

Homework2 (Corrigido)'

Davi Jorge

Atividade 2

Questão 1

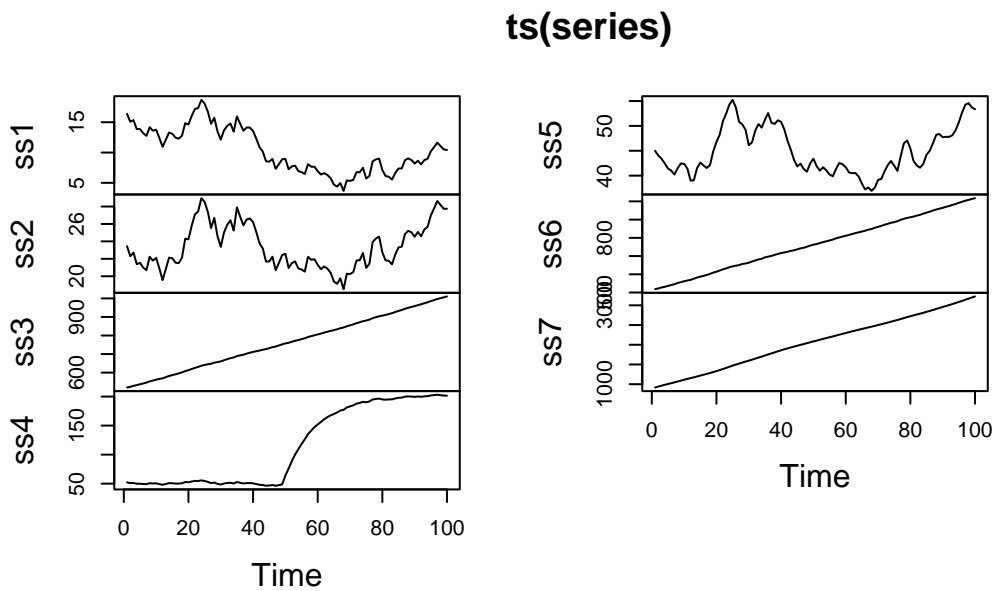
De posse das séries disponibilizadas no drive do curso, implemente testes de RU e verifique a estacionariedade das séries. Os dados estão em formato txt, csv e xls. Indique se as séries são estacionárias ou não. Aplique os testes ADF, PP, KPSS, Zivot-Andrews, e outros que julgar necessário. Você deve apresentar os códigos no apêndice e explicar (se perguntado) sobre os seus resultados e códigos.

Resposta

Gráfico das séries

```
# Libraries -----  
library(tseries)  
library(aTSA)  
library(dplyr)  
library(urca)  
library(ipeadata)  
library(gt)  
  
# Question 1 -----  
  
# Read datasets -----  
series <- read.table('data/data_exercicio2.txt') %>%  
  .[, -1]  
  
# Plot series -----
```

```
plot(ts(series))
```



Teste Dickey Fuller

```
# Dickey Fuller

dk_resultado = dplyr::tibble(serie = NULL, p_value = NULL)

for (i in 1:length(series)) {

  dk_test = tseries::adf.test(series[,i])
  res = dplyr::tibble(serie = i,
                     p_value = dk_test$p.value)

  dk_resultado = dk_resultado %>%
    rbind(res)

}
```

Teste Phillips-Perron

Resultados do Teste Dickey Fuller

P-values por série

Série	P-valor
1	0.788
2	0.788
3	0.788
4	0.788
5	0.788
6	0.788
7	0.788

A função ela define os testes como:

- Type 1: Sem drift e sem trend;
- Type 2: Com drift e sem trend;
- Type 3: Com ambos.

```
pp_resultado = dplyr::tibble(serie = NULL, p_value = NULL)

for (i in 1:length(series)) {

  pp_test = aTSA::pp.test(series[,i])

  res = dplyr::tibble(serie = rep(i, 3),
                     type = rownames(pp_test),
                     p_value = pp_test[,3])

  pp_resultado = pp_resultado %>%
    rbind(res) %>%
    dplyr::group_by(serie) %>%
    dplyr::arrange(p_value) %>%
    dplyr::slice(1)

}
```

Teste KPSS

As séries foram testadas para estacionariedade em nível e tendência estacionária.

Teste Phillips-Perron

Série	Tipo	P-valor
1	type 2	0.449
2	type 2	0.331
3	type 3	0.686
4	type 1	0.907
5	type 2	0.420
6	type 3	0.625
7	type 1	0.910

Teste KPSS

Série	P-valor (Tendência)	P-valor (Intercepto)
1	0.010	0.01
2	0.010	0.10
3	0.010	0.01
4	0.010	0.01
5	0.010	0.10
6	0.010	0.01
7	0.010	0.01

```
# KPSS

KPSS_resultado = dplyr::tibble(serie = NULL, p_value_T = NULL, p_value_I = NULL)

for (i in 1:length(series)) {

  KPSS_test = tseries::kpss.test(ts(series[,i]), null = 'Trend')
  KPSS_test2 = tseries::kpss.test(ts(series[,i]))

  res = dplyr::tibble(serie = i,
                     p_value_T = KPSS_test$p.value,
                     p_value_I = KPSS_test2$p.value)

  KPSS_resultado = KPSS_resultado %>%
    rbind(res)

}
```

Teste Zivot-Andrews

Série	Estatística teste	5%	10%	Break Point
1	-14.06	-5.08	-4.82	49
2	-14.06	-5.08	-4.82	49
3	-14.06	-5.08	-4.82	49
4	-14.06	-5.08	-4.82	49
5	-14.06	-5.08	-4.82	49
6	-14.06	-5.08	-4.82	49
7	-14.06	-5.08	-4.82	49

Teste Zivot-Andrews

```
# ZA

ZA_resultado = dplyr::tibble(serie = NULL, p_value = NULL)

for (i in 1:length(series)) {

  ZA_test = urca::ur.za(ts(series[,4]), model = 'both') %>%
    urca::summary()
  res = dplyr::tibble(serie = i,
                     teststat = round(ZA_test@teststat, 2),
                     cval_5 = ZA_test@cval[2],
                     cval_10 = ZA_test@cval[3],
                     bpoint = ZA_test@bpoint)

  ZA_resultado = ZA_resultado %>%
    rbind(res)

}
```

Questão 2

Aplique também os testes de RU nas séries do Produto interno bruto (PIB) a preços de mercado: índice encadeado (média 1995 = 100) e IPCA - geral - índice (dez. 1993 = 100), que você obteve no site do IPEADATA.

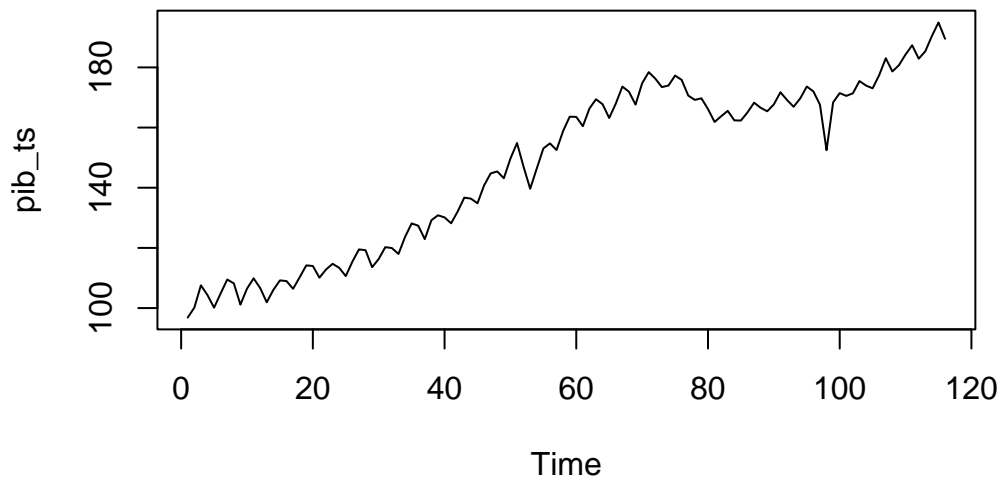
Resposta

Buscando as séries

```
# Get Series -----  
  
pib <- ipeadata('SCN104_PIBPM104')  
pib_ts <- ts(pib$value)  
  
ipca <- ipeadata('PRECOS12_IPCA12')  
ipca_ts <- ts(ipca$value)
```

Testando para o PIB

```
# PIB -----  
  
plot(pib_ts)
```



```
tseries::adf.test(pib_ts)
```

Augmented Dickey-Fuller Test

```
data: pib_ts
Dickey-Fuller = -1.6771, Lag order = 4, p-value = 0.7106
alternative hypothesis: stationary
```

```
aTSA::pp.test(pib_ts)
```

Phillips-Perron Unit Root Test
alternative: stationary

Type 1: no drift no trend

lag	Z_rho	p.value
4	0.571	0.816

Type 2: with drift no trend

lag	Z_rho	p.value
4	-1.15	0.864

Type 3: with drift and trend

lag	Z_rho	p.value
4	-9.61	0.47

Note: p-value = 0.01 means p-value <= 0.01

```
tseries::kpss.test(pib_ts, null = 'Trend')
```

KPSS Test for Trend Stationarity

```
data: pib_ts
KPSS Trend = 0.38699, Truncation lag parameter = 4, p-value = 0.01
```

```
tseries::kpss.test(pib_ts)
```

KPSS Test for Level Stationarity

```
data: pib_ts
KPSS Level = 2.275, Truncation lag parameter = 4, p-value = 0.01
```

```
urca::ur.za(pib_ts, model = 'both') %>%
  urca::summary()
```

```
#####
# Zivot-Andrews Unit Root Test #
#####
```

```
Call:
lm(formula = testmat)
```

```
Residuals:
      Min       1Q   Median       3Q      Max
-17.3468  -2.6062   0.3021   2.4399   8.7834
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  29.50734     5.97112   4.942 2.79e-06 ***
y.l1         0.68103     0.06567  10.370 < 2e-16 ***
trend        0.36214     0.07642   4.739 6.47e-06 ***
du          -6.82595     1.78363  -3.827 0.000216 ***
dt          -0.11318     0.06879  -1.645 0.102768
---

```

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

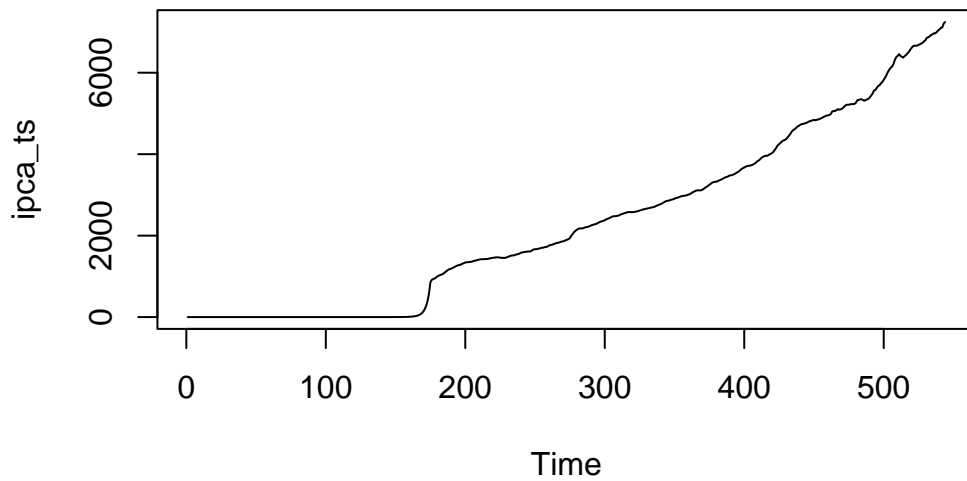
```
Residual standard error: 3.9 on 110 degrees of freedom
(1 observation deleted due to missingness)
Multiple R-squared:  0.9809,    Adjusted R-squared:  0.9802
F-statistic: 1414 on 4 and 110 DF,  p-value: < 2.2e-16
```

```
Teststatistic: -4.8568
Critical values: 0.01= -5.57 0.05= -5.08 0.1= -4.82
```

```
Potential break point at position: 76
```

Testando para o IPCA


```
# IPCA -----  
  
plot(ipca_ts)
```



```
tseries::adf.test(ipca_ts)
```

Augmented Dickey-Fuller Test

```
data: ipca_ts  
Dickey-Fuller = -0.89031, Lag order = 8, p-value = 0.9535  
alternative hypothesis: stationary
```

```
aTSA::pp.test(ipca_ts)
```

Phillips-Perron Unit Root Test
alternative: stationary

Type 1: no drift no trend

```

lag Z_rho p.value
  6  2.59    0.99
-----
Type 2: with drift no trend
lag Z_rho p.value
  6  1.97    0.99
-----
Type 3: with drift and trend
lag Z_rho p.value
  6 -1.04   0.987
-----
Note: p-value = 0.01 means p.value <= 0.01

```

```
tseries::kpss.test(ipca_ts, 'Trend')
```

KPSS Test for Trend Stationarity

```

data: ipca_ts
KPSS Trend = 1.4315, Truncation lag parameter = 6, p-value = 0.01

```

```
tseries::kpss.test(ipca_ts)
```

KPSS Test for Level Stationarity

```

data: ipca_ts
KPSS Level = 7.4957, Truncation lag parameter = 6, p-value = 0.01

```

```

urca::ur.za(ipca_ts, model = "both") %>%
  urca::summary()

```

```

#####
# Zivot-Andrews Unit Root Test #
#####

```

```

Call:
lm(formula = testmat)

Residuals:
    Min       1Q   Median       3Q      Max
-78.960  -7.665  -2.539   3.622  265.206

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -5.044875    2.827543  -1.784   0.0750 .
y.l1         0.994340    0.002096  474.370 < 2e-16 ***
trend        0.108174    0.025670   4.214 2.94e-05 ***
du           28.924668    5.780793   5.004 7.63e-07 ***
dt           -0.333284    0.169795  -1.963   0.0502 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 19.35 on 538 degrees of freedom
(1 observation deleted due to missingness)
Multiple R-squared:  0.9999,    Adjusted R-squared:  0.9999
F-statistic: 1.668e+06 on 4 and 538 DF,  p-value: < 2.2e-16

Teststatistic: -2.7002
Critical values: 0.01= -5.57 0.05= -5.08 0.1= -4.82

Potential break point at position: 489

```

Corrigindo Series da Questão 1

Para considerarmos uma série estaionária, precisamos que apenas um teste de raiz unitária ou de estacionariedade aponte que a série é estacionária. Para esse caso, temos no teste KPSS que as séries 2 e 5 podem ser estacionárias em nível. Já o teste Zivot-Andrews sinalizou que a série 4 é estacionária com quebra na observação 49.

Como nenhuma das outras séries possuem tendência estacionária, podemos diferencia-las sem penalização.

```

series_diff <- series %>%
  .[, -c(2,4,5)] %>%
  sapply(diff) %>%
  ts()

```

Resultados do Teste Dickey Fuller

P-values por série

Série	P-valor	Diferenciações
ss1	0.010	1
ss3	0.010	1
ss6	0.010	1
ss7	0.010	2

```
dk_resultado = dplyr::tibble(serie = NULL, diff = NULL, p_value = NULL)

for (i in 1:4) {

  dk_test = tseries::adf.test(series_diff[,i])

  if (dk_test$p.value > 0.1) {

    dk_test = tseries::adf.test(diff(series_diff[,i]))

    res = dplyr::tibble(serie = colnames(series_diff)[i],
                        p_value = dk_test$p.value,
                        diff = 2)

  }else{

    res = dplyr::tibble(serie = colnames(series_diff)[i],
                        p_value = dk_test$p.value,
                        diff = 1)

  }

  dk_resultado = dk_resultado %>%
    rbind(res)

}
```

Portanto, bastou uma diferenciação para que as series se tornassem estacionárias

```
series_diff[,4] <- c(0, diff(series_diff[,4]))

plot(series_diff)
```

