %>% vars::vec2var(),
  multivariate.only=FALSE
)

## `$`resids of Inflação`
##
## JB-Test (univariate)
##
## data: Residual of resids of Inflação equation
## Chi-squared = 31.49, df = 2, p-value = 1.452e-07
##
##
## `$`resids of gZ`
##
## JB-Test (univariate)
##
## data: Residual of resids of gZ equation
## Chi-squared = 6.3198, df = 2, p-value = 0.04243
##
##
## $JB
##
## JB-Test (multivariate)
##
## data: Residuals of VAR object model.coint %>% vars::vec2var()
## Chi-squared = 38.496, df = 4, p-value = 8.851e-08
##
##
## $Skewness
##
## Skewness only (multivariate)
##
## data: Residuals of VAR object model.coint %>% vars::vec2var()
## Chi-squared = 6.3533, df = 2, p-value = 0.04173
##
##
## $Kurtosis
##
## Kurtosis only (multivariate)
##
## data: Residuals of VAR object model.coint %>% vars::vec2var()
## Chi-squared = 32.143, df = 2, p-value = 1.048e-07

```

Função resposta ao impulso

Não-ortogonalizada

```

tsDyn::irf(
  model.coint %>% vars::vec2var(),

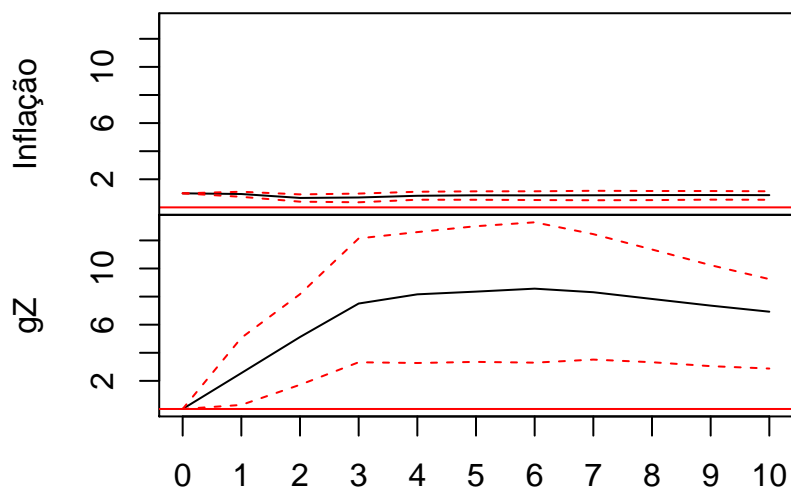
```

```

ortho=FALSE,
impulse="Inflação"
) %>% plot()

```

Impulse Response from Inflação



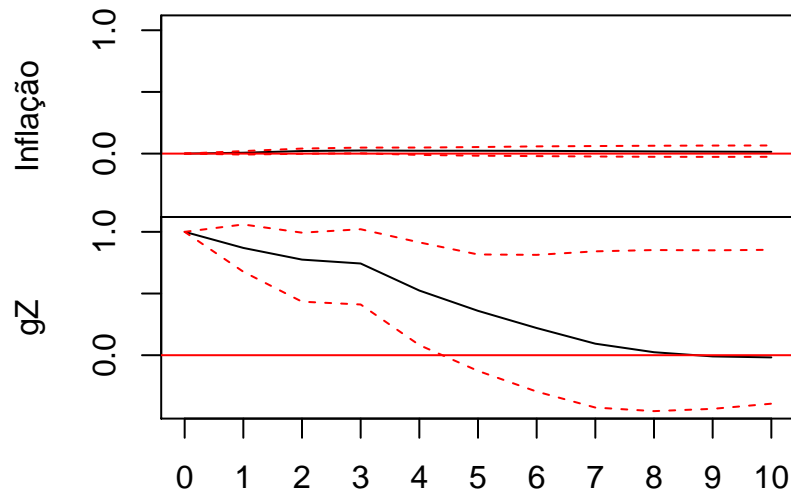
95 % Bootstrap CI, 100 runs

```

tsDyn::irf(
  model.cooint %>% vars::vec2var(),
  ortho=FALSE,
  impulse="gZ"
) %>% plot()

```

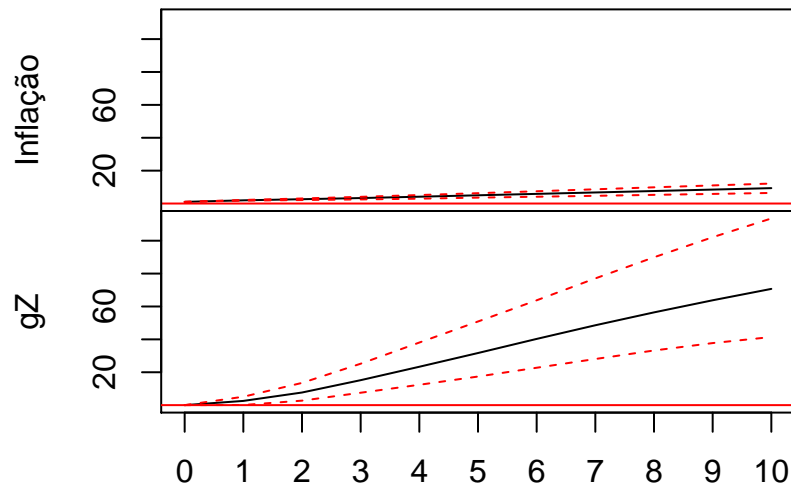
Impulse Response from gZ



95 % Bootstrap CI, 100 runs

```
tsDyn::irf(  
  model.coint %>% vars::vec2var(),  
  ortho=FALSE,  
  impulse="Inflação",  
  cumulative = TRUE  
) %>% plot()
```

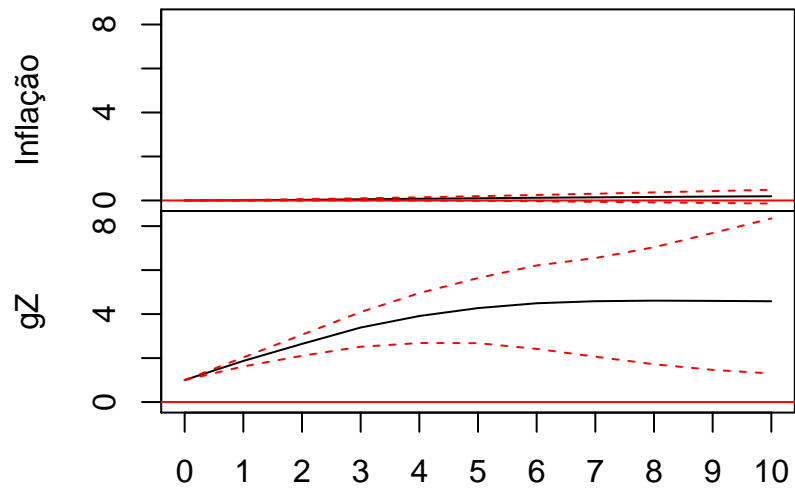
Impulse Response from Inflação (cumulative)



95 % Bootstrap CI, 100 runs

```
tsDyn::irf(  
  model.coint %>% vars::vec2var(),  
  ortho=FALSE,  
  impulse="gZ",  
  cumulative = TRUE  
) %>% plot()
```


Impulse Response from gZ (cumulative)



95 % Bootstrap CI, 100 runs