

Estimando o repasse externo sobre a inflação no Brasil

<http://rleripio.com.br/estimando-o-repasse-externo-sobre-a-inflacao-no-brasil>

6 de novembro de 2018

Introdução

Medidas de repasse externo fazem parte da caixa de ferramentas de todos aqueles que trabalham com inflação e política monetária. Sobretudo em momentos de forte depreciação do câmbio, como observado ao longo deste ano, ter estimativas do impacto sobre o nível geral de preços torna-se especialmente relevante. Não à toa, o Banco Central tem se referido com frequência ao tema tanto em publicações oficiais como em declarações de sua diretoria.

Existem diversas metodologias para estimar o repasse, desde modelos mais complexos do tipo DSGE a abordagens mais simples que utilizam formas reduzidas das Curvas de Phillips derivadas de modelos estruturais menores. O objetivo do exercício é oferecer uma medida de repasse externo a partir da estimação de um modelo inspirado na forma reduzida da Curva de Phillips do Modelo Semiestrutural de Pequeno Porte do BCB (ver aqui), conforme descrito abaixo:

$$INF_t^{livres} = \sum_{i>0} \beta_{1i} E_t INF_{t+i}^{total} + \sum_{j>0} \beta_{2j} INF_{t-j}^{total} + \sum_{k\geq 0} \beta_{3k} INF_{t-k}^{importada} + \sum_{l>0} \beta_{4l} hiatot_{t-l} + \sum_{m=1}^4 D_m + \epsilon_t$$

com a imposição da restrição de verticalidade de longo prazo:

$$\sum_{i>0} \beta_{1i} + \sum_{j>0} \beta_{2j} + \sum_{k\geq 0} \beta_{3k} = 1$$

Basicamente, a inflação dos preços livres é uma função linear da **inflação passada**, das **expectativas de inflação**, do **hiato do produto** e da **inflação importada**. Vale ressaltar que a medida de inflação importada é capturada no modelo do BCB através de um índice de commodities em dólar convertido em reais (o índice CRB ou o **IC-Br** calculado pelo próprio BCB) e não diretamente pela variação cambial. Foram utilizados dados trimestrais e incluídas **dummies sazonais** (D_m).

A especificação final contou com as primeiras duas defasagens do IPCA total, a terceira defasagem do hiato do produto (calculado pelo IPEA), a expectativa de inflação formada no trimestre corrente para o trimestre imediatamente à frente e a primeira defasagem do índice IC-Br – o qual mede o repasse externo.

Código para replicar (Acompanha arquivo .RData)

Passo 1: Carregar pacotes necessários. O pacote ‘rbcb’ foi retirado recentemente do CRAN devido a problemas técnicos e, por esta razão, será instalada a versão disponibilizada na conta GitHub do autor.

```
library(devtools)
library(tidyverse)
library(lubridate)
library(readxl)
library(restriktor)
library(RcppRoll)
library(broom)
library(scales)
```

```
devtools::install_github('wilsonfreitas/rbcb')
```

```
library(rbcb)
```

Passo 2: Importar dados. Note que o modelo será estimado em frequência trimestral. Entretanto, o sistema Focus do BCB reporta em frequência diária as expectativas para os meses seguintes. É preciso, portanto, obter a série da expectativa no trimestre corrente para os trimestres seguintes. Mais especificamente, vamos calcular as expectativas diárias para os trimestres seguintes e, em seguida, obter a média trimestral destas expectativas.

```
exp_ipca <- rbcb::get_monthly_market_expectations("IPCA", end_date = "2018-09-31")
```

```
diff_quarter <- function(end_date, start_date){
```

```
  year_to_quarters <- (floor(end_date)*4 + (end_date %% 1)*10) -  
                      (floor(start_date)*4 + (start_date %% 1)*10)
```

```
  return(year_to_quarters)
```

```
}
```

```
exp_ipca_aux <- exp_ipca %>%
```

```
  dplyr::select(date, reference_month, median) %>%
```

```
  dplyr::mutate(reference_month = lubridate::ymd(paste(reference_month,  
                                                       "01",  
                                                       sep = "-")) %>%
```

```
  dplyr::mutate(date_year = lubridate::year(date),  
                date_month = lubridate::month(date)) %>%
```

```
  dplyr::group_by(date_year, date_month, reference_month) %>%
```

```
  dplyr::summarise(median_month = mean(median)) %>%
```

```
  dplyr::ungroup() %>%
```

```
  dplyr::mutate(date = lubridate::make_date(year = date_year,  
                                             month = date_month)) %>%
```

```
  dplyr::select(date, reference_month, median_month) %>%
```

```
  dplyr::filter(date > "2001-12-01") %>%
```

```
  dplyr::mutate(ref_quarter = lubridate::quarter(reference_month,  
                                                  with_year = T)) %>%
```

```
  dplyr::group_by(date, ref_quarter) %>%
```

```
  dplyr::summarise(median_quarter = last(((cumprod(1+(median_month/100)))-1)*100)) %>%
```

```
  dplyr::mutate(date_quarter = lubridate::quarter(date, with_year = T)) %>%
```

```

dplyr::group_by(date_quarter, ref_quarter, add = F) %>%

dplyr::summarise(median_quarter = mean(median_quarter)) %>%

dplyr::filter(ref_quarter > date_quarter) %>%

dplyr::mutate(diff = round(diff_quarter(ref_quarter, date_quarter)),1) %>%

dplyr::select(-ref_quarter) %>%

tidyr::spread(key = diff, value = median_quarter)

colnames(exp_ipca_aux)[-1] <- paste("EInf_t+", colnames(exp_ipca_aux)[-1], sep = "")

```

Para as séries do IPCA total e preços livres, é necessário calcular o acumulado trimestral.

```

series <- list("ipca_livres" = 11428,
              "ipca_total" = 433)

acum_quarter <- function(x){

  x_fac <- 1+(x/100)

  x_cum <- RcppRoll::roll_prodr(x_fac, n = 3)

  x_qr <- last((x_cum-1)*100)

  return(x_qr)

}

dados_ipca <- rcbcb::get_series(series) %>%

purrr::reduce(inner_join) %>%

dplyr::mutate(date_quarter = lubridate::quarter(date, with_year = TRUE)) %>%

dplyr::group_by(date_quarter) %>%

dplyr::summarise_at(vars(ipca_total, ipca_livres), funs(acum_quarter))

```

Para o IC-Br, basta calcular a diferença percentual entre as observações no início e no final de cada trimestre.

```

acum_ic <- function(x){

  x_diff <- log(x/first(x))*100

  x_acum <- last(x_diff)

  return(x_acum)

}

```

```
dados_ic <- rbcbr::get_series(list("ic_br" = 27574)) %>%

  dplyr::mutate(date_quarter = lubridate::quarter(date, with_year = TRUE)) %>%

  dplyr::group_by(date_quarter) %>%

  dplyr::summarise_at(vars(ic_br), funs(acum_ic))
```

O hiato do produto é divulgado pelo IPEA em frequência trimestral e, logo, não necessita de transformações.

```
hiato <- read_excel("hiato.xlsx") %>%

  dplyr::mutate(date_quarter = as.numeric(gsub(" T", ".", date_qr)),
    hiato = hiato*100) %>%

  dplyr::select(date_quarter, hiato)
```

Passo 3: Reunir os dados e criar as variáveis dummies e defasagens utilizadas.

```
dados_reg <- dplyr::inner_join(dados_ipca, dados_ic) %>%

  dplyr::inner_join(hiato) %>%

  dplyr::inner_join(exp_ipca_aux) %>%

  dplyr::mutate(quarter = sub('.*\\.\\.', '', date_quarter)) %>%

  dplyr::filter(date_quarter >= 2002.1) %>%

  dplyr::mutate(ipca_l1 = lag(ipca_total, 1),
    ipca_l2 = lag(ipca_total, 2),
    hiato_l3 = lag(hiato, 3),
    ic_l1 = lag(ic_br, 1),
    Einf_1 = `EInf_t+1`)
```

Passo 4: estimar o modelo sem a restrição (unr) e, em seguida, com a restrição (res)

```
modelo_unr <- lm(ipca_livres ~ -1 + ipca_l1 + ipca_l2 + Einf_1 + hiato_l3 + ic_l1 + quarter, data = dados_reg)

modelo_res <- restriktor(modelo_unr, constraints = ' ipca_l1 + ipca_l2 + Einf_1 + ic_l1 == 1 ')

summary(modelo_res)
```

```
##
## Call:
## conLM.lm(object = modelo_unr, constraints = " ipca_l1 + ipca_l2 + Einf_1 + ic_l1 == 1 ")
##
## Restriktor: restricted linear model:
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.868032 -0.510813 -0.070717  0.267915  2.741512
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## ipca_l1      0.427546   0.125067  3.4185  0.001231 **
```

```
## ipca_l2 -0.268246 0.129212 -2.0760 0.042854 *
## Einf_1 0.784595 0.144402 5.4334 1.487e-06 ***
## hiato_l3 0.069509 0.039476 1.7608 0.084149 .
## ic_l1 0.056105 0.016335 3.4346 0.001174 **
## quarter1 0.532901 0.200849 2.6532 0.010549 *
## quarter2 0.042560 0.214448 0.1985 0.843456
## quarter3 -0.184367 0.199325 -0.9250 0.359261
## quarter4 0.410004 0.182891 2.2418 0.029267 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.66685 on 52 degrees of freedom
## Standard errors: standard
## Multiple R-squared reduced from 0.888 to 0.888
##
## Generalized Order-Restricted Information Criterion:
## Loglik Penalty Goric
## -57.551 9.000 133.102
```

Podemos também criar uma representação gráfica para os coeficientes do modelo da seguinte maneira:

```
unr_tidy <- broom::tidy(modelo_unr)

res_tidy <- tibble(term = unr_tidy$term,
  estimate = modelo_res$b.restr,
  std.error = summary(modelo_res)$coefficients[, 2],
  low = estimate - 1.64*std.error,
  high = estimate + 1.64*std.error)

res_tidy %>%

  dplyr::filter(!term %in% c("quarter1", "quarter2", "quarter3", "quarter4")) %>%

  ggplot(aes(estimate, term, xmin = low, xmax = high, height = 0)) +

  geom_point() +

  geom_vline(xintercept = 0) +

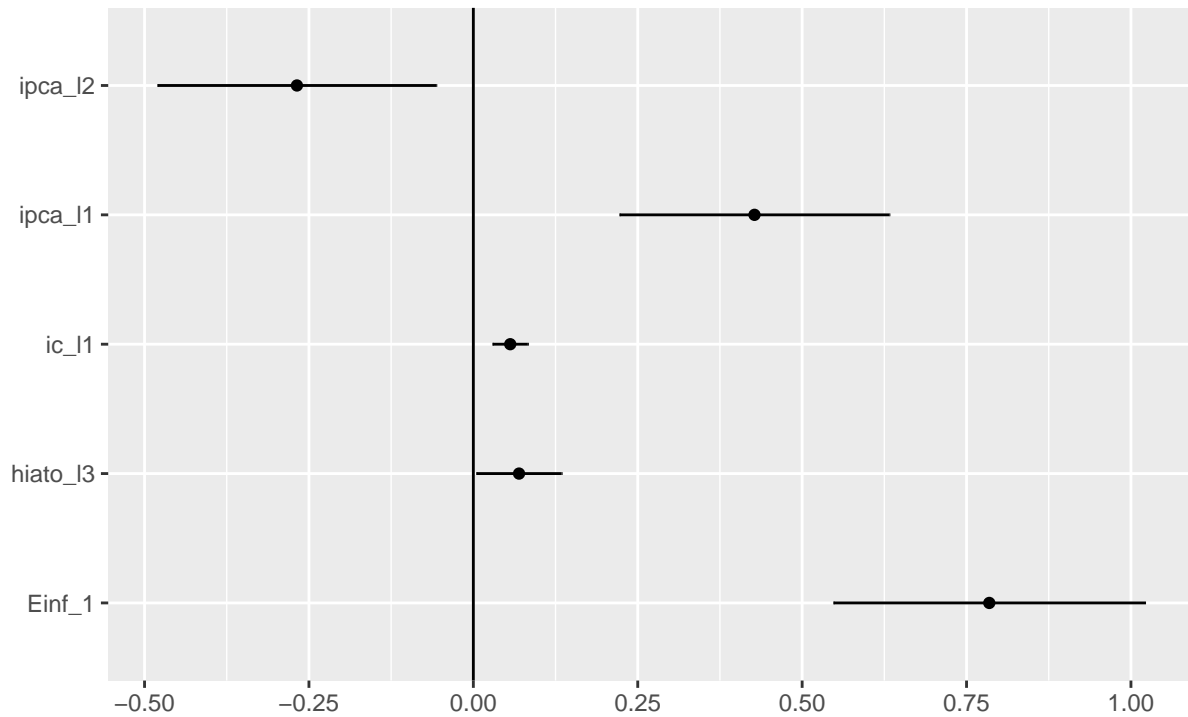
  geom_errorbarh() +

  labs(x = "",
    y = "",
    title = "Coeficientes estimados da Curva de Phillips",
    subtitle = "Modelo com restrição de verticalidade") +

  scale_x_continuous(breaks = seq(-0.5, 1, 0.25))
```

Coefficientes estimados da Curva de Phillips

Modelo com restrição de verticalidade



O coeficiente de interesse – IC-Br (ic_l1) – foi estimado entre 0.03 e 0.08, com valor centrado em 0.056. Isto é, com base no modelo, um aumento de 10% na inflação importada (IC-Br) provoca, em média, elevação de 0.56 p.p nos preços livres dentro do horizonte de um trimestre, tudo o mais constante.