

Lista III

Alberson Miranda

2022-12-11

```
# configurações
knitr::opts_chunk$set(
  out.width = "100%"
)

# reproducibilidade
set.seed(1)

# pacotes
pacman::p_load(
  "ggplot2"
)

# dados
load("data/lista_III.RData")
```

1 Considere o banco de dados dadosSVAR. Este banco contém 3 séries temporais. Com base nessas informações faça o que se pede:

A Estime um VAR(p). Analise os resíduos e selecione a melhor ordem para o modelo. Escreva a equação do modelo indicando os elementos das matrizes.

```
# coercing to tsibble
data = data.frame(
  inflacao = inf,
  juros = juros,
  produto = y,
```

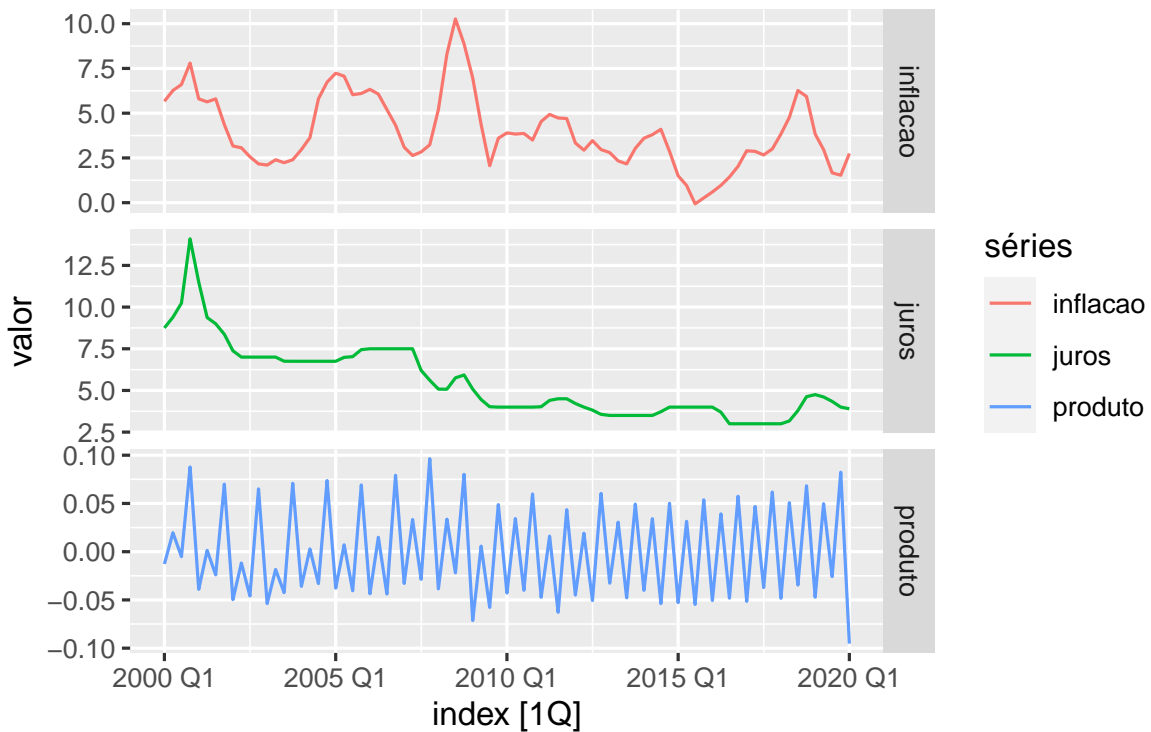
```

index = tsibble::yearquarter(
  seq(
    as.Date("2000-01-01"),
    by = "quarter",
    length.out = length(y)
  )
)
) |> tsibble::tsibble(index = index)

# plot série
data |>
tidyr::pivot_longer(
  -index,
  names_to = "séries",
  values_to = "valor") |>
fabletools::autoplot() +
facet_grid(vars(séries), scales = "free")

```

Plot variable not specified, automatically selected `vars = valor`



Para seleção da ordem do VAR(p), usarei os critérios de informação de Akaike corrigido (AICc) e o de Schwarz (BIC). Pelo princípio da parcimônia, o modelo escolhido será o de menor ordem dentre os que apresentarem ruído branco. O AICc apontou para um VAR(4), enquanto o BIC para um VAR(5).

```
# ajuste
fit = data |>
  fabletools::model(
    aicc = fable::VAR(vars(inflacao, juros, produto) ~ 1 + AR(p = 0:6), ic = "aicc"),
    bic = fable::VAR(vars(inflacao, juros, produto) ~ 1 + AR(p = 0:6), ic = "bic")
  )

# ordem
fit
```

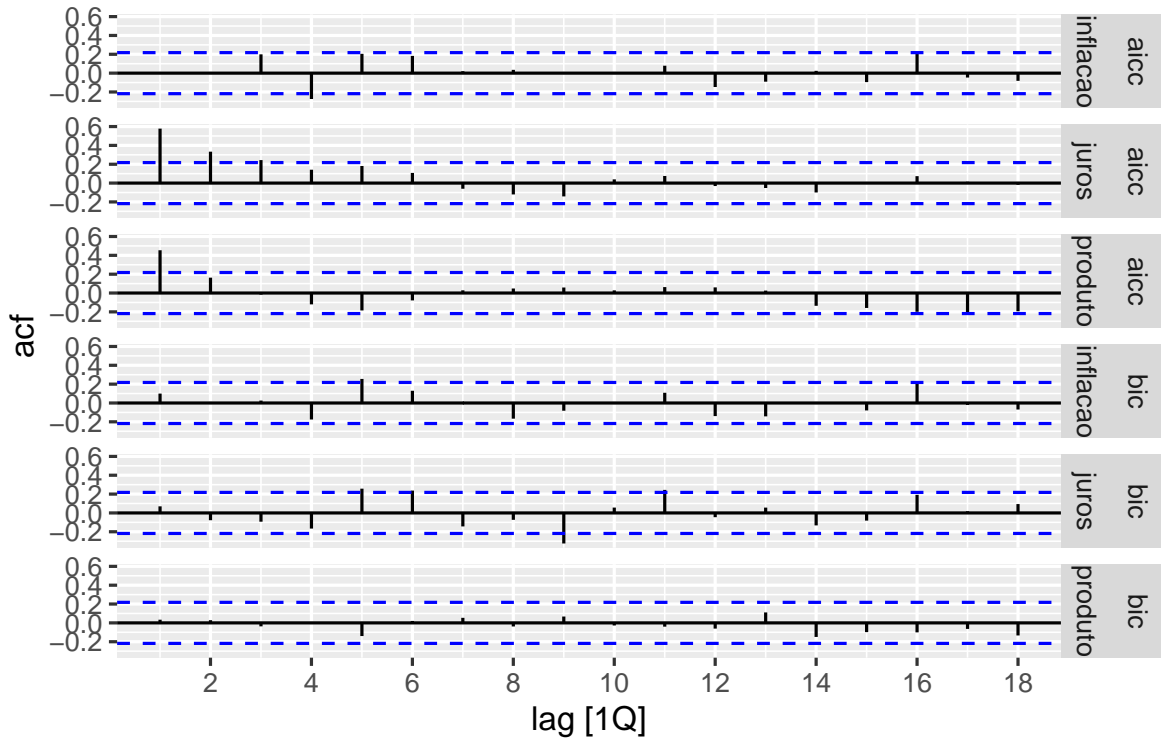
```
# A mable: 1 x 2
      aicc      bic
  <model>  <model>
1 <VAR(4) w/ mean> <VAR(5) w/ mean>
```

```
# critérios de informação
fabletools::glance(fit)
```

```
# A tibble: 2 x 6
  .model sigma2      log_lik  AIC  AICc  BIC
  <chr>  <list>      <dbl> <dbl> <dbl> <dbl>
1 aicc  <dbl [3 x 3]>    133. -170. -1.54 -57.0
2 bic   <dbl [3 x 3]>    179. -243. 124. -111.
```

Analisando os resíduos, a ACF do VAR(4) ainda demonstrou autocorrelação significativa, enquanto a do VAR(5) se mostrou ruído branco. Assim, selecionamos o VAR(5).

```
# ACF dos resíduos
fit |>
  fabletools::augment() |>
  feasts::ACF(.innov) |>
  autoplot()
```



```
# Portmanteau test
fabletools::augment(fit) |>
  subset(.model == "bic") |>
  fabletools::features(.innov, feasts::ljung_box, lag = 5)
```

```
# A tibble: 3 x 4
  .model .response lb_stat lb_pvalue
  <chr>   <fct>       <dbl>   <dbl>
1 bic    inflacao      8.82    0.117
2 bic    juros        9.38    0.0948
3 bic    produto       1.90    0.863
```

```
# coeficientes
fit |>
  subset(select = bic) |>
  fabletools::report()
```

Series: inflacao, juros, produto
Model: VAR(5) w/ mean

Coefficients for inflacao:

	lag(inflacao,1)	lag(juros,1)	lag(produto,1)	lag(inflacao,2)
	1.4346	-0.4282	6.2583	-0.7060
s.e.	0.1289	0.2955	8.5931	0.2218
	lag(juros,2)	lag(produto,2)	lag(inflacao,3)	lag(juros,3)
	0.4701	6.9992	0.2142	-0.1166
s.e.	0.3269	4.5723	0.2370	0.2673
	lag(produto,3)	lag(inflacao,4)	lag(juros,4)	lag(produto,4)
	5.6129	-0.4017	0.0960	4.2916
s.e.	4.4977	0.2187	0.2586	4.4633
	lag(inflacao,5)	lag(juros,5)	lag(produto,5)	constant
	0.2900	-0.0319	-1.1704	0.6477
s.e.	0.1277	0.1620	8.9462	0.3236

Coefficients for juros:

	lag(inflacao,1)	lag(juros,1)	lag(produto,1)	lag(inflacao,2)
	0.1373	1.4102	-1.3161	-0.1466
s.e.	0.0433	0.0993	2.8864	0.0745
	lag(juros,2)	lag(produto,2)	lag(inflacao,3)	lag(juros,3)
	-0.5880	-1.3960	0.0367	0.2679
s.e.	0.1098	1.5359	0.0796	0.0898
	lag(produto,3)	lag(inflacao,4)	lag(juros,4)	lag(produto,4)
	-1.4235	-0.0016	-0.2028	-0.2008
s.e.	1.5108	0.0735	0.0869	1.4992
	lag(inflacao,5)	lag(juros,5)	lag(produto,5)	constant
	0.0042	0.0687	-0.7161	0.0764
s.e.	0.0429	0.0544	3.0051	0.1087

Coefficients for produto:

	lag(inflacao,1)	lag(juros,1)	lag(produto,1)	lag(inflacao,2)
	0.0053	-0.0014	0.5377	-0.0066
s.e.	0.0018	0.0041	0.1192	0.0031
	lag(juros,2)	lag(produto,2)	lag(inflacao,3)	lag(juros,3)
	0.0053	-0.1374	-0.0010	-0.0020
s.e.	0.0045	0.0634	0.0033	0.0037
	lag(produto,3)	lag(inflacao,4)	lag(juros,4)	lag(produto,4)
	-0.2272	0.0049	-0.0029	0.8161
s.e.	0.0624	0.0030	0.0036	0.0619
	lag(inflacao,5)	lag(juros,5)	lag(produto,5)	constant
	-0.0018	0.0014	-0.6744	-0.0047
s.e.	0.0018	0.0022	0.1241	0.0045

Residual covariance matrix:

```

      inflacao   juros produto
inflacao  0.5813  0.0644 -0.0011
juros     0.0644  0.0656 -0.0003
produto   -0.0011 -0.0003  0.0001

```

log likelihood = 178.72

AIC = -243.45 AICc = 123.89 BIC = -110.59

A equação do modelo é:

$$\begin{bmatrix} Y_{A,t} \\ Y_{B,t} \\ Y_{C,t} \end{bmatrix} = \begin{bmatrix} \phi_{0A} \\ \phi_{0B} \\ \phi_{0C} \end{bmatrix} + \begin{bmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ \phi_{21} & \phi_{22} & \phi_{23} \\ \phi_{31} & \phi_{32} & \phi_{33} \end{bmatrix} \begin{bmatrix} Y_{A,t-1} \\ Y_{B,t-1} \\ Y_{C,t-1} \end{bmatrix} + \begin{bmatrix} \phi_{14} & \phi_{15} & \phi_{16} \\ \phi_{24} & \phi_{25} & \phi_{26} \\ \phi_{34} & \phi_{35} & \phi_{36} \end{bmatrix} \begin{bmatrix} Y_{A,t-2} \\ Y_{B,t-2} \\ Y_{C,t-2} \end{bmatrix} + \dots + \begin{bmatrix} \epsilon_{A,t} \\ \epsilon_{B,t} \\ \epsilon_{C,t} \end{bmatrix}$$

$$\begin{bmatrix} Y_{i,t} \\ Y_{j,t} \\ Y_{p,t} \end{bmatrix} = \begin{bmatrix} 0.6477 \\ 0.0764 \\ -0.0047 \end{bmatrix} + \begin{bmatrix} 1.4346 & -0.4282 & 6.2583 \\ 0.1373 & 1.4102 & -1.3161 \\ 0.0053 & -0.0014 & 0.5377 \end{bmatrix} \begin{bmatrix} Y_{i,t-1} \\ Y_{j,t-1} \\ Y_{p,t-1} \end{bmatrix} + \begin{bmatrix} -0.7060 & 0.4701 & 6.9992 \\ -0.1466 & -0.5880 & -1.3960 \\ -0.0066 & 0.0053 & -0.1374 \end{bmatrix} \begin{bmatrix} Y_{i,t-2} \\ Y_{j,t-2} \\ Y_{p,t-2} \end{bmatrix} + \dots + \begin{bmatrix} 0.5813 & 0.0644 & -0.0011 \\ 0.0644 & 0.0656 & -0.0003 \\ -0.0011 & -0.0003 & 0.0001 \end{bmatrix} \begin{bmatrix} Y_{i,t-3} \\ Y_{j,t-3} \\ Y_{p,t-3} \end{bmatrix} + \dots$$

B Faça o teste de causalidade de Granger para todas as variáveis.

1. Juros x Inflação

A 5% de significância, não há evidências para rejeição da hipótese nula de que a inflação não Granger-causa os juros (e vice-versa).

```
# juros x inflação
lmtest::grangertest(juros ~ inflacao, order = 5, data = data)
```

Granger causality test

Model 1: juros ~ Lags(juros, 1:5) + Lags(inflacao, 1:5)

Model 2: juros ~ Lags(juros, 1:5)

	Res.Df	Df	F	Pr(>F)
1	65			
2	70	-5	2.1235	0.07365 .

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
lmtest::grangertest(inflacao ~ juros, order = 5, data = data)
```

Granger causality test

Model 1: inflacao ~ Lags(inflacao, 1:5) + Lags(juros, 1:5)

Model 2: inflacao ~ Lags(inflacao, 1:5)

	Res.Df	Df	F	Pr(>F)
1	65			
2	70	-5	1.6342	0.1635

2. Inflação x Produto

A 5% de significância, não há evidências para rejeição da hipótese nula de que o produto não Granger-causa a inflação (e vice-versa).

```
# produto x inflação
lmtest::grangertest(inflacao ~ produto, order = 5, data = data)
```

Granger causality test

Model 1: inflacao ~ Lags(inflacao, 1:5) + Lags(produto, 1:5)

Model 2: inflacao ~ Lags(inflacao, 1:5)

	Res.Df	Df	F	Pr(>F)
1	65			
2	70	-5	1.7257	0.1412

```
lmtest::grangertest(produto ~ inflacao, order = 5, data = data)
```

Granger causality test

Model 1: produto ~ Lags(produto, 1:5) + Lags(inflacao, 1:5)

Model 2: produto ~ Lags(produto, 1:5)

	Res.Df	Df	F	Pr(>F)
1	65			
2	70	-5	1.7717	0.1311

3. Produto x Juros

A 5% de significância, não há evidências para rejeição da hipótese nula de que o produto não Granger-causa os juros (e vice-versa).

```
# produto x juros
lmtest::grangertest(juros ~ produto, order = 5, data = data)
```

Granger causality test

Model 1: juros ~ Lags(juros, 1:5) + Lags(produto, 1:5)

Model 2: juros ~ Lags(juros, 1:5)

	Res.Df	Df	F	Pr(>F)
1	65			
2	70	-5	1.5902	0.1753

```
lmtest::grangertest(produto ~ juros, order = 5, data = data)
```

Granger causality test

Model 1: produto ~ Lags(produto, 1:5) + Lags(juros, 1:5)

Model 2: produto ~ Lags(produto, 1:5)

	Res.Df	Df	F	Pr(>F)
1	65			
2	70	-5	0.093	0.9931

C Estime um SVAR considerando o modelo da letra A e as seguintes restrições para as relações contemporâneas: (i) y afeta inf e juros; (ii) inf afeta apenas juros porém não impacta y; (iii) juros não impacta y ou inf.


```

# wrangling
data = cbind(inf, juros, y)

# restrições
restricoes = diag(3)
restricoes[3, 1] = NA
restricoes[1, 2] = NA
restricoes[3, 2] = NA

# ajuste
fit = vars::VAR(data, p = 5)
fit_s = vars::SVAR(fit, Amat = restricoes, estmethod = "direct")

# coeficientes
summary(fit_s)

```

SVAR Estimation Results:

=====

Call:

```
vars::SVAR(x = fit, estmethod = "direct", Amat = restricoes)
```

Type: A-model

Sample size: 76

Log Likelihood: -323.518

Method: direct

Number of iterations: 158

Convergence code: 0

LR overidentification test:

LR overidentification

data: data

Chi^2 = 951, df = 3, p-value <2e-16

Estimated A matrix:

	inf	juros	y
inf	1.000000	-0.982810	0

```
juros 0.000000 1.000000 0
y      0.001883 0.001882 1
```

Estimated B matrix:

```
      inf juros y
inf    1      0 0
juros  0      1 0
y       0      0 1
```

Covariance matrix of reduced form residuals (*100):

```
      inf      juros      y
inf  196.5915  98.2810 -0.5551
juros 98.2810 100.0000 -0.3732
y     -0.5551 -0.3732 100.0017
```

E os demais coeficientes?

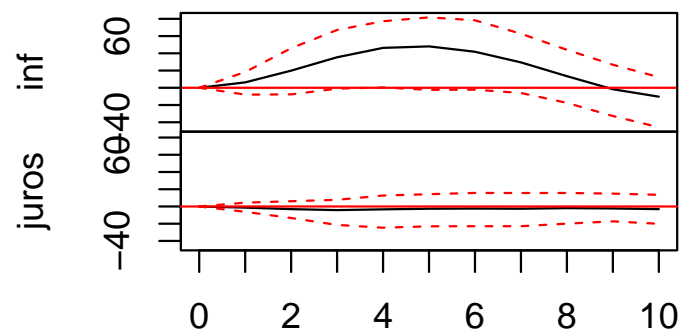
$$\begin{bmatrix} 1 & -0.982810 & 0 \\ 0 & 1 & 0 \\ 0.001883 & 0.001882 & 1 \end{bmatrix} \begin{bmatrix} Y_{i,t} \\ Y_{j,t} \\ Y_{p,t} \end{bmatrix}$$

D Obtenha e analise a função impulso resposta e a decomposição da variância do erro de previsão. Você deve utilizar o VAR(p) ou o SVAR para essa análise? Justifique sua resposta.

Deve-se usar o VAR estrutural para que os parâmetros estruturais sejam identificáveis. Do contrário, o problema da endogeneidade tornaria impossível atribuir a resposta a uma variável.

```
par(mai = c(0,0,0,0))
# função impulso-resposta: produto
plot(
  vars::irf(
    fit_s,
    impulse = "y",
    response = c("inf", "juros"),
    n.ahead = 10,
    ortho = FALSE,
    cumulative = FALSE
  )
)
```

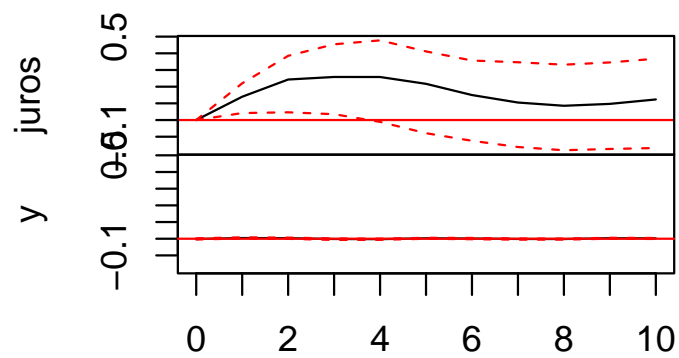
SVAR Impulse Response from y



95 % Bootstrap CI, 100 runs

```
# função impulso-resposta: inflação
plot(
  vars::irf(
    fit_s,
    impulse = "inf",
    response = c("y", "juros"),
    n.ahead = 10,
    ortho = FALSE,
    cumulative = FALSE
  )
)
```

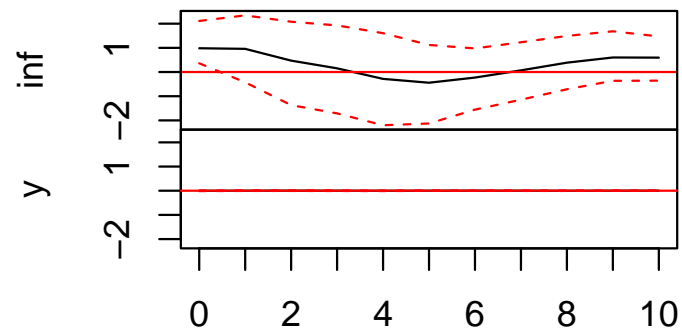
SVAR Impulse Response from inf



95 % Bootstrap CI, 100 runs

```
# função impulso-resposta: juros
plot(
  vars::irf(
    fit_s,
    impulse = "juros",
    response = c("y", "inf"),
    n.ahead = 10,
    ortho = FALSE,
    cumulative = FALSE
  )
)
```

SVAR Impulse Response from juros



95 % Bootstrap CI, 100 runs

```
# decomposição da variância  
plot(  
  vars::fevd(fit_s, n.ahead = 10)  
)
```

