Automating update of an international database for the Euro Area

Published on December 17, 2019 by Thomas Brand (/author/thomas-brand.html), Mateo Caicedo (/author/mateo-caicedo.html)

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

DOWNLOAD (HTTPS://GIT.NOMICS.WORLD/MACRO/OPEN-EA-DATA)

Our purpose is to create an international quarterly database for the Euro area that could be updated automatically. We want to build the following series:

- Foreign demand (without trade between Euro area countries)
- Foreign interest rate
- Oil prices
- Real effective exchange rate
- Import and export

To construct these series we use data from DBnomics (https://db.nomics.world/). The DBnomics API (https://api.db.nomics.world/) is called using the rdbnomics (https://cran.r-project.org/web/packages/rdbnomics/index.html) package. All the code is written in R, thanks to the RCoreTeam (2016) and RStudioTeam (2016).

Foreign demand

We want to build a series that describes the evolution of the foreign demand for the Eurozone, without trade between Euro area countries. We proceed in three steps:

- we calculate the growth of imports in volume of main trading partners;
- we calculate the relative importance of each trading partner in Eurozone exports;
- we sum over the growth rates of imports weighted by the relative importance of each trading partner.

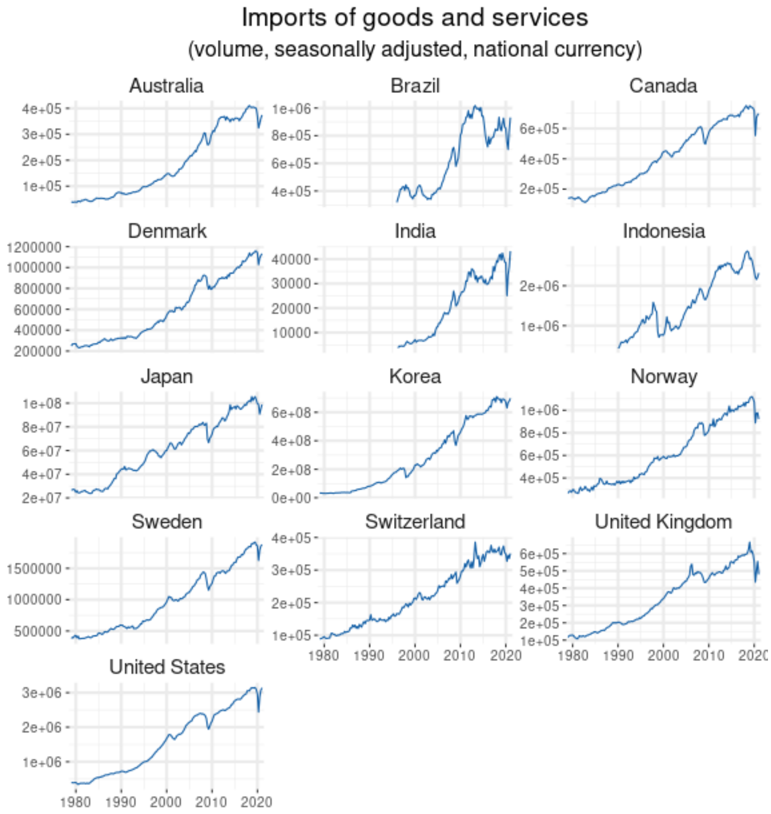
Imports of goods and services of Eurozone main commercial partners (volume, quarterly, seasonally adjusted)

First of all, we need to compute the variation of the demand originating from each trading partner of the Euro area. We select 14 trading partners that channel most of Eurozone's exports.

General case

Data comes from the OECD Economic Outlook database: we use imports of goods and services in volume.

```
partner_country_iso3 <- c('USA','GBR','DNK','NOR','SWE','CAN','CHE','JPN','AUS','B
RA','IND','IDN','KOR','CHN')
partner_country_name <- c('United-States','United-Kingdom','Denmark','Norway','Swe
den','Canada','Switzerland','Japan','Australia','Brazil','India','Indonesia','Sout
h Korea','China')
url_country_iso3 <- paste0(partner_country_iso3,collapse = "+")
filter <- paste0(url_country_iso3,".P7.VOBARSA.Q")
df <- rdb("OECD","QNA",mask=filter)

imports <-
  df %>%
  select(period,value,country=Country) %>%
  filter(year(period)>=1979) %>%
  mutate(country = plyr::mapvalues(country, from = partner_country_iso3, to = part
ner_country_name))

ggplot(imports ,aes(period,value)) +
  geom_line(colour = blueObsMacro) +
  facet_wrap(~country, ncol = 3, scales = "free_y") +
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL) +
  ggtitle("Imports of goods and services",subtitle="(volume, seasonally adjusted,
national currency)")
```



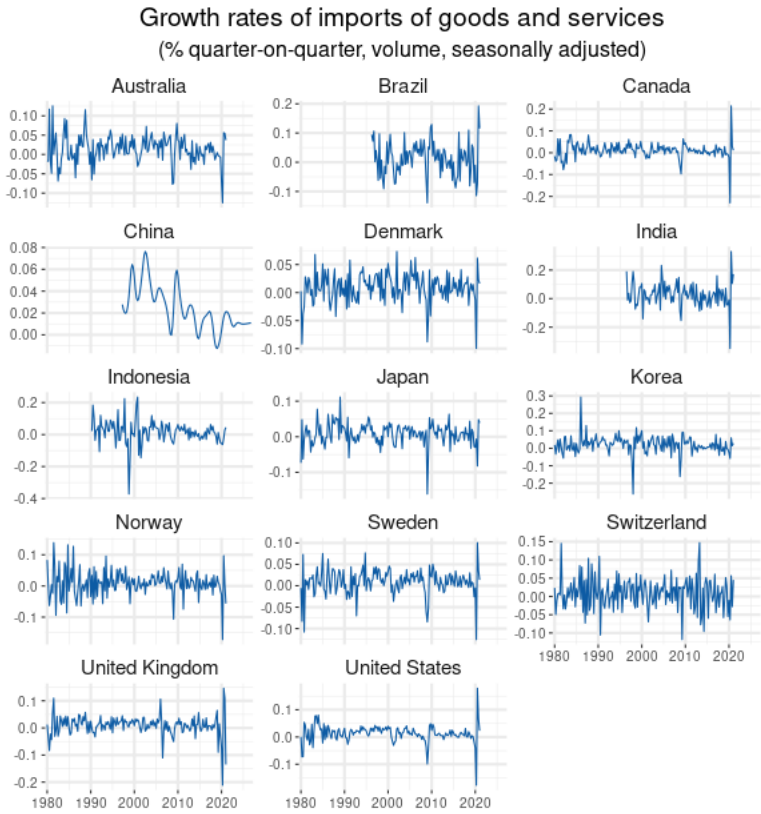
China special case

Data series of imports of goods and services from China are not available in our dataset. We decide to take Chinese imports of goods and services from the WEO database (IMF). As it is annual, we use a spline interpolation to obtain a quarterly series.

```
df <- rdb(ids="IMF/WEO:latest/CHN.TM_RPCH.pcent_change")

imports_cn <-
  df %>%
  select(period,
         value) %>%
  na.omit() %>%
  arrange(period) %>%
  mutate(value=100*cumprod(1+value/100)) %>%
  bind_rows(data.frame(period=as.Date("1997-01-01"),
                       value=100)) %>%
  arrange(period)

imports_cn_q <-
  tibble(period=seq(min(imports_cn$period),
                    length.out=nrow(imports_cn)*4,
                    by = "quarter")) %>%
  left_join(imports_cn,by="period") %>%
  mutate(value=na.spline(value),
         country="China")
```

Growth rates

```
imports_growth_rate <-
  imports %>%
  filter(country != "China") %>%
  bind_rows(imports_cn_q) %>%
  arrange(country,period) %>%
  group_by(country) %>%
  mutate(value=value/lag(value,1)-1) %>%
  ungroup() %>%
  filter(year(period)>=1980)

ggplot(imports_growth_rate,aes(period,value)) +
  geom_line(colour = blueObsMacro) +
  facet_wrap(~country, ncol = 3, scales = "free_y") +
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL) +
  ggtitle("Growth rates of imports of goods and services", subtitle="(% quarter-on
-quarter, volume, seasonally adjusted)")
```




```
Mintime <-
  imports_growth_rate %>%
  group_by(country) %>%
  summarize(MinTime = min(period)) %>%
  ungroup()
kable(Mintime)
```

country	MinTime
Australia	1980-01-01
Brazil	1996-01-01
Canada	1980-01-01
China	1997-01-01
Denmark	1980-01-01
India	1996-04-01
Indonesia	1990-01-01
Japan	1980-01-01
Korea	1980-01-01
Norway	1980-01-01
Sweden	1980-01-01
Switzerland	1980-01-01
United Kingdom	1980-01-01
United States	1980-01-01

We have uncomplete series only for Brazil, China, India and Indonesia.

Eurozone’s exports of goods to main commercial partners (values US dollars, annual)

To compute the relative importance of each trading partner, we use data series of values of exports of goods (Free on board, in US dollars), from DOT database (IMF), for each Eurozone country towards extra-area countries.

```
# Exporter countries of the Eurozone
ea_country <- c("AT","BE","R1","FR","DE","IT","LU","NL","FI","GR","IE","MT","P
T","ES","CY","SK","EE","LV","LT","SI")
ea_country_name <- c('Austria','Belgium','Luxembourg-Belgium','France','German
y','Italy','Luxembourg','Netherlands','Finland','Greece','Ireland','Malta','Portug
al','Spain','Cyprus','Slovak Republic','Estonia','Latvia','Lithuania','Slovenia')
url_ea_country <- paste0(ea_country, collapse = "+")

# Importer countries outside the Eurozone
partner_country <- c("US","GB","DK","NO","SE","CA","CH","JP","AU","BR","IN","I
D","KR","CN")
url_partner_country <- paste0(partner_country, collapse = "+")
```

```
filter <- paste0('A.',url_ea_country,'.TXG_FOB_USD.', url_partner_country)
df <- rdb("IMF","DOT",mask = filter)

bilatx <-
  df %>%
  select(exporter = REF_AREA,
         importer = COUNTERPART_AREA,
         value,
         period) %>%
  mutate(exporter = plyr::mapvalues(exporter, from = ea_country, to = ea_country_n
ame),
         importer = plyr::mapvalues(importer, from = partner_country, to = partner
_country_name)) %>%
  filter(period >= '1979-01-01')
```

The following list shows, for each Eurozone country, the date from which we have data on exports towards each one of the 14 trading partners selected. We show the begining of the sample for each country.

```
start_sample <-
  bilatx %>%
  group_by(exporter, importer) %>%
  summarize(MinTime = min(year(period))) %>%
  ungroup() %>%
  spread(importer,MinTime)

start_sample[,1:8] %>%
  kable()
```

exporter	Australia	Brazil	Canada	China	Denmark	India	Indonesia
Austria	1979	1979	1979	1979	1979	1979	1979
Belgium	1997	1997	1997	1997	1997	1997	1997
Cyprus	1979	1982	1979	1980	1979	1979	1985
Estonia	1992	1993	1993	1992	1992	1993	1993
Finland	1979	1979	1979	1979	1979	1979	1979
France	1979	1979	1979	1979	1979	1979	1979
Germany	1979	1979	1979	1979	1979	1979	1979
Greece	1979	1979	1979	1979	1979	1979	1979
Ireland	1979	1979	1979	1979	1979	1979	1979
Italy	1979	1979	1979	1979	1979	1979	1979
Latvia	1992	1994	1993	1992	1992	1992	1993
Lithuania	1992	1994	1993	1992	1992	1993	1994
Luxembourg	1997	1997	1997	1997	1997	1997	1997
Luxembourg-Belgium	1979	1979	1979	1979	1979	1979	1979
Malta	1979	1979	1979	1979	1979	1979	1981
Netherlands	1979	1979	1979	1979	1979	1979	1979
Portugal	1979	1979	1979	1979	1979	1979	1979
Slovak Republic	1993	1993	1993	1993	1993	1993	1993
Slovenia	1993	1993	1993	1993	1993	1993	1993
Spain	1979	1979	1979	1979	1979	1979	1979

```
start_sample[,c(1,9:15)] %>%
  kable()
```

exporter	Japan	Norway	South Korea	Sweden	Switzerland	United-Kingdom	United-States
Austria	1979	1979	1979	1979	1979	1979	1979
Belgium	1997	1997	1997	1997	1997	1997	1997
Cyprus	1979	1979	1980	1979	1979	1979	1979
Estonia	1992	1992	1993	1992	1992	1992	1992
Finland	1979	1979	1979	1979	1979	1979	1979
France	1979	1979	1979	1979	1979	1979	1979
Germany	1979	1979	1979	1979	1979	1979	1979
Greece	1979	1979	1979	1979	1979	1979	1979
Ireland	1979	1979	1979	1979	1979	1979	1979
Italy	1979	1979	1979	1979	1979	1979	1979
Latvia	1992	1992	1994	1992	1992	1992	1992
Lithuania	1992	1992	1993	1992	1992	1992	1992
Luxembourg	1997	1997	1997	1997	1997	1997	1997
Luxembourg-Belgium	1979	1979	1979	1979	1979	1979	1979
Malta	1979	1979	1979	1979	1979	1979	1979
Netherlands	1979	1979	1979	1979	1979	1979	1979
Portugal	1979	1979	1979	1979	1979	1979	1979
Slovak Republic	1993	1993	1993	1993	1993	1993	1993
Slovenia	1993	1993	1993	1993	1993	1993	1993
Spain	1979	1979	1979	1979	1979	1979	1979

Special case of Belgium-Luxembourg

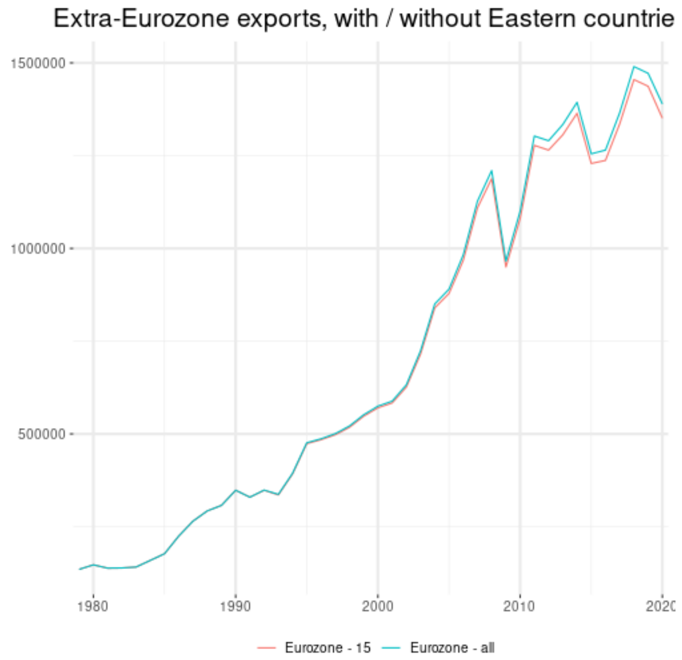
We have data for Belgium-Luxembourg as a single exporter until 1997. So we compute extra-area trade of Belgium and Luxembourg since 1997 to create a series for the whole period.

```
bilatx.Belux <-
  filter(bilatx, exporter %in% c('Belgium','Luxembourg'))  %>%
  group_by(importer, period) %>%
  summarize(value = sum(value)) %>%
  ungroup() %>%
  mutate(exporter = "Luxembourg-Belgium")

bilatx %<>%
  filter(!exporter %in% c('Belgium','Luxembourg')) %>%
  rbind(bilatx.Belux)
```

Special case of Eastern European countries

Before 1992, five countries lack some data: the Baltic states, Slovenia and Slovak Republic. On the following graph, we represent the sum of exports of the Eurozone with and without these five countries.

```
export_15 <-
  bilatx %>%
  filter(!exporter %in% c("Slovenia","Slovak Republic","Latvia","Estonia","Lithuan
ia")) %>%
  mutate(var = 'Eurozone - 15') %>%
  group_by(var, period) %>%
  summarize(value = sum(value)) %>%
  ungroup()

export_all <-
  bilatx %>%
  mutate(var = 'Eurozone - all') %>%
  group_by(var,period) %>%
  summarize(value = sum(value)) %>%
  ungroup()

plot_export <-
  rbind(export_15, export_all)

ggplot(plot_export,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("Extra-Eurozone exports, with / without Eastern countries")
```

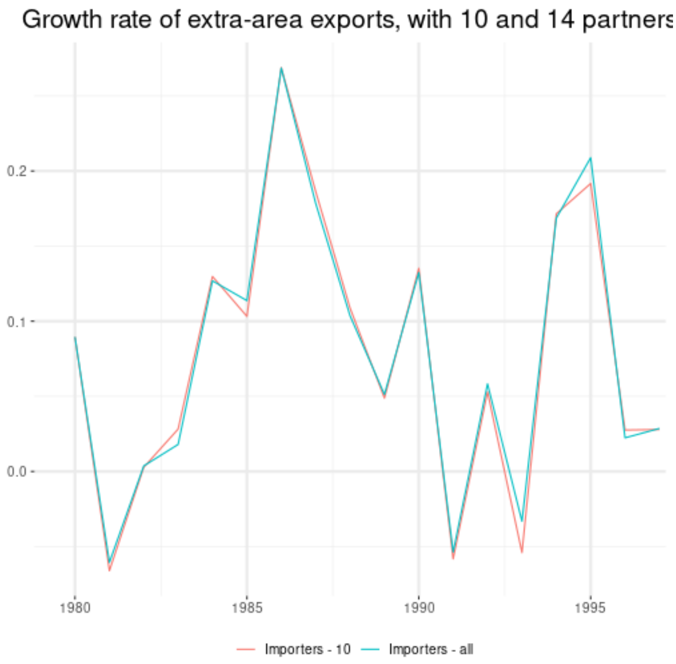


Before 2003, both series are very similar. So we choose to keep the whole dataset as it is.

Special case of Brazil, China, India and Indonesia

We saw in the previous section that we have uncomplete series of imports of goods and services for Brazil, China, India and Indonesia, with a lack of data before 1997. As these specific countries developed their imports mainly after 1997, we want to check the growth rates of extra-area exports with and without these partners before 1997.

```
import_10 <-
  bilatx %>%
  filter(!importer %in% c("Brazil","China","India","Indonesia")) %>%
  group_by(period) %>%
  summarize(value=sum(value)) %>%
  ungroup() %>%
  mutate(var= "Importers - 10")

plot_export2 <-
  bind_rows(mutate(export_all,var="Importers - all"),
            import_10) %>%
  group_by(var) %>%
  mutate(value2=value/lag(value)-1) %>%
  filter(year(period)<=1997)

ggplot(plot_export2,aes(period,value2, colour = var)) +
  geom_line() +
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL) +
  theme(legend.title=element_blank()) +
  ggtitle("Growth rate of extra-area exports, with 10 and 14 partners")
```



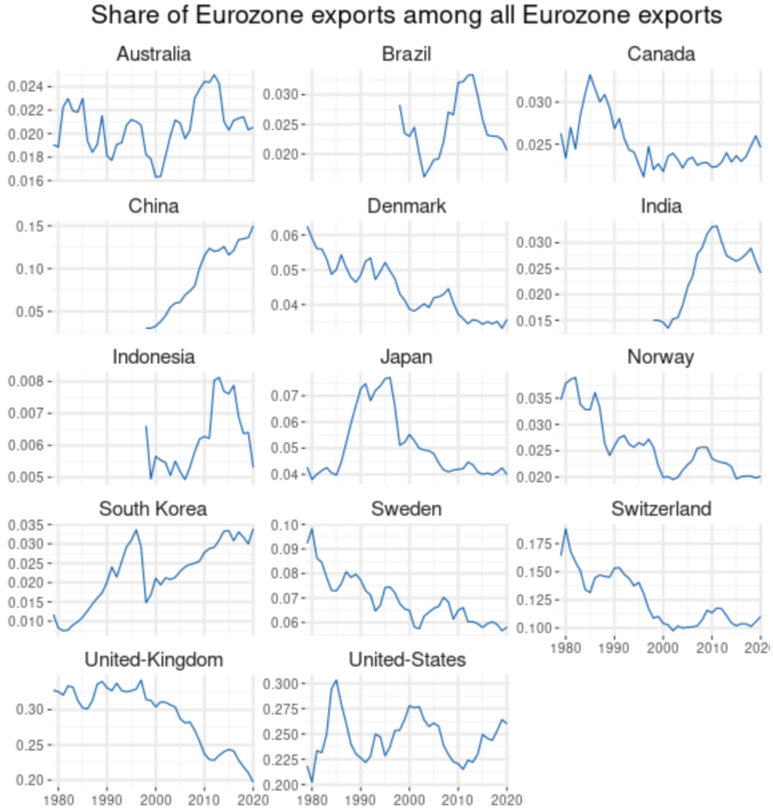
Before 1997, both series are very similar. So we choose to compute weights of 14 commercial partners after 1997 but of only 10 partner before 1997 (without Brazil, China, India and Indonesia).

Weights of main commercial partners in Eurozone's exports

For each commercial partner i, we compute α_i, the share of EA exports X among all EA exports towards these partners, at time t :

$$\alpha_{i,t} = \frac{X_{i,t}}{\sum_i X_{i,t}}$$

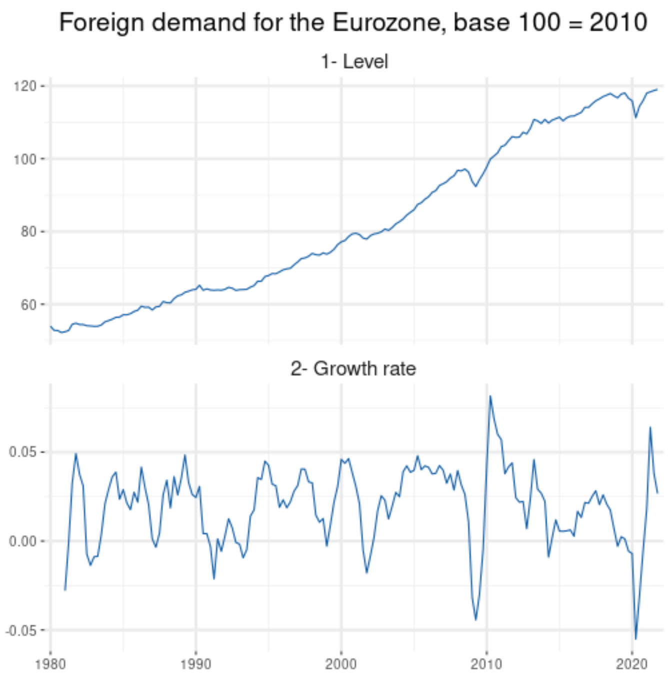
```
#Sum of exports of Euro area by importer
bilatx %<>%
  group_by(importer,period) %>%
  summarize(value = sum(value)) %>%
  ungroup()

#Sum of exports of Euro area to 14 importers
sumX_EA_importer_all <-
  bilatx %>%
  group_by(period) %>%
  summarise(xsum = sum(value)) %>%
  mutate(exporter = 'Eurozone') %>%
  ungroup()

alphas_importer_all <-
  left_join(sumX_EA_importer_all, bilatx, by = 'period') %>%
  mutate(alpha = value/xsum) %>%
  select(period,country=importer,alpha)

#Sum of exports of Euro area to 14 importers
sumX_EA_importer_10 <-
  bilatx %>%
  filter(! importer %in% c("Brazil","China","India","Indonesia")) %>%
  group_by(period) %>%
  summarise(xsum = sum(value)) %>%
  mutate(exporter = 'Eurozone') %>%
  ungroup()

alphas_importer_10 <-
  left_join(sumX_EA_importer_10,
            filter(bilatx,! importer %in% c("Brazil","China","India","Indonesi
a")),
            by = 'period') %>%
  mutate(alpha = value/xsum) %>%
  select(period,country=importer,alpha)

alphas <-
  bind_rows(
    filter(alphas_importer_10,year(period)<=1997),
    filter(alphas_importer_all,year(period)>1997)
  )

ggplot(alphas,aes(period,alpha)) +
  geom_line(colour = blueObsMacro) +
  facet_wrap(~country, ncol = 3, scales = "free_y") +
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL) +
  ggtitle("Share of Eurozone exports among all Eurozone exports")
```



Final index

We sum over the growth rates of imports in volume weighted by the relative importance of each trading partner during the previous year. Then we create a global index.

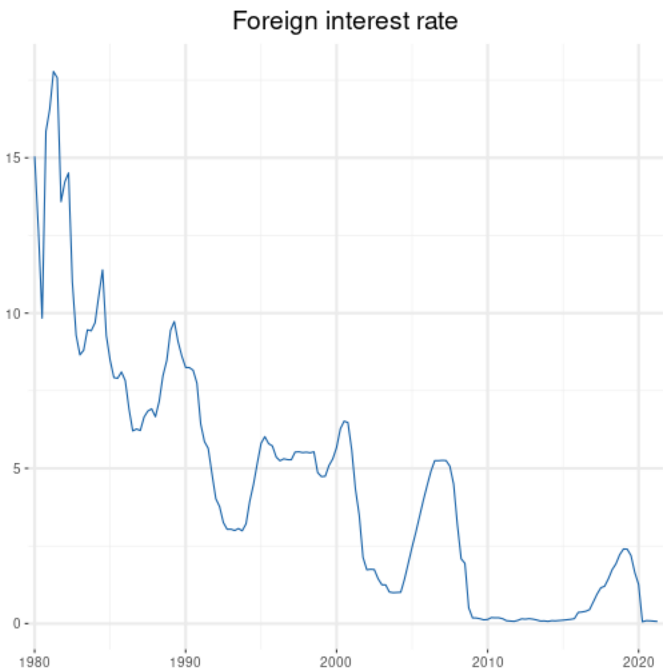
```
imports_growth_rate %<>% mutate(year=year(period))
alphas %<>% mutate(year=year(period)+1) %>%
  select(-period)

wd <-
  right_join(alphas, imports_growth_rate, by = c("year", "country")) %>%
  mutate(value = alpha * value) %>%
  na.omit() %>%
  select(period,value,country) %>%
  group_by(period) %>%
  summarise(value = sum(value)) %>%
  mutate(value = cumprod(1+value))

wd_index2010 <-
  wd %>%
  mutate(year = year(period)) %>%
  filter(year == "2010") %>%
  group_by(year) %>%
  summarize(value = mean(value)) %>%
  ungroup()

wd_index <-
  wd %>%
  mutate(period,
         value = 100*value/wd_index2010$value)

wd_index_growth <-
  wd_index %>%
  mutate(value=value/lag(value,4)-1,
         var="2- Growth rate")

plot_wd <-
  bind_rows(wd_index_growth,
            mutate(wd_index,var="1- Level"))

ggplot(plot_wd,aes(period,value)) +
  geom_line(colour = blueObsMacro) +
  facet_wrap(~var, scales = "free_y",ncol=1) +
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL) +
  ggtitle("Foreign demand for the Eurozone, base 100 = 2010")
```



Foreign interest rate

We use the US federal funds rate overnight as a proxy for the foreign interest rate.

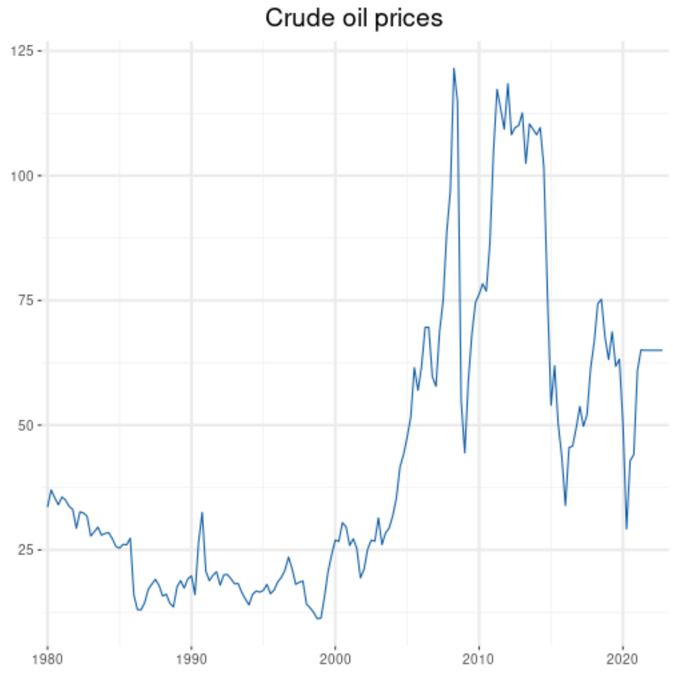
```
df <- rdb("FED","H15",mask="129.FF.O")

shortrate <-
  df %>%
  mutate(period=paste(year(period),quarter(period),sep="-")) %>%
  group_by(period) %>%
  summarise(value=mean(value)) %>%
  ungroup() %>%
  mutate(period=yq(period)) %>%
  filter(period >= "1980-01-01")

ggplot(shortrate,aes(period,value)) +
  geom_line(colour = blueObsMacro) +
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL) +
  ggtitle('Foreign interest rate')
```



Oil prices

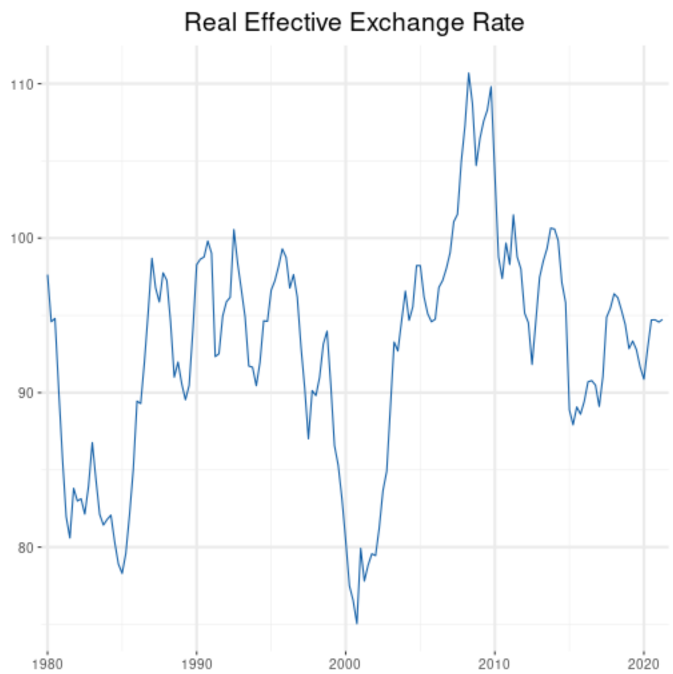
We need to download a series that reflects oil prices to build a foreign block. We take the series from the OECD Economic Outlook database.

```
df <- rdb(ids = "OECD/EO/OTO.WPBRENT.Q")

oil_prices <-
  df %>%
  select(period, value) %>%
  filter(period >= "1980-01-01")

ggplot(oil_prices,aes(period,value)) +
  geom_line(colour = blueObsMacro) +
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL) +
  ggtitle('Crude oil prices')
```



Real effective exchange rate

```
df <- rdb(ids = "BIS/eer/M.R.N.XM")

reer <-
  df %>%
  mutate(period=paste(year(period),quarter(period),sep="-")) %>%
  group_by(period) %>%
  summarize(value=mean(value)) %>%
  ungroup() %>%
  mutate(period=yq(period)) %>%
  filter(period >= "1980-01-01")

ggplot(reer,aes(period,value)) +
  geom_line(colour = blueObsMacro) +
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL) +
  ggtitle('Real Effective Exchange Rate')
```



Extra Euro area imports and exports

```
df <- rdb("ECB","TRD",mask = 'M.I8.Y.M+X.TTT.J8.4.VOX')

trade <-
  df %>%
  transmute(period=paste(year(period),quarter(period),sep="-"),
            value,
            var=ifelse(grepl("Import",series_name),"imports","exports")) %>%
  group_by(var,period) %>%
  summarize(value = mean(value)) %>%
  ungroup() %>%
  mutate(period=yq(period))

ggplot(trade,aes(period,value)) +
  geom_line(colour = blueObsMacro) +
  facet_wrap(~var)+
  scale_x_date(expand = c(0.01,0.01)) +
  theme + xlab(NULL) + ylab(NULL) +
  ggtitle('Extra euro area imports / exports, in volume, seasonally adjusted')
```

Final international database for the Euro area

We eventually build an international database for the Euro area.

```
rawdata <-
  bind_rows(
    mutate(wd_index, var = 'world_demand'),
    mutate(shortrate, var = 'foreign_rate'),
    mutate(oil_prices, var = 'oil_prices'),
    mutate(reer, var = "reer"),
    trade)
```

We can check the last date available for each variable.

```
maxDate <-
  rawdata %>%
  group_by(var) %>%
  summarize(maxdate=max(period)) %>%
  arrange(maxdate)
kable(maxDate)
```

var	maxdate
exports	2021-04-01
foreign_rate	2021-04-01
imports	2021-04-01
reer	2021-04-01
world_demand	2021-10-01
oil_prices	2022-10-01

```
minmaxDate <- min(maxDate$maxdate)

EA_Open_rawdata <-
  rawdata %>%
  filter(period <= minmaxDate) %>%
  spread(var,value)

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

So we filter the database until 2021 Q2. You can download all the raw series here (http://shiny.cepremap.fr/data/EA_Open_rawdata.csv).

```
sw03 <-
  read.csv("http://shiny.cepremap.fr/data/EA_SW_rawdata.csv") %>%
  mutate(period=ymd(period))

EA_Open_data <-
  EA_Open_rawdata %>%
  inner_join(sw03,by="period") %>%
  transmute(period,
            world_demand,
            foreign_rate,
            oil_prices,
            reer,
            imports,
            exports)

EA_Open_data %>%
  mutate(period=gsub(" ","",as.yearqtr(period))) %>%
  write.csv(file = "EA_Open_data.csv",row.names = FALSE)
```

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

Appendix

Chain

This function chain two series, using the growth rate of the historical one to deduce new points on the original series. It allows to go further back in time with one series while keeping the most recent points.

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

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

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

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

  return(res)

}
```

Bibliography

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 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/)

Automating update of a finance database for the Euro Area (/article/2016-06/cmr14-EA-data/)

Description step by step to build automatic update a finance database for the Euro Area, similar to the Christiano, Motto and Rostagno...

▸ MORE (/ARTICLE/2016-06/CMR14-EA-DATA/)

Follow us

Gitlab (https://git.nomics.world/macro)

Twitter (https://twitter.com/obsmacro)

Links

Dynare (http://www.dynare.org/)

DBnomics (https://db.nomics.world/)

R-bloggers (http://www.r-bloggers.com/)