Análisis macro

Juan Alvaro Díaz Raimond Kedilhac

Setup

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.6
## v forcats   1.0.1     v stringr   1.6.0
## v ggplot2   4.0.1     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.2
## v purrr     1.2.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
library(urca)
library(vars)
```

```
## Loading required package: MASS
##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
##
##     select
##
## Loading required package: strucchange
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
##
## Loading required package: sandwich
##
## Attaching package: 'strucchange'
##
```

```
## The following object is masked from 'package:stringr':
##
##     boundary
##
## Loading required package: lmtest
```

```r
library(lmtest)
library(sandwich)
library(kableExtra)
```

```
##
## Attaching package: 'kableExtra'
##
## The following object is masked from 'package:dplyr':
##
##     group_rows
```

Data load (long format)

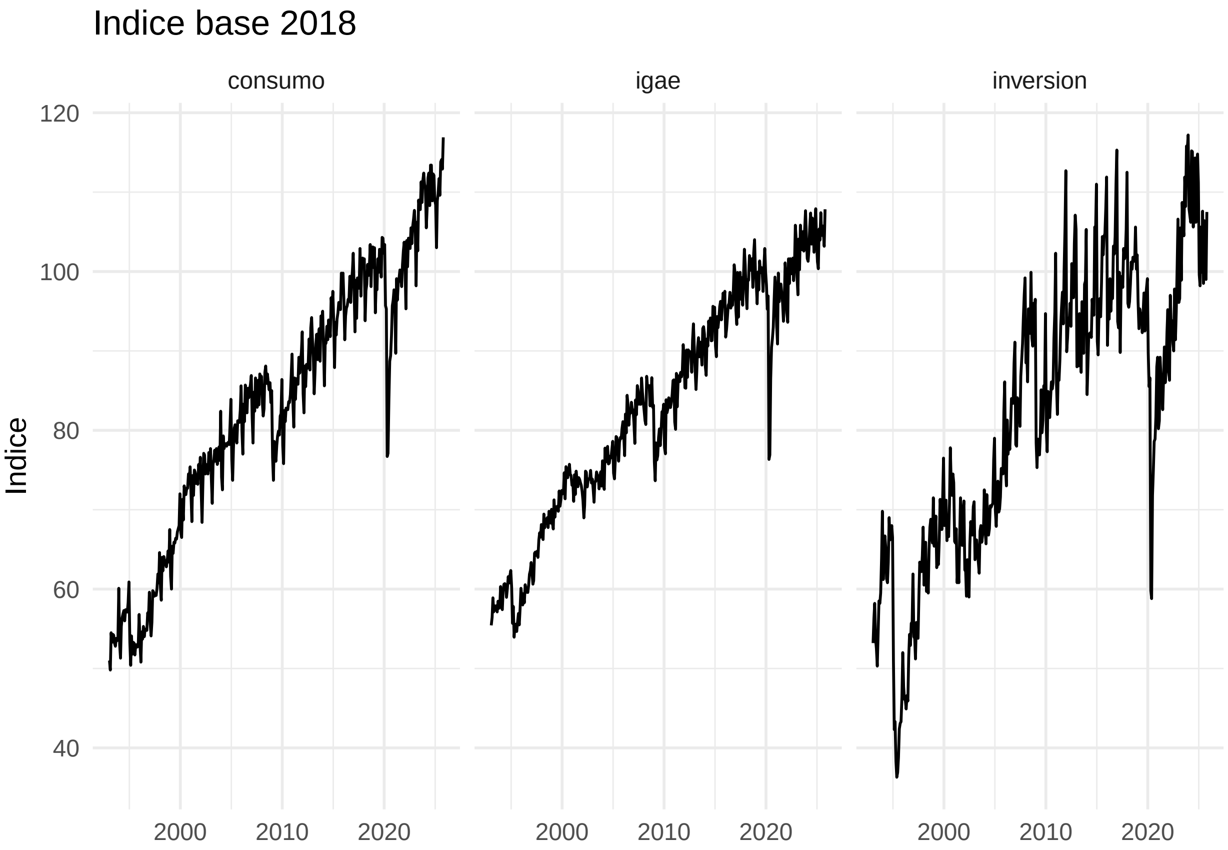
```r
df <- read_csv("data/series.csv") %>%
  arrange(unique_id, ds)
```

```
## Rows: 1182 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): unique_id
## dbl  (1): y
## date (1): ds
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Exploratory plots

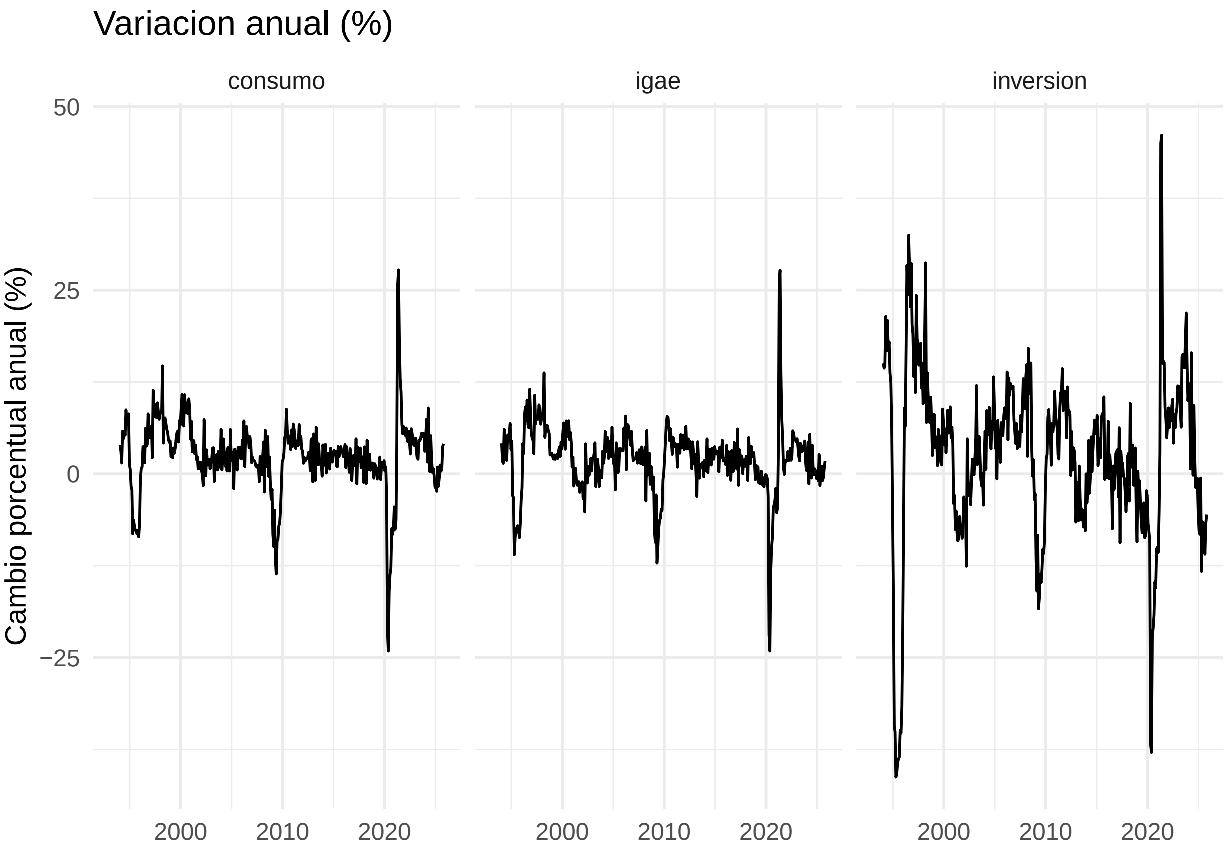
```r
plot_facets <- function(df_long, value_col, ylabel, title) {
  ggplot(df_long, aes(x = ds, y = .data[[value_col]])) +
    geom_line() +
    facet_wrap(~unique_id) +
    labs(title = title, x = NULL, y = ylabel) +
    theme_minimal(base_size = 11)
}
```

```r
plot_facets(df, "y", "Indice", "Indice base 2018")
```



```r
df <- df %>%
  group_by(unique_id) %>%
  mutate(yoy = (y / lag(y, 12) - 1) * 100) %>%
  ungroup()
```

```r
plot_facets(df, "yoy", "Cambio porcentual anual (%)", "Variacion anual (%)")
```



Stationarity tests

```
adf_test <- function(series) {
  test <- adf.test(na.omit(series))
  tibble(
    adf_stat = unname(test$statistic),
    p_value = test$p.value,
    n_lags = unname(test$parameter)
  )
}

kpss_test <- function(series, null = "Trend") {
  test <- kpss.test(na.omit(series), null = null)
  tibble(
    kpss_stat = unname(test$statistic),
    kpss_p = test$p.value,
    kpss_lags = unname(test$parameter)
  )
}
```

```
adf_df <- df %>%
  group_by(unique_id) %>%
  group_modify(~adf_test(.x$y)) %>%
  ungroup()

kpss_df <- df %>%
  group_by(unique_id) %>%
  group_modify(~kpss_test(.x$y, null = "Trend")) %>%
  ungroup()
```

```
adf_df %>% kable()
```

unique_id	adf_stat	p_value	n_lags
consumo	-3.528909	0.0399396	7
igae	-3.647127	0.0285724	7
inversion	-3.030894	0.1418548	7

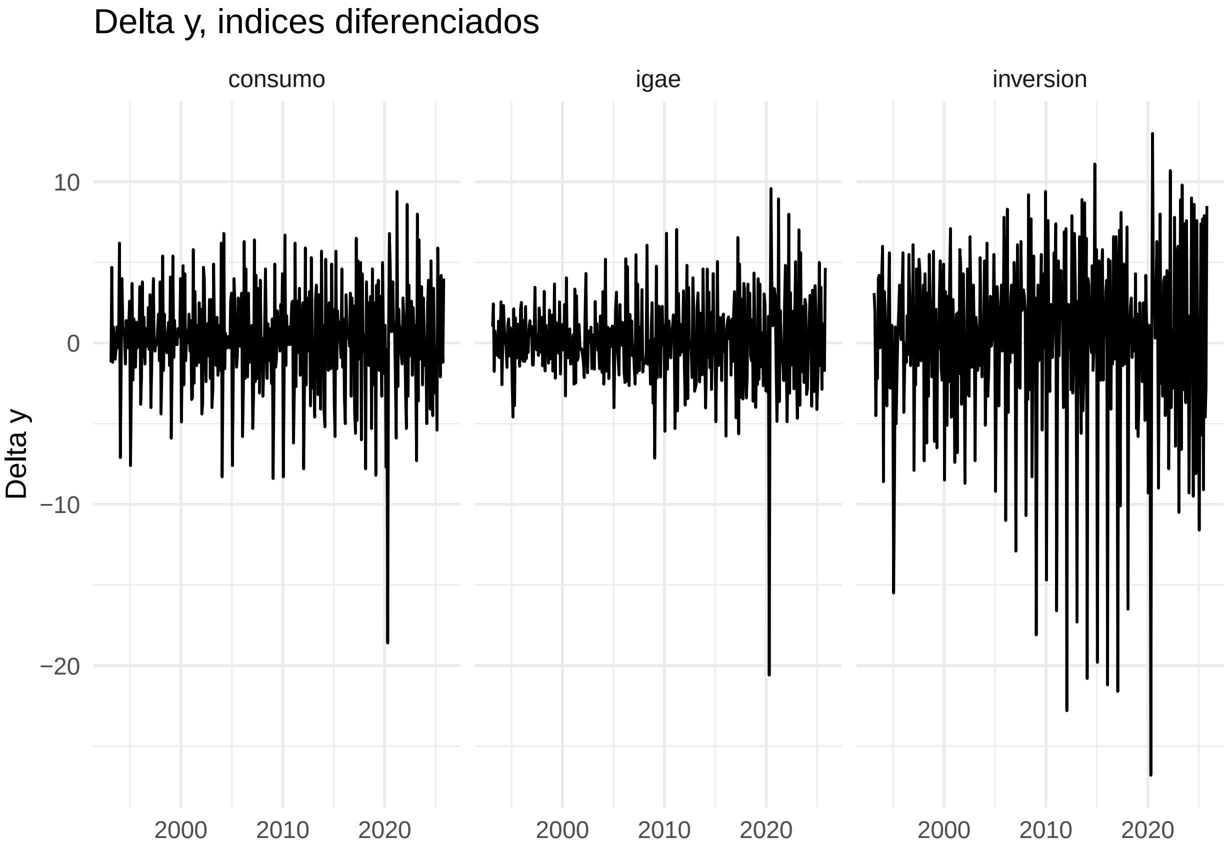
```
kpss_df %>% kable()
```

unique_id	kpss_stat	kpss_p	kpss_lags
consumo	0.5011787	0.01	5
igae	0.3544670	0.01	5
inversion	0.4684660	0.01	5

Differences

```
df <- df %>%
  group_by(unique_id) %>%
  mutate(dy = y - lag(y)) %>%
  ungroup()
```

```
plot_facets(df, "dy", "Delta y", "Delta y, indices diferenciados")
```



```
adf_dy <- df %>%
  group_by(unique_id) %>%
  group_modify(~adf_test(.x$dy)) %>%
  ungroup()

kpss_dy <- df %>%
  group_by(unique_id) %>%
  group_modify(~kpss_test(.x$dy, null = "Level")) %>%
  ungroup()
```

```
adf_dy %>% kable()
```

unique_id	adf_stat	p_value	n_lags
consumo	-9.637344	0.01	7
igae	-9.067185	0.01	7
inversion	-9.892597	0.01	7

```
kpss_dy %>% kable()
```

unique_id	kpss_stat	kpss_p	kpss_lags
consumo	0.0196341	0.1	5
igae	0.0170070	0.1	5
inversion	0.0150415	0.1	5

VAR and cointegration

```
df_wide <- df %>%
  dplyr::select(ds, unique_id, y) %>%
  tidyr::pivot_wider(names_from = unique_id, values_from = y) %>%
  dplyr::arrange(ds) %>%
  tidyr::drop_na(igae, consumo, inversion)

start_year <- lubridate::year(min(df_wide$ds))
start_month <- lubridate::month(min(df_wide$ds))

Y_ts <- ts(
  df_wide %>% dplyr::select(igae, consumo, inversion),
  start = c(start_year, start_month),
  frequency = 12
)

lag_sel <- VARselect(Y_ts, lag.max = 12, type = "const")
lag_sel$selection
```

```
## AIC(n)  HQ(n)  SC(n) FPE(n)
##     12     12     12     12
```

```
as.data.frame(t(lag_sel$selection)) %>%
  kable() %>%
  kable_styling(full_width = FALSE)
```

AIC(n)	HQ(n)	SC(n)	FPE(n)
12	12	12	12

```r
K <- as.integer(lag_sel$selection["AIC(n)"])

joh_trace <- ca.jo(Y_ts, type = "trace", ecdet = "const", K = K)
summary(joh_trace)
```

```
##
## ######################
## # Johansen-Procedure #
## ######################
##
## Test type: trace statistic , without linear trend and constant in cointegration
##
## Eigenvalues (lambda):
## [1] 7.314055e-02 2.262057e-02 5.484421e-03 1.206013e-16
##
## Values of teststatistic and critical values of test:
##
##           test 10pct  5pct  1pct
## r <= 2 |  2.10  7.52  9.24 12.97
## r <= 1 | 10.84 17.85 19.96 24.60
## r = 0  | 39.86 32.00 34.91 41.07
##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##                 igae.l12 consumo.l12 inversion.l12   constant
## igae.l12       1.0000000   1.0000000     1.0000000  1.0000000
## consumo.l12   -0.7941830   0.6528752    -0.9026245 -0.7926319
## inversion.l12 -0.1038854  -1.2660298     0.2127151 -0.1131089
## constant      -4.1466228 -33.7108089   -25.4918940 -8.5388173
##
## Weights W:
## (This is the loading matrix)
##
##               igae.l12 consumo.l12 inversion.l12      constant
## igae.d      0.11411580 0.006519593   -0.02354182 -1.006000e-14
## consumo.d   0.12865019 0.019503057   -0.01019157 -1.958215e-14
## inversion.d 0.09950496 0.047420276   -0.05948143 -3.023837e-14
```

```r
joh_eigen <- ca.jo(Y_ts, type = "eigen", ecdet = "const", K = K)
summary(joh_eigen)
```

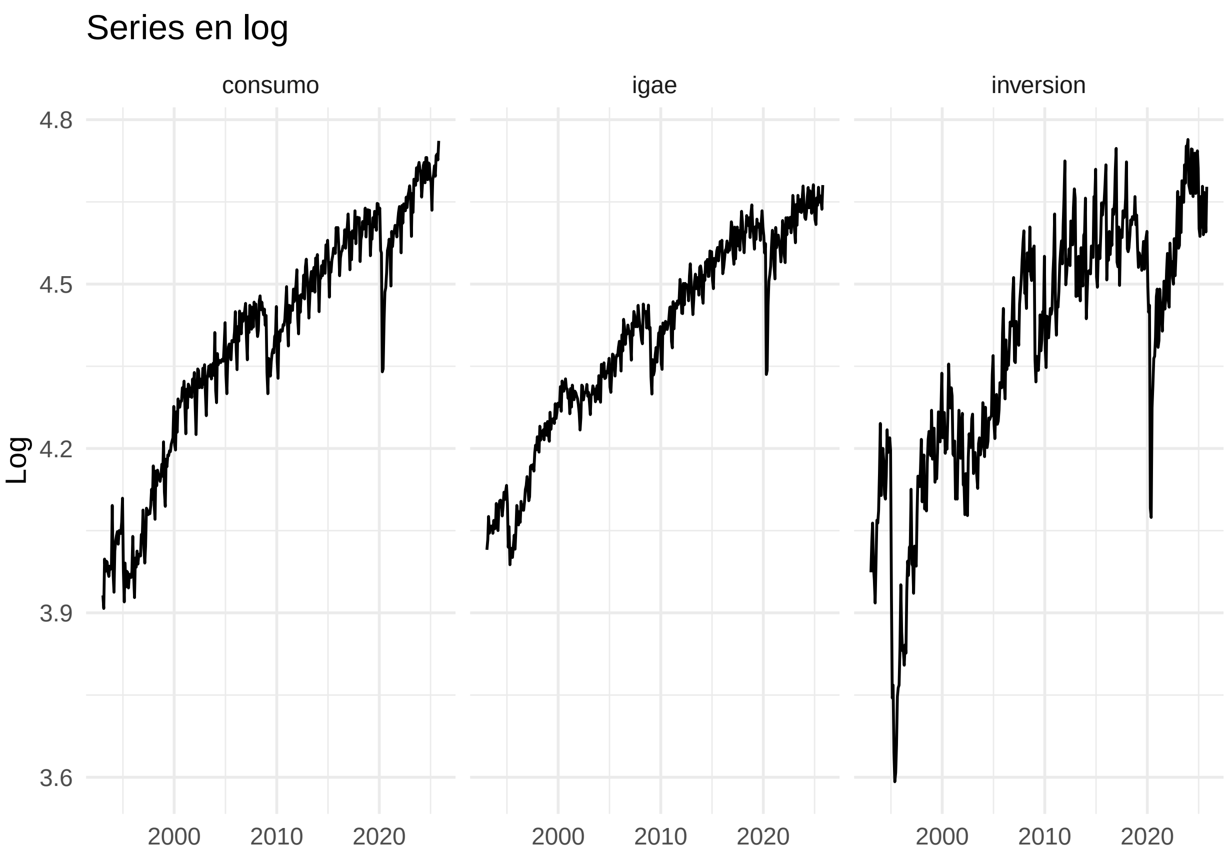
```
##
## ######################
## # Johansen-Procedure #
## ######################
##
## Test type: maximal eigenvalue statistic (lambda max) , without linear trend and constant in cointegra
##
## Eigenvalues (lambda):
```

```
## [1] 7.314055e-02 2.262057e-02 5.484421e-03 1.206013e-16
##
## Values of teststatistic and critical values of test:
##
##           test 10pct  5pct  1pct
## r <= 2 |  2.10  7.52  9.24 12.97
## r <= 1 |  8.74 13.75 15.67 20.20
## r = 0  | 29.01 19.77 22.00 26.81
##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##                 igae.l12 consumo.l12 inversion.l12   constant
## igae.l12       1.0000000   1.0000000     1.0000000  1.0000000
## consumo.l12   -0.7941830   0.6528752    -0.9026245 -0.7926319
## inversion.l12 -0.1038854  -1.2660298     0.2127151 -0.1131089
## constant      -4.1466228 -33.7108089   -25.4918940 -8.5388173
##
## Weights W:
## (This is the loading matrix)
##
##               igae.l12 consumo.l12 inversion.l12      constant
## igae.d      0.11411580 0.006519593   -0.02354182 -1.006000e-14
## consumo.d   0.12865019 0.019503057   -0.01019157 -1.958215e-14
## inversion.d 0.09950496 0.047420276   -0.05948143 -3.023837e-14
```

Log transform and regression

```
df <- df %>%
  mutate(ly = log(y))
```

```
plot_facets(df, "ly", "Log", "Series en log")
```



```r
df_wide_ly <- df %>%
  dplyr::filter(is.finite(ly)) %>%
  dplyr::select(ds, unique_id, ly) %>%
  tidyr::pivot_wider(names_from = unique_id, values_from = ly) %>%
  dplyr::arrange(ds) %>%
  tidyr::drop_na(igae, consumo, inversion)

model <- lm(igae ~ consumo + inversion, data = df_wide_ly)
coeftest(model, vcov = vcovHAC(model))
```

```
##
## t test of coefficients:
##
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.754107   0.073725 10.2286  < 2e-16 ***
## consumo     0.738154   0.043006 17.1640  < 2e-16 ***
## inversion   0.091926   0.037519  2.4501  0.01472 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(model)
```

```
##
## Call:
## lm(formula = igae ~ consumo + inversion, data = df_wide_ly)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.071474 -0.016278 -0.001982  0.018409  0.060193
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.75411    0.02523  29.891  < 2e-16 ***
## consumo      0.73815    0.01446  51.050  < 2e-16 ***
## inversion    0.09193    0.01297   7.085 6.52e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02446 on 391 degrees of freedom
## Multiple R-squared:  0.9819, Adjusted R-squared:  0.9818
## F-statistic: 1.062e+04 on 2 and 391 DF,  p-value: < 2.2e-16
```