Automating update of a finance database for the Euro Area

Published on June 1, 2016 by Thomas Brand (/author/thomas-brand.html)

database (/tag/database.html) , model (/tag/model.html) , estimation (/tag/estimation.html) , R (/tag/r.html)

The aim of this post is to automatically update a database as the one used in Christiano et al. (2014), but for the Euro area, on the basis of the Smets and Wouters (2003) database.

Four financial series are originally used in Christiano et al. (2014) :

- Loans to non-financial corporations
- Bank lending rates
- Entrepreneurial net worth
- Long-term interest rates

To those series we add 2 series :

- House prices
- Loans to households

The sources we use here are :

- the Area-Wide Model (AWM), originally constructed by Fagan et al. (2001)
- International Financial Statistics (IMF)
- Bank for International Settlements
- European Central Bank

We take data directly from DBnomics (https://db.nomics.world/). The DBnomics API can be accessed through R with the rdbnomics (https://cran.r-project.org/web/packages/rdbnomics/index.html) package. All the following code is written in R, thanks to the RCoreTeam (2016) and the RStudioTeam (2016).

Loans to non-financial corporations and to households

We download loans series from the Bank for International Settlements.

```
# List of Euro area countries available in BIS database
EAtot_code <- c("DE", "FI", "FR", "IT", "PT", "AT",
                "GR", "IE", "NL", "BE", "ES", "XM")
url_country <- paste0(EAtot_code, collapse = "+")
filter <- paste0("Q.",url_country,".N+H.A.M.XDC.A")

# N or H: Borrowing sector : NFC or Households
# A: Lending sector : All
# M: Valuation method : Market value
# XDC: Unit type: Domestic currency
# A: Adjustment : Adjustment for breaks

df <- rdb("BIS","total_credit",mask = filter)

loans <-
  df   %>%
  select(period, series_code, value, BORROWERS_CTY, series_name) %>%
  rename(var = series_code,
         country = BORROWERS_CTY) %>%
  na.omit() %>%
  filter(year(period)>=1980) %>%
  arrange(var, period)

loans_nfc <-
  loans %>%
  filter(substr(var,6,6)=="N") %>%
  mutate(var="loans_nfc")

varname_nfc <- unique(as.character(filter(loans_nfc,country == "XM")$series_name))
loans_nfc %<>% select(-series_name)

loans_hh <- loans %>%
  filter(substr(var,6,6)=="H") %>%
  mutate(var=as.factor("loans_hh"))

varname_hh <- unique(as.character(filter(loans_hh,country == "XM")$series_name))
loans_hh %<>% select(-series_name)
```

We can check the first date available for loans to non-financial corporations and to households (XM stands for Euro area).

```
loans_nfc %>%
  group_by(country) %>%
  summarize(firstdate = min(period)) %>%
  arrange(firstdate) %>%
  ungroup()
```

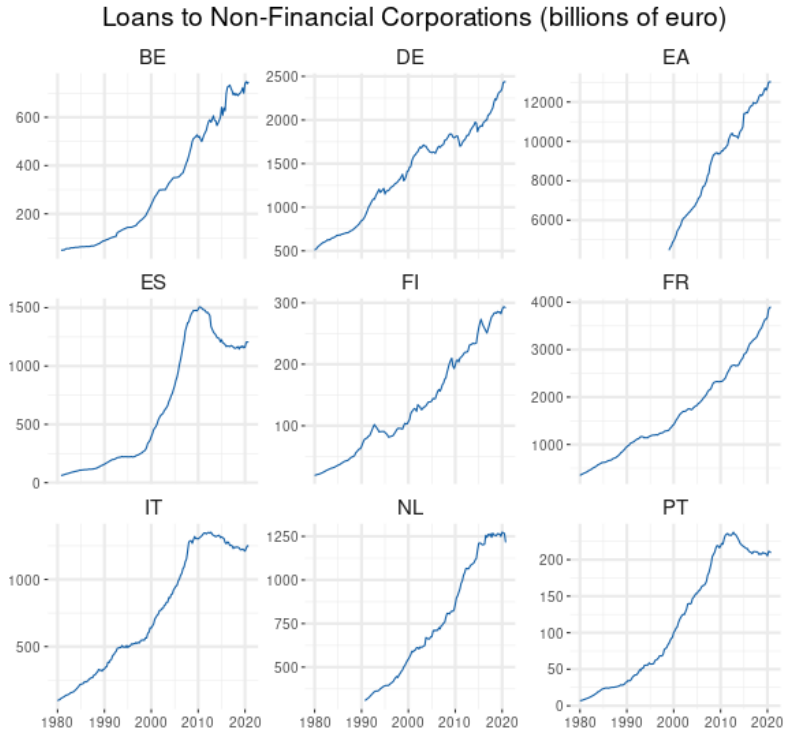
```
## # A tibble: 12 x 2
##    country firstdate
##    <chr>   <date>
##  1 DE      1980-01-01
##  2 FI      1980-01-01
##  3 FR      1980-01-01
##  4 IT      1980-01-01
##  5 PT      1980-01-01
##  6 BE      1980-10-01
##  7 ES      1980-10-01
##  8 NL      1990-10-01
##  9 GR      1994-10-01
## 10 AT      1995-10-01
## 11 XM      1997-10-01
## 12 IE      2002-01-01
```

```
loans_hh %>%
  group_by(country) %>%
  summarize(firstdate = min(period)) %>%
  arrange(firstdate) %>%
  ungroup()
```

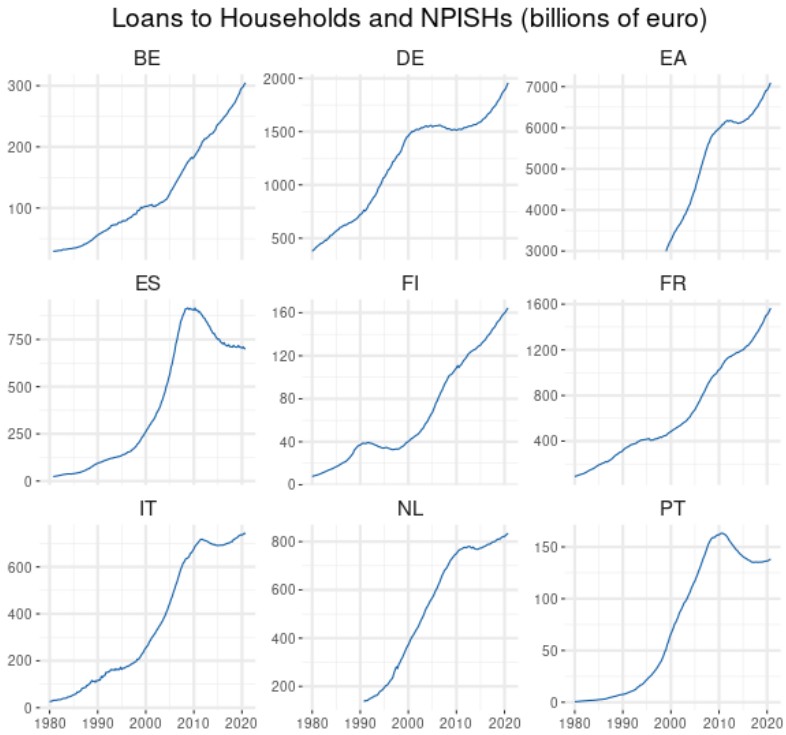
```
## # A tibble: 12 x 2
##    country firstdate
##    <chr>   <date>
##  1 DE      1980-01-01
##  2 FI      1980-01-01
##  3 FR      1980-01-01
##  4 IT      1980-01-01
##  5 PT      1980-01-01
##  6 BE      1980-10-01
##  7 ES      1980-10-01
##  8 NL      1990-10-01
##  9 GR      1994-10-01
## 10 AT      1995-10-01
## 11 XM      1999-01-01
## 12 IE      2002-01-01
```

We decide to retain only countries which are available before 1990 to compute the aggregated series.

```
available_country <- filter(loans_nfc,period=="1990-10-01")$country

loans_nfc_countries <-
  loans_nfc %>%
  filter(country %in% available_country)
loans_nfc_EA <-
  loans_nfc %>%
  filter(country=="XM",
         period>="1999-01-01") %>%
  mutate(country = "EA")

ggplot(bind_rows(loans_nfc_countries,loans_nfc_EA),aes(period,value))+
  geom_line(colour=blueObsMacro)+
  facet_wrap(~country,ncol=3,scales = "free_y")+
  theme + xlab(NULL) + ylab(NULL)+
  ggtitle("Loans to Non-Financial Corporations (billions of euro)")
```

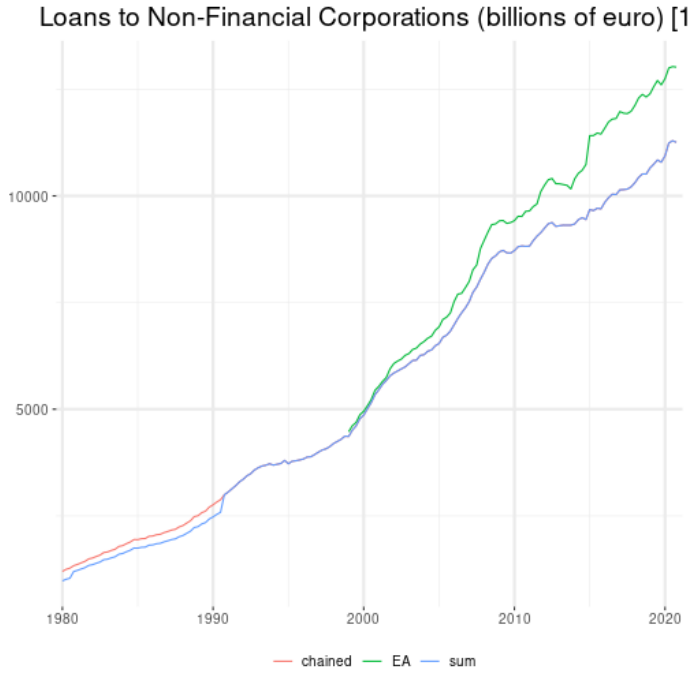



```
loans_hh_countries <-
  loans_hh %>%
  filter(country %in% available_country)
loans_hh_EA <- loans_hh %>%
  filter(country=="XM") %>%
  mutate(country = "EA")

ggplot(bind_rows(loans_hh_countries,loans_hh_EA),aes(period,value))+
  geom_line(colour=blueObsMacro)+
  facet_wrap(~country,ncol=3,scales = "free_y")+
  theme + xlab(NULL) + ylab(NULL)+
  ggtitle("Loans to Households and NPISHs (billions of euro)")
```



Now we compare raw sum of countries, chained sum of countries and EA series.

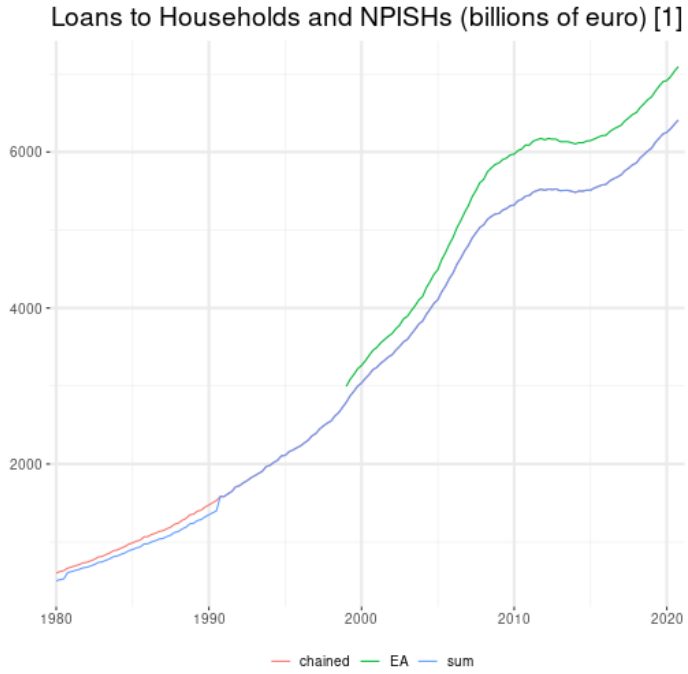
```
loans_nfc_countries %<>%
  select(-var) %>%
  mutate(var=country)
loans_nfc_sumAll <-
  loans_nfc_countries %>%
  group_by(period) %>%
  summarize(value=sum(value)) %>%
  mutate(var="sum")
loans_nfc_sumNoNL <-
  loans_nfc_countries %>%
  filter(! var == "NL") %>%
  group_by(period) %>%
  summarize(value=sum(value)) %>%
  mutate(var="sum")
loans_nfc_sumNoNLESBE <-
  loans_nfc_countries %>%
  filter(! var %in% c("NL","ES","BE")) %>%
  group_by(period) %>%
  summarize(value=sum(value)) %>%
  mutate(var="sum")

loans_nfc_chainedNL <-
  chain(to_rebase = loans_nfc_sumNoNL,
        basis = loans_nfc_sumAll,
        date_chain = "1990-10-01")
loans_nfc_chained <-
  chain(to_rebase = loans_nfc_sumNoNLESBE,
        basis = loans_nfc_chainedNL,
        date_chain = "1980-10-01") %>%
  mutate(var="chained")

loans_nfc_EA %<>% select(-country) %>% mutate(var="EA")

ggplot(bind_rows(loans_nfc_sumAll, loans_nfc_EA, loans_nfc_chained), aes(period, v
alue,colour=var))+
  geom_line()+
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL)+
  theme(legend.title=element_blank()) +
  ggtitle("Loans to Non-Financial Corporations (billions of euro) [1]")
```




```
varname_nfc
```

```
## [1] "Quarterly - Euro area - Non-financial corporations - All sectors - Market
value - Domestic currency (incl. conversion to current currency made using a fix p
arity) - Adjusted for breaks"
```

```
loans_hh_countries %<>%
  select(-var) %>%
  mutate(var=country)
loans_hh_sumAll <-
  loans_hh_countries %>%
  group_by(period) %>%
  summarize(value=sum(value)) %>%
  mutate(var="sum")
loans_hh_sumNoNL <-
  loans_hh_countries %>%
  filter(! var == "NL") %>%
  group_by(period) %>%
  summarize(value=sum(value)) %>%
  mutate(var="sum")
loans_hh_sumNoNLESBE <-
  loans_hh_countries %>%
  filter(! var %in% c("NL","ES","BE")) %>%
  group_by(period) %>%
  summarize(value=sum(value)) %>%
  mutate(var="sum")

loans_hh_chainedNL <-
  chain(to_rebase = loans_hh_sumNoNL,
        basis = loans_hh_sumAll,
        date_chain = "1990-10-01")
loans_hh_chained <-
  chain(to_rebase = loans_hh_sumNoNLESBE,
        basis = loans_hh_chainedNL,
        date_chain = "1980-10-01") %>%
  mutate(var="chained")

loans_hh_EA %<>% select(-country) %>% mutate(var="EA")

ggplot(bind_rows(loans_hh_sumAll, loans_hh_EA, loans_hh_chained), aes(period, valu
e,colour=var))+
  geom_line()+
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL)+
  theme(legend.title=element_blank()) +
  ggtitle("Loans to Households and NPISHs (billions of euro) [1]")
```




```
varname_hh
```

```
## [1] "Quarterly - Euro area - Households & NPISHs - All sectors - Market value -
Domestic currency (incl. conversion to current currency made using a fix parity) -
Adjusted for breaks"
```

Eventually, we use EA series in levels after 1999, and the growth rates of the sum of loans for available countries to complete the series for historical data.

```
loans_nfc <-
  chain(to_rebase = mutate(loans_nfc_chained,var="loans_nfc"),
        basis = mutate(loans_nfc_EA,var="loans_nfc"),
        date = "1999-01-01")

loans_hh <-
  chain(to_rebase = mutate(loans_hh_chained,var="loans_hh"),
        basis = mutate(loans_hh_EA,var="loans_hh"),
        date = "1999-01-01")
```

Bank lending rates

Historical data from OECD

To build long series of lending rates, we use historical data from the OECD Main Economic Indicators. Historical series will be used before 2000, because such series are available from ECB since 2000Q1. Thus, we consider only five countries. As in the AWM methodology, we weigth the sum of the lending rates by the gross domestic product based on purchasing-power-parity (PPP) of each country in 1995, from the IMF World Economic Outlook.

```
country_code <- c("BEL","FRA","DEU","ITA","ESP")
url_country <- paste0(country_code, collapse = "+")

# Download the 5 countries' lending rates from OECD
filter <- paste0(url_country,".IR3TIB01.ST.Q")
df <- rdb("OECD","MEI",mask = filter)
lendingrate_bycountry <-
  df %>%
  select(country=LOCATION,period, value) %>%
  filter(year(period)>=1980)

# Download the 5 countries' PPP GDP from WEO
filter <- paste0(url_country,".PPPGDP")
df <- rdb("IMF","WEO:latest",mask = filter)
pppgdp <-
  df %>%
  filter(period == "1995-01-01") %>%
  select(country = `weo-country`,
         value_pppgdp = value)
sum_pppgdp <- sum(pppgdp$value_pppgdp)

# Merge databases and build a weighted mean
lendingrate_old <-
  left_join(lendingrate_bycountry, pppgdp, by = "country") %>%
  transmute(period = period,
            country = country,
            value = value * value_pppgdp) %>%
  group_by(period) %>%
  summarise(value = sum(value) / sum_pppgdp) %>%
  mutate(var="lendingrate_old")
```

Recent data from ECB

```
df <- rdb(ids="ECB/MIR/M.U2.B.A2A.A.R.A.2240.EUR.N")

varname <- unique(as.character(df$series_name))

lendingrate_recent <-
  df %>%
  select(period, value) %>%
  mutate(period=paste(year(period),quarter(period))) %>%
  group_by(period) %>%
  summarize(value=mean(value)) %>%
  mutate(var= "lendingrate_recent",
         period=yq(period))
```

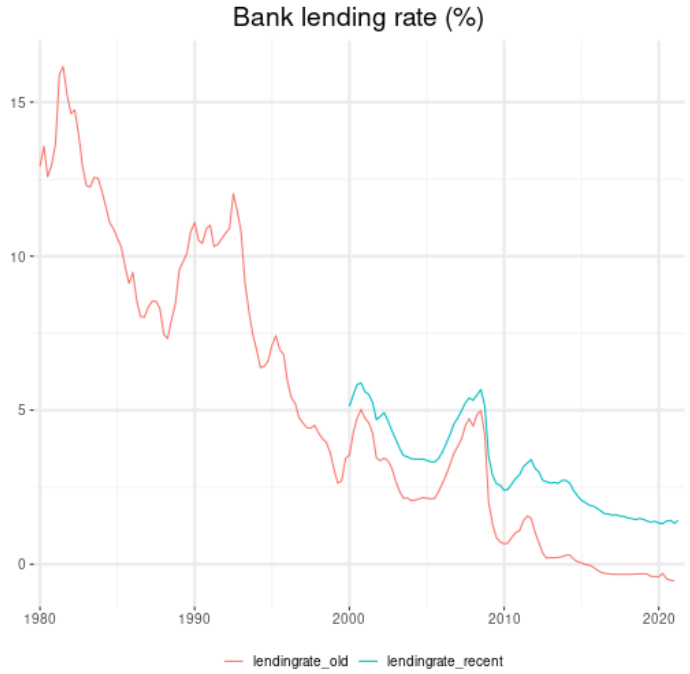
More precisely, the recent bank lending rates come from the ECB and are described as:

```
varname
```

```
## [1] "Monthly - Euro area (changing composition) - Credit and other institutions
(MFI except MMFs and central banks) - Loans other than revolving loans and overdra
fts, convenience and extended credit card debt - Total - Annualised agreed rate (A
AR) / Narrowly defined effective rate (NDER) - Total - Non-Financial corporations
(S.11) - Euro - New business"
```

Compare historical and recent data

```
dataplot <- bind_rows(lendingrate_recent,
                      lendingrate_old)

ggplot(dataplot,aes(period,value, colour=var)) +
  geom_line() +
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL)+
  theme(legend.title=element_blank()) +
  ggtitle("Bank lending rate (%)")
```



Chain historical and recent data

```
lendingrecent_value2000Q1 <-
  lendingrate_recent %>%
  filter(period=="2000-01-01") %>%
  pull(value)
lendingold_value2000Q1 <-
  lendingrate_old %>%
  filter(period=="2000-01-01") %>%
  pull(value)
diff <- lendingrecent_value2000Q1 - lendingold_value2000Q1

lendingrate <-
  lendingrate_old %>%
  filter(period<="1999-10-01") %>%
  mutate(value=value+diff) %>%
  bind_rows(lendingrate_recent) %>%
  mutate(var="lendingrate")
```

Long-term interest rate

The historical long-term interest rate come from the AWM database. The recent one is taken from the ECB.

```
link_to_awm <- "http://www.eabcn.org/sites/default/files/awm19up15.csv"

if (! "awm19up15.csv" %in% list.files()) {
  download.file(link_to_awm,
                destfile = "awm19up15.csv",
                method = "auto")
}

awm <- read.csv("awm19up15.csv", sep=",")
longrate_old <-
  awm %>%
  transmute(longrate = LTN, # Long-Term Interest Rate (Nominal)
            period = as.Date(as.yearqtr(X))) %>%
  gather(var, value, -period, convert = TRUE) %>%
  filter(year(period)>=1980)
```

```
df <- rdb(ids="ECB/IRS/M.I8.L.L40.CI.0000.EUR.N.Z")

varname <- unique(as.character(df$series_name))

longrate_recent <- df %>%
  select(period, value) %>%
  mutate(period=paste(year(period),quarter(period))) %>%
  group_by(period) %>%
  summarize(value=mean(value)) %>%
  mutate(var= "longrate",
         period=yq(period))

dataplot <- bind_rows(data.frame(longrate_recent,ind="recent"),
                      data.frame(longrate_old,ind="old"))

ggplot(dataplot,aes(period,value, colour=ind)) +
  geom_line() +
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL)+
  theme(legend.title=element_blank()) +
  ggtitle("Long-term interest rate (%)")
```


```
longrate <- chain(basis = longrate_recent,
                  to_rebase = longrate_old,
                  date_chain = "2001-01-01")
```

More precisely, the recent long-term interest rates come from the ECB and are described as

```
varname
```

```
## [1] "Monthly - Euro area 19 (fixed composition) as of 1 January 2015 - Long-ter
m interest rate for convergence purposes - Debt security issued - 10 years - Unspe
cified counterpart sector - Euro - New business - Unspecified"
```

Entrepreunarial net worth

The entrepreunarial net worth is approximated through the Dow Jones index for the Euro area, in a similar way of what is chosen by Christiano et al. (2014) in the US database.

```
df <- rdb(ids="ECB/FM/Q.U2.EUR.DS.EI.DJEURST.HSTA")

varname <- unique(as.character(df$series_name))

networth <- df %>%
  select(value, period) %>%
   mutate(var = as.factor("networth"))

ggplot(networth,aes(period,value)) +
  geom_line(colour=blueObsMacro) +
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL)+
  theme(legend.title=element_blank()) +
  ggtitle("Entrepreunarial net worth (index)")
```

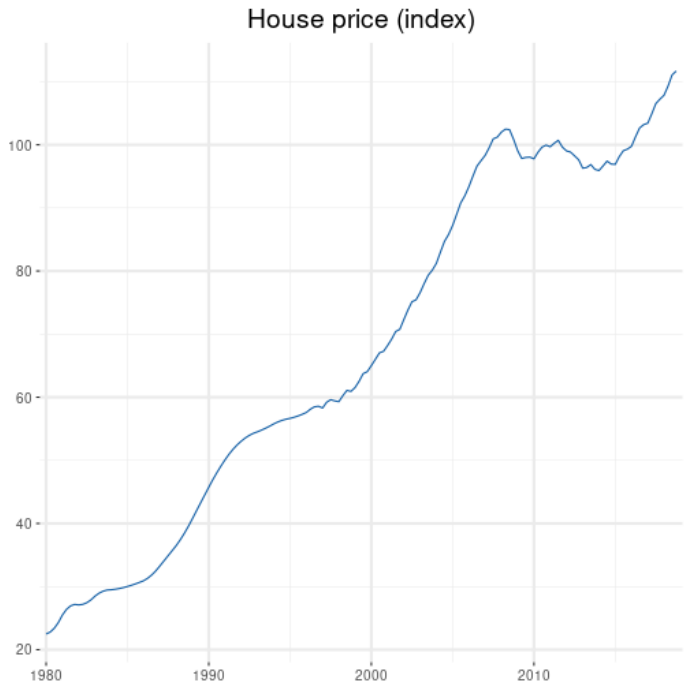
More precisely, the index come from the ECB and is described as

```
varname
```

```
## [1] "Quarterly - Euro area (changing composition) - Euro - DataStream - Equity/
index - Dow Jones Euro Stoxx Price Index - Historical close, average of observatio
ns through period"
```

House prices

```
df <- rdb(ids="ECB/RPP/Q.I8.N.TD.00.3.00")

varname <- unique(as.character(df$series_name))

houseprice<- df %>%
  select(value, period) %>%
    mutate(var = as.factor("houseprice"))

ggplot(houseprice,aes(period,value)) +
  geom_line(colour=blueObsMacro) +
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL)+
  theme(legend.title=element_blank()) +
  ggtitle("House price (index)")
```



More precisely, house prices come from the BIS and are described as

```
varname
```

```
## [1] "Quarterly - Euro area 19 (fixed composition) as of 1 January 2015 - Neithe
r seasonally nor working day adjusted - Residential property prices, New and exist
ing dwellings - Whole country - ECB - Residential property in good and poor condit
ion"
```

Final financial database for the Euro area

We build the final financial database with the 6 series described before.

```
final_df <- bind_rows(loans_nfc,
                      loans_hh,
                      lendingrate,
                      longrate,
                      networth,
                      houseprice)
```

We can check the last date available for each variable.

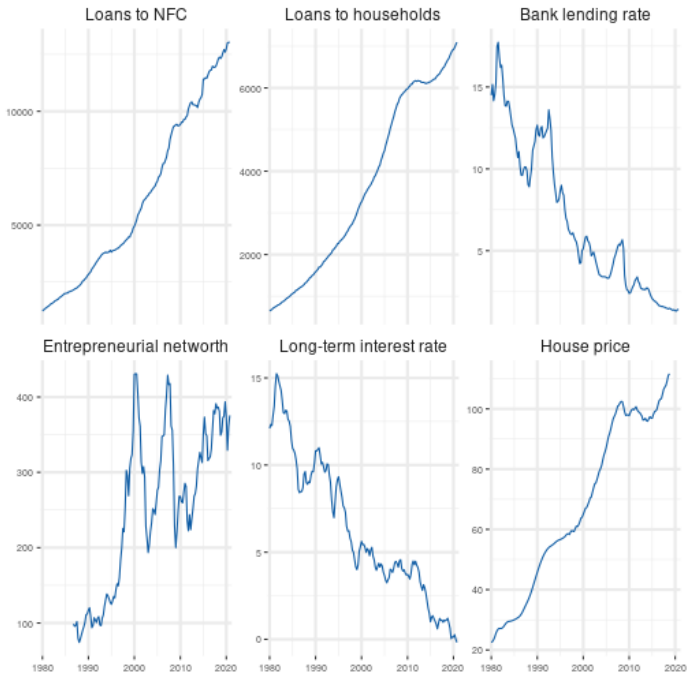
```
maxDate <-
  final_df %>%
  filter(var!="houseprice") %>%
  group_by(var) %>%
  summarize(maxdate=max(period)) %>%
  arrange(maxdate)
kable(maxDate)
```

var	maxdate
loans_hh	2020-10-01
loans_nfc	2020-10-01
lendingrate	2021-04-01
longrate	2021-04-01
networth	2021-04-01

```
minmaxDateFinance <- min(maxDate$maxdate)
final_df %<>% filter(period<=minmaxDateFinance)
```

So we filter the database until 2020 Q4.

```
plot_df <- final_df
listVar <- list("Loans to NFC" = "loans_nfc",
                "Loans to households" = "loans_hh",
                "Bank lending rate" = "lendingrate",
                "Entrepreneurial networth" = "networth",
                "Long-term interest rate" = "longrate",
                "House price" = "houseprice")

plot_df$var <- factor(plot_df$var)
levels(plot_df$var)<-listVar

ggplot(plot_df,aes(period,value))+
  geom_line(colour=blueObsMacro)+
  facet_wrap(~var,scales = "free_y",ncol = 3)+
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL) +
  theme(strip.text=element_text(size=12),
        axis.text=element_text(size=7))
```



You can download the 6 financial series directly here (http://shiny.cepremap.fr/data/EA_Finance_rawdata.csv)

```
EA_Finance_rawdata <-
  final_df %>%
  spread(key = var, value = value)

EA_Finance_rawdata %>%
  write.csv("EA_Finance_rawdata.csv", row.names=FALSE)
```

Final CMR database for the Euro area

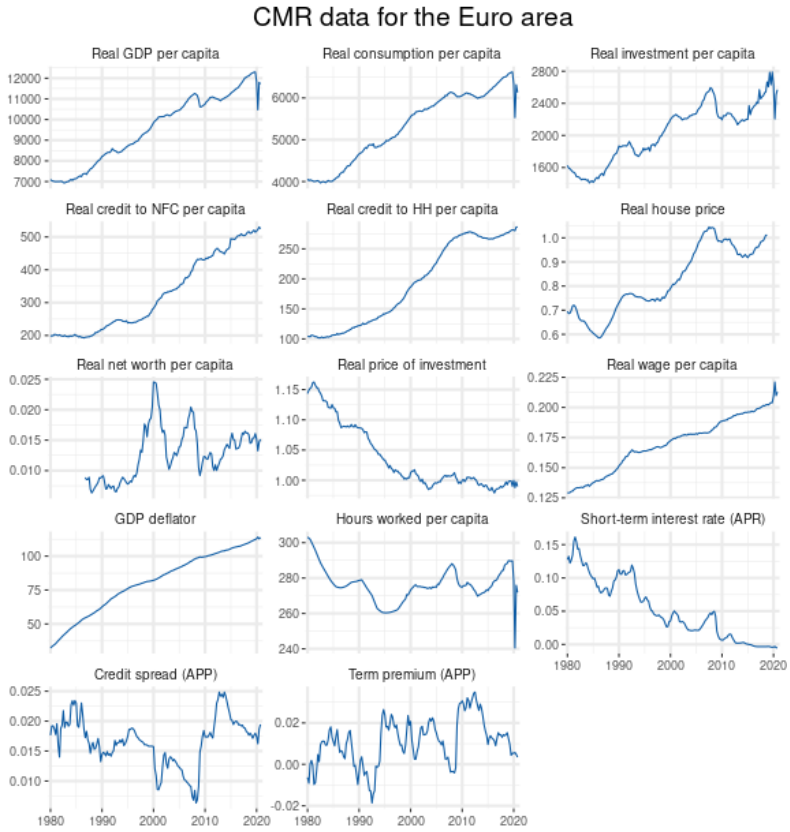
We eventually want to build a database similar to the Christiano et al. (2014) database, but for the Euro area, on the basis of the Smets and Wouters (2003) database. The database will begin in 1980Q1, as the financial series are not available before. You can download all the raw series here (http://shiny.cepremap.fr/data/EA_CMR_rawdata.csv).

```
# Import EA_SW_rawdata.csv
EA_SW_rawdata <-
  read.csv("https://shiny.cepremap.fr/data/EA_SW_rawdata.csv") %>%
  mutate(period=ymd(period))

minmaxDateRaw <- max(EA_SW_rawdata$period)

EA_CMR_rawdata <-
  EA_SW_rawdata %>%
  gather(var, value, -period) %>%
  bind_rows(final_df) %>%
  filter(period <= min(minmaxDateRaw,minmaxDateFinance),
         period >= "1980-01-01") %>%
  spread(var,value)

EA_CMR_rawdata %>%
  write.csv("EA_CMR_rawdata.csv", row.names=FALSE)
```

Then data are normalized by capita and price if needed. Eventually we have 14 series : the 12 series similar to Christiano et al. (2014) plus loans to households and house price series.

```
EA_CMR_data <-
  EA_CMR_rawdata %>%
  transmute(period=gsub(" ", "", as.yearqtr(period)),
            gdp_rpc=1e+6*gdp/(pop*1000),
            conso_rpc=1e+6*conso/(pop*1000),
            inves_rpc=1e+6*inves/(pop*1000),
            defgdp = defgdp,
            wage_rph=1e+6*wage/defgdp/(hours*1000),
            hours_pc=1000*hours/(pop*1000),
            pinves_defl=definves/defgdp,
            loans_nfc_rpc=1e+9*loans_nfc/(pop*1000)/defgdp,
            loans_hh_rpc=1e+9*loans_hh/(pop*1000)/defgdp,
            houseprice_defl=houseprice/defgdp,
            networth_rpc=1e+6*networth/(pop*1000)/defgdp,
            re=shortrate/100,
            slope=(longrate - shortrate)/100,
            creditspread = (lendingrate - shortrate)/100)

EA_CMR_data %>%
  #na.omit() %>%
  write.csv("EA_CMR_data.csv", row.names=FALSE)
```

```
plot_EA_CMR_data <-
  EA_CMR_data %>%
  gather(var, value, -period) %>%
  mutate(period=as.Date(as.yearqtr(period)),
         var=as.factor(var))
levels(plot_EA_CMR_data$var)<-listVar

ggplot(plot_EA_CMR_data,aes(period,value))+
  geom_line(colour=blueObsMacro)+
  facet_wrap(~var,ncol=3,scales = "free_y")+
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL)+
  theme(strip.text=element_text(size=10),
        axis.text=element_text(size=9))+
  ggtitle("CMR data for the Euro area")
```



You can also download ready-to-use (normalized) data for the estimation here (http://shiny.cepremap.fr/data/EA_CMR_data.csv).

Appendix

Chaining function

To chain two datasets, we build a chain function whose input must be two dataframes with three standard columns (`period` , `var` , `value`). It returns a dataframe composed of chained values, ie the dataframe “to rebase” will be chained on the “basis” dataframe.

More specifically, the function :

- computes the growth rates from `value` in the dataframe of the 1st argument
- multiplies it with the value of reference chosen in `value` in the dataframe of the 2nd argument
- at the `date` specified in the 3rd argument.

```
chain <- function(to_rebase, basis, date_chain) {

  date_chain <- as.Date(date_chain, "%Y-%m-%d")

  valref <- basis %>%
    filter(period == date_chain) %>%
    transmute(var, value_ref = value)

  res <- to_rebase %>%
    filter(period <= date_chain) %>%
    arrange(desc(period)) %>%
    group_by(var) %>%
    mutate(growth_rate = c(1, value[-1]/lag(value)[-1])) %>%
    full_join(valref, by = "var") %>%
    group_by(var) %>%
    transmute(period, value = cumprod(growth_rate)*value_ref)%>%
    ungroup() %>%
    bind_rows(filter(basis, period > date_chain)) %>%
    arrange(period)

  return(res)

}
```

Bibliography

L Christiano, Roberto Motto, and Massimo Rostagno. Risk shocks. *American Economic Review*, 104(1):27–65, 2014. URL: http://www.aeaweb.org/articles.php?doi=10.1257/aer.104.1.27 (http://www.aeaweb.org/articles.php?doi=10.1257/aer.104.1.27), doi:10.1257/aer.104.1.27 (https://doi.org/10.1257/aer.104.1.27). ↩ 1 2 3 4 5

G. Fagan, J. Henry, and R. Mestre. An area-wide model (awm) for the euro area. *ECB Working Paper Series*, 2001. ↩

F. Smets and R. Wouters. An estimated dynamic stochastic general equilibrium model of the euro area. *Journal of the European Economic Association*, 2003. ↩ 1 2

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2016. URL: https://www.R-project.org (https://www.R-project.org). ↩

RStudio Team. *RStudio: Integrated Development Environment for R*. RStudio, Inc., Boston, MA, 2016. URL: http://www.rstudio.com/ (http://www.rstudio.com/). ↩

RELATED POSTS

Macroeconomic data for France, Germany, Italy, Spain & the Euro Area (/article/2021-02/five-countries-data/)

Description step by step to build automatic update of a macroeconomic database for France, Germany, Italy, Spain & the Euro Area.

▸ MORE (/ARTICLE/2021-02/FIVE-COUNTRIES-DATA/)

Automating update of an international database for the Euro Area (/article/2019-12/open-EA-data/)

Description step by step to build automatic update of an international database for the Euro Area.

▸ MORE (/ARTICLE/2019-12/OPEN-EA-DATA/)

Automating update of a fiscal database for the Euro Area (/article/2019-11/fipu-EA-data/)

Description step by step to build automatic update of a quarterly fiscal database for the Euro Area, similar to the Paredes, Pedregal...

▸ MORE (/ARTICLE/2019-11/FIPU-EA-DATA/)

Automating update of the Christiano, Motto and Rostagno (2014) database for the United States (/article/2016-06/cmr14-data/)

Description step by step to build automatic update of the Christiano, Motto and Rostagno (2014) database for the United States.

▸ MORE (/ARTICLE/2016-06/CMR14-DATA/)