Testing a very basic function in R

Default chunk options

Required libraries

```
library(RPostgreSQL)
```

```
## Loading required package: DBI
```

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------- tidyverse 1.
```

```
## ✓ ggplot2 2.2.1     ✓ purrr   0.2.4
## ✓ tibble  1.4.1     ✓ dplyr   0.7.4
## ✓ tidyr   0.7.2     ✓ stringr 1.2.0
## ✓ readr   1.1.1     ✓ forcats 0.2.0
```

```
## -- Conflicts ------------------------------------------------------------------------- tidyverse_conflict
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(dbplyr)
```

```
##
## Attaching package: 'dbplyr'
```

```
## The following objects are masked from 'package:dplyr':
##
##     ident, sql
```

```
library(rjson)
library(DBI)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
library(tibble)
library(olsrr)
```

```
##
## Attaching package: 'olsrr'
```

```
## The following object is masked from 'package:datasets':
##
##     rivers
```

```
library(ggplot2)
library(ggExtra)
library(gridExtra)
```

Figure 1: A sea cucumber in all its glory. This creature kills hundreds of people every year.

```
##
## Attaching package: 'gridExtra'
## The following object is masked from 'package:dplyr':
##
##     combine
```

Get the auxiliary data

```
source("get_HMRC_aux_data.R")
list1 <- get_HMRC_aux_data()
comcode <- data.frame(Reduce(rbind, list1[1]))
port    <- data.frame(Reduce(rbind, list1[2]))
country <- data.frame(Reduce(rbind, list1[3]))
write.csv(comcode,file="comcode.csv")
write.csv(country,file="country.csv")
```

Find the comcodes for

- Chicken
- Beef
- Cucumbers (watch out, beacuse there are *sea cucumbers*!)

```
cc_chicken      <- comcode[grep('CHICKEN',   toupper(comcode$description)),]
cc_all_cucumber <- comcode[grep('CUCUMBER',  toupper(comcode$description)),]
```

```
#cc_cucumber     <- cc_all_cucumber[grep('VEGETABLES',toupper(cc_all_cucumber$description)),]
cc_beef         <- comcode[grep('BEEF',      toupper(comcode$description)),]
```

This is Warren's magic with a little bit of extra work

Is ten minutes too long? Then load the csv files written at the end of this notebook

```
source("get_Comtrade_data.R")
stime <- Sys.time()
polish_chicken   <- get_Comtrade_data(201001,201601,"default","02071","616")
spanish_cucumber <- get_Comtrade_data(201001,201601,"default","070700","724")
brazilian_beef   <- get_Comtrade_data(201001,201601,"default","160250","76")
etime <- Sys.time()
(etime-stime)
```

```
## Time difference of 9.38858 mins
```

Removing irrelevant variables (columns)

```
polish_chicken   <- polish_chicken   %>% select(-reporter_code,-partner,-partner_code)
spanish_cucumber <- spanish_cucumber %>% select(-reporter_code,-partner,-partner_code)
brazilian_beef   <- brazilian_beef   %>% select(-reporter_code,-partner,-partner_code)
```

In case the same 'product' comes under several commodity codes, add them together:

For instance: different chicken cuts have different commodity codes.

```
polish_chicken   <- polish_chicken %>%
                    group_by(trade_flow,reporter,period) %>%
                    summarize(net_weight_kg   = sum(netweight_kg),
                              trade_value_usd = sum(trade_value_usd)) %>% ungroup()
spanish_cucumber <- spanish_cucumber %>%
                    group_by(trade_flow,reporter,period) %>%
                    summarize(net_weight_kg   = sum(netweight_kg),
                              trade_value_usd = sum(trade_value_usd)) %>% ungroup()
brazilian_beef   <- brazilian_beef %>%
                    group_by(trade_flow,reporter,period) %>%
                    summarize(net_weight_kg   = sum(netweight_kg),
                              trade_value_usd = sum(trade_value_usd)) %>% ungroup()
```

Get the price in usd per kilogram

```
polish_chicken   <- polish_chicken   %>% mutate(price_usd_kg = trade_value_usd/net_weight_kg)
spanish_cucumber <- spanish_cucumber %>% mutate(price_usd_kg = trade_value_usd/net_weight_kg)
brazilian_beef   <- brazilian_beef   %>% mutate(price_usd_kg = trade_value_usd/net_weight_kg)
```

Refurbish the date into something R understand

```
polish_chicken   <- polish_chicken %>%
                    mutate(period_date = ymd(paste(period,"01",sep="")))
spanish_cucumber <- spanish_cucumber %>%
                    mutate(period_date = ymd(paste(period,"01",sep="")))
brazilian_beef   <- brazilian_beef %>%
                    mutate(period_date = ymd(paste(period,"01",sep="")))
```

Clean the data by removing incomplete cases

Use simpler nomenclature for each data frame

```
polc <- polish_chicken[complete.cases(polish_chicken),]
spac <- spanish_cucumber[complete.cases(spanish_cucumber),]
brab <- brazilian_beef[complete.cases(brazilian_beef),]
```

Restrict data to imports

```
polci <- polc %>% filter(trade_flow=="Imports")
spaci <- spac %>% filter(trade_flow=="Imports")
brabi <- brab %>% filter(trade_flow=="Imports")
```

Searching for outstanding values

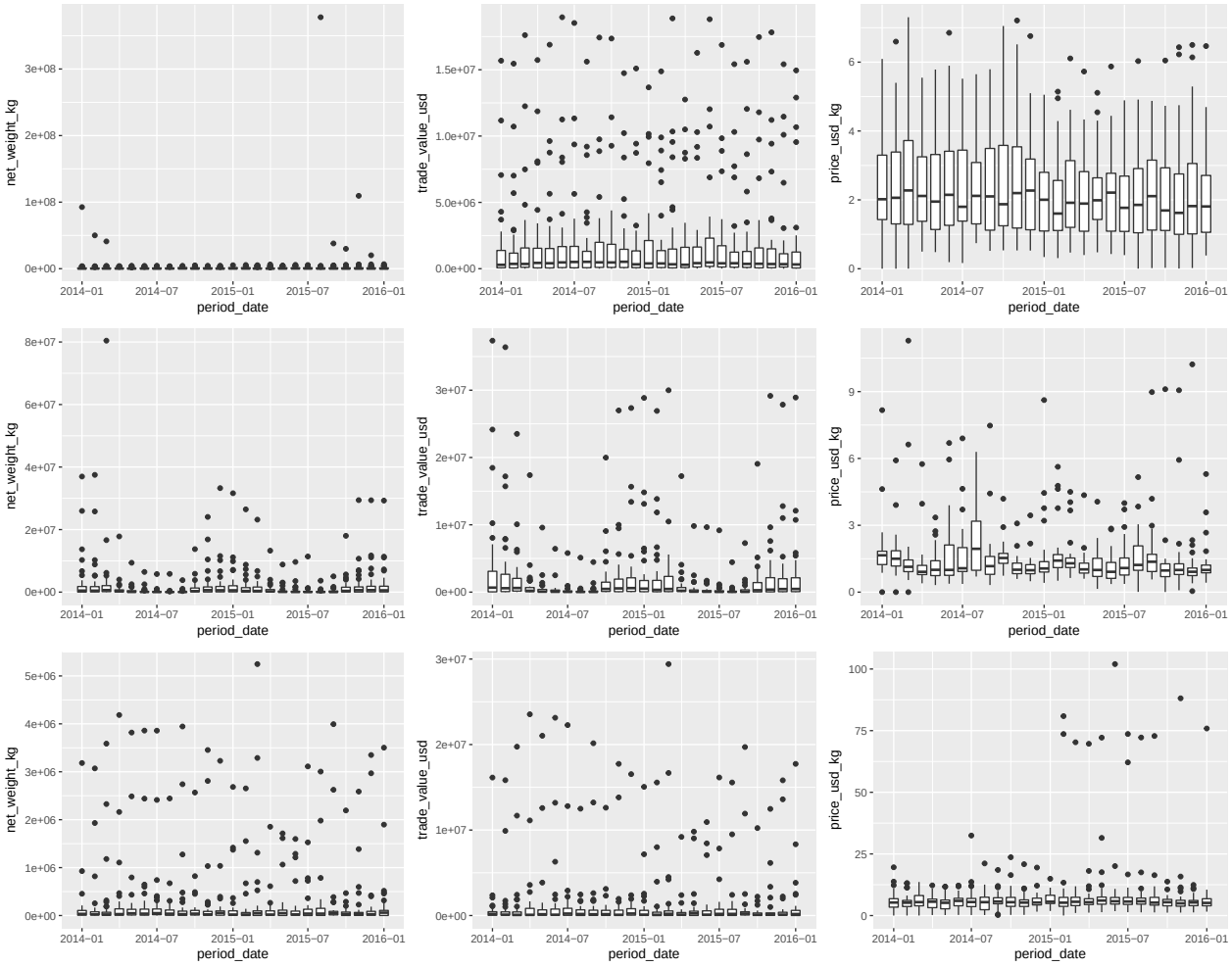
In box plots, the line in the blox represent the median of the data.

The box spans over the IQR, i.e. the 25 and 75% percentiles.

Outliers are located at distanced larger than 1.5 IQR and are represented by dots.

Why do we need the bloxplots of the new weight and trade value (extensive magnitudes)? Erase them... The boxplot of the price shows are relatively homogeneity in prices for polish chicken *The opposite is true for the brazilian beef with many huge outliers* Spanish cucumbers are somehow in between these two extremes *The most imported food (total net weight) is the spanish cucumber* ... followed by the polish chicken and brazilian beef

```
tmp <- polci
p1 <- ggplot(data=tmp) +
  geom_boxplot(mapping = aes(x=period_date,y=net_weight_kg,group=period_date))
p2 <- ggplot(data=tmp) +
  geom_boxplot(mapping = aes(x=period_date,y=trade_value_usd,group=period_date))
p3 <- ggplot(data=tmp) +
  geom_boxplot(mapping = aes(x=period_date,y=price_usd_kg,group=period_date))
tmp <- spaci
p4 <- ggplot(data=tmp) +
  geom_boxplot(mapping = aes(x=period_date,y=net_weight_kg,group=period_date))
p5 <- ggplot(data=tmp) +
  geom_boxplot(mapping = aes(x=period_date,y=trade_value_usd,group=period_date))
p6 <- ggplot(data=tmp) +
  geom_boxplot(mapping = aes(x=period_date,y=price_usd_kg,group=period_date))
tmp <- brabi
p7 <- ggplot(data=tmp) +
  geom_boxplot(mapping = aes(x=period_date,y=net_weight_kg,group=period_date))
p8 <- ggplot(data=tmp) +
  geom_boxplot(mapping = aes(x=period_date,y=trade_value_usd,group=period_date))
p9 <- ggplot(data=tmp) +
  geom_boxplot(mapping = aes(x=period_date,y=price_usd_kg,group=period_date))
grid.arrange(p1,p2,p3,p4,p5,p6,p7,p8,p9,ncol=3,nrow=3)
```



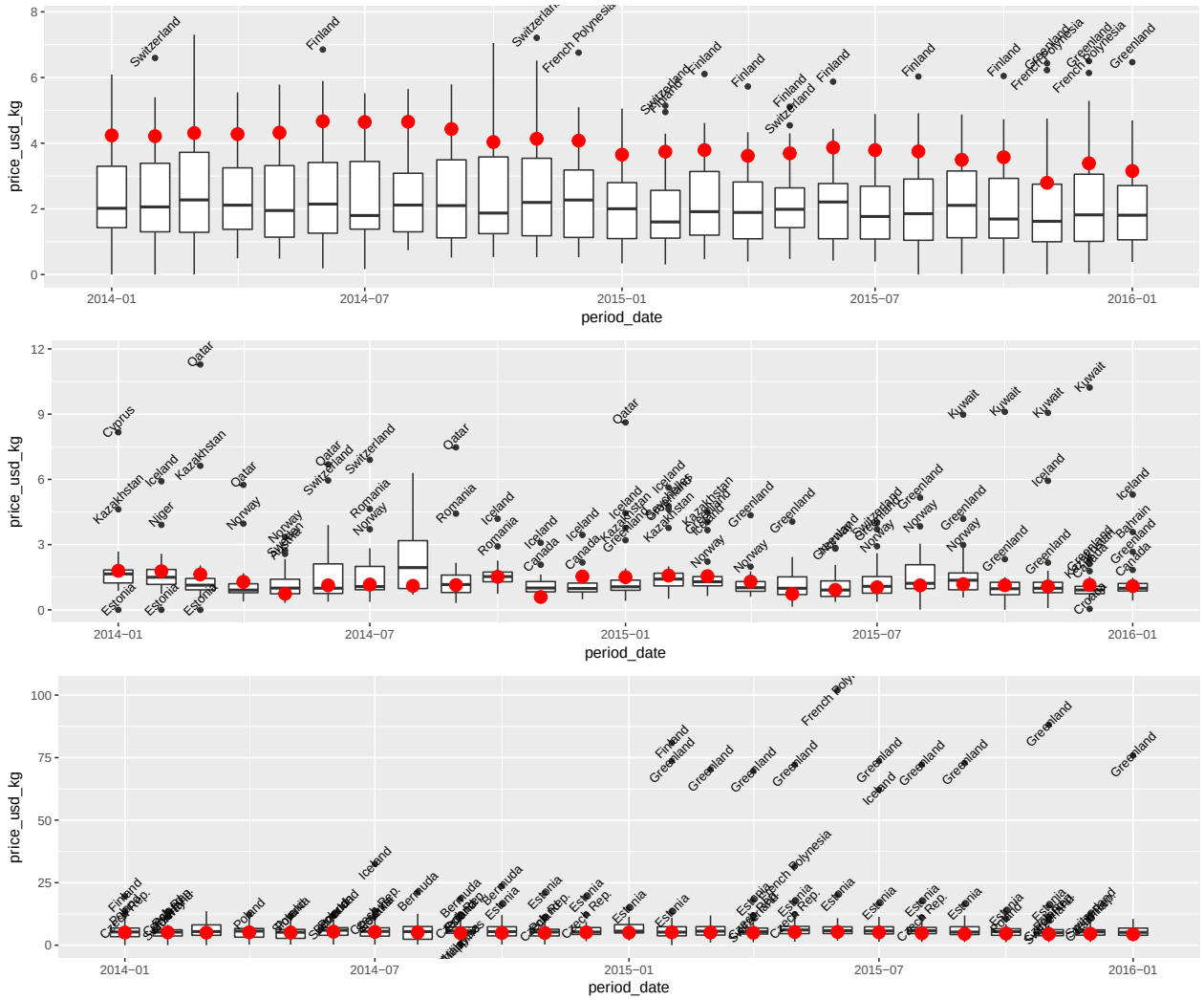
Get plots comparing prices between UK and World

These plots are almost identical but provide additional information

The red dots in the boxplot correspond to UK data

Many outliers correspond to countries with elevated transport costs This includes Finland, Greenland, French polynesia, Iceland, Cyprus *Also, several middle east countries: Qatar, Kuwait* #No seasonality in the price (although clear in net weight and trade value) #*UK pays more for the polish chicken but the same for spanish cucumbers and brazilian beef

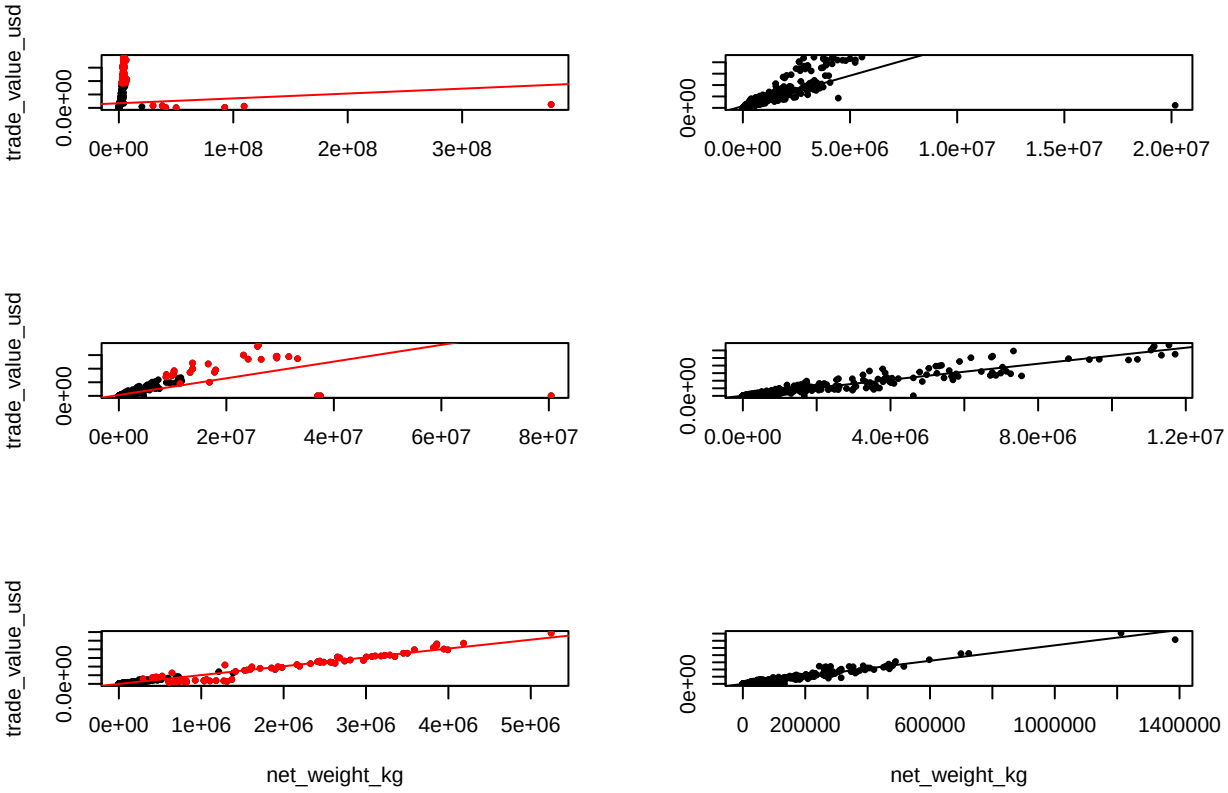
```
source("make_global_plots.R")
some_plots <- make_global_plots(polci)
#p1 <- some_plots[[1]] + theme(legend.position = "bottom")
p2 <- some_plots[[2]]
#
some_plots <- make_global_plots(spaci)
#p3 <- some_plots[[1]] + theme(legend.position = "bottom")
p4 <- some_plots[[2]]
#
some_plots <- make_global_plots(brabi)
#p5 <- some_plots[[1]] + theme(legend.position = "bottom")
p6 <- some_plots[[2]]
#grid.arrange(p1,p2,p3,p4,p5,p6,ncol=2,nrow=3)
grid.arrange(p2,p4,p6,nrow=3)
```



Another look at the influence measures (outliers and leverag points)

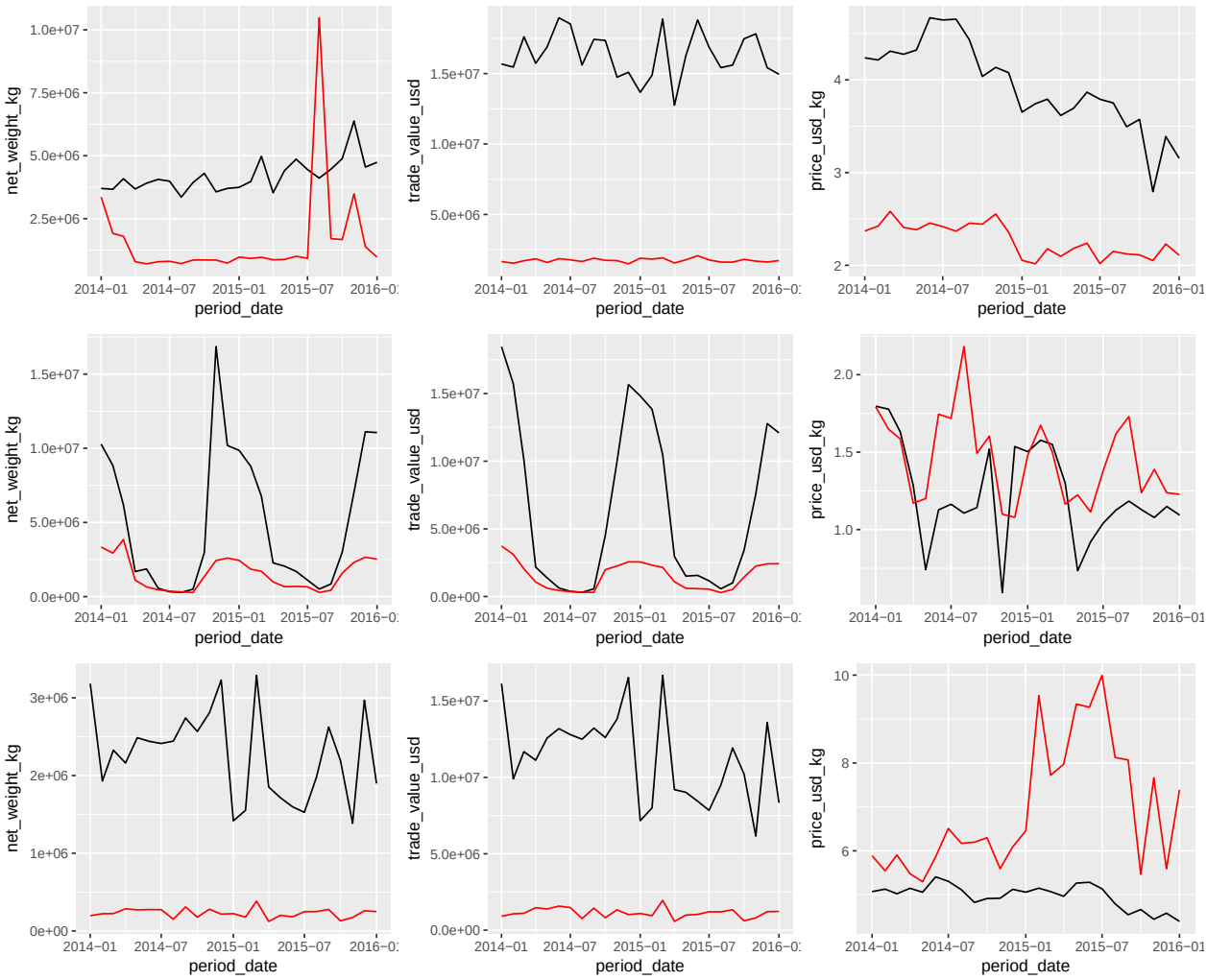
```
par(mfrow=c(3,2))
tmp <- polci
fitdata <- lm(trade_value_usd ~ net_weight_kg,tmp)
infl <- influence.measures(fitdata)
caca <- which(apply(infl$is.inf, 1, any))
cc <- tmp[caca,]
plot(data=tmp,trade_value_usd ~ net_weight_kg,cex=0.5,pch=19,xlab='')
points(data=cc,trade_value_usd ~ net_weight_kg,cex=0.5,pch=19,col="red")
tmp2 <- tmp[-caca,]
fitdata2 <- lm(trade_value_usd ~ net_weight_kg,tmp2)
abline(fitdata,col="red")
plot(data=tmp2,trade_value_usd ~ net_weight_kg,cex=0.5,pch=19,xlab='',ylab='')
abline(fitdata2)
#
tmp <- spaci
fitdata <- lm(trade_value_usd ~ net_weight_kg,tmp)
```

```r
infl <- influence.measures(fitdata)
caca <- which(apply(infl$is.inf, 1, any))
cc <- tmp[caca,]
plot(data=tmp,trade_value_usd ~ net_weight_kg,cex=0.5,pch=19,xlab='')
points(data=cc,trade_value_usd ~ net_weight_kg,cex=0.5,pch=19,col="red")
tmp2 <- tmp[-caca,]
fitdata2 <- lm(trade_value_usd ~ net_weight_kg,tmp2)
abline(fitdata,col="red")
plot(data=tmp2,trade_value_usd ~ net_weight_kg,cex=0.5,pch=19,xlab='',ylab='')
abline(fitdata2)
#
tmp <- brabi
fitdata <- lm(trade_value_usd ~ net_weight_kg,tmp)
infl <- influence.measures(fitdata)
caca <- which(apply(infl$is.inf, 1, any))
cc <- tmp[caca,]
plot(data=tmp,trade_value_usd ~ net_weight_kg,cex=0.5,pch=19)
points(data=cc,trade_value_usd ~ net_weight_kg,cex=0.5,pch=19,col="red")
tmp2 <- tmp[-caca,]
fitdata2 <- lm(trade_value_usd ~ net_weight_kg,tmp2)
abline(fitdata,col="red")
plot(data=tmp2,trade_value_usd ~ net_weight_kg,cex=0.5,pch=19,ylab='')
abline(fitdata2)
```



Temporal evolution of the UK trade

```
tmp <- polci
p1 <- ggplot(data=tmp %>%
         filter(reporter=="United Kingdom"),aes(x=period_date,y=net_weight_kg)) + geom_line() +
         geom_line(data=tmp %>% group_by(period_date) %>% summarize(caca=mean(net_weight_kg)),
                   aes(x=period_date,y=caca),color="red")
p2 <- ggplot(data=tmp %>%
         filter(reporter=="United Kingdom"),aes(x=period_date,y=trade_value_usd)) + geom_line() +
         geom_line(data=tmp %>% group_by(period_date) %>% summarize(caca=mean(trade_value_usd)),
                   aes(x=period_date,y=caca),color="red")
p3 <- ggplot(data=tmp %>%
         filter(reporter=="United Kingdom"),aes(x=period_date,y=price_usd_kg)) + geom_line() +
         geom_line(data=tmp %>% group_by(period_date) %>% summarize(caca=mean(price_usd_kg)),
                   aes(x=period_date,y=caca),color="red")
#
tmp <- spaci
p4 <- ggplot(data=tmp %>%
         filter(reporter=="United Kingdom"),aes(x=period_date,y=net_weight_kg)) + geom_line() +
         geom_line(data=tmp %>% group_by(period_date) %>% summarize(caca=mean(net_weight_kg)),
                   aes(x=period_date,y=caca),color="red")
p5 <- ggplot(data=tmp %>%
         filter(reporter=="United Kingdom"),aes(x=period_date,y=trade_value_usd)) + geom_line() +
         geom_line(data=tmp %>% group_by(period_date) %>% summarize(caca=mean(trade_value_usd)),
                   aes(x=period_date,y=caca),color="red")
p6 <- ggplot(data=tmp %>%
         filter(reporter=="United Kingdom"),aes(x=period_date,y=price_usd_kg)) + geom_line() +
         geom_line(data=tmp %>% group_by(period_date) %>% summarize(caca=mean(price_usd_kg)),
                   aes(x=period_date,y=caca),color="red")
#
tmp <- brabi
p7 <- ggplot(data=tmp %>%
         filter(reporter=="United Kingdom"),aes(x=period_date,y=net_weight_kg)) + geom_line() +
         geom_line(data=tmp %>% group_by(period_date) %>% summarize(caca=mean(net_weight_kg)),
                   aes(x=period_date,y=caca),color="red")
p8 <- ggplot(data=tmp %>%
         filter(reporter=="United Kingdom"),aes(x=period_date,y=trade_value_usd)) + geom_line() +
         geom_line(data=tmp %>% group_by(period_date) %>% summarize(caca=mean(trade_value_usd)),
                   aes(x=period_date,y=caca),color="red")
p9 <- ggplot(data=tmp %>%
         filter(reporter=="United Kingdom"),aes(x=period_date,y=price_usd_kg)) + geom_line() +
         geom_line(data=tmp %>% group_by(period_date) %>% summarize(caca=mean(price_usd_kg)),
                   aes(x=period_date,y=caca),color="red")
grid.arrange(p1,p2,p3,p4,p5,p6,p7,p8,p9,ncol=3,nrow=3)
```


net_weight_kg
trade_value_usd
price_usd_kg
period_date
2014–01
2014–07
2015–01
2015–07
2016–01