

## IRF

CLMENT CARRIER

### SHOCK RESPONSE

Consider a VAR(1) in difference where the parameters have been estimated:

$$\Delta Y_t = \Delta Y_{t-1} \hat{A} + \epsilon_t,$$

To compute the response to an exogenous shock of size  $s$  to the first variable in  $Y$ , construct the vector  $\delta = [s, 0, \dots, 0]$ . The response at after  $h$  periods is given by  $\delta \hat{A}^h$ .

- For models with multiple lags the companion form (or simply recursive computations) can be used.
- For models with exogenous variables, at path for the exogenous variables has to be specified. In general it should be equal to zero or to a random walk forecast, at least in periods following an initial shock.
- The deterministics can be omitted.
- For VECM models, things are more complicated but not much more difficult.

### CONDITIONAL FORECASTS

The options so far:

- Create a model for the exogenous variable.
- Use naïve methods (i.e. RW).
- Compute prediction density for the exogenous variable and plug in the model.  
Gaussian + linear = Gaussian.
- Condition on true value.

### APPLICATION SUR R

```
require(lassovar)
## Loading required package: lassovar
require(ggplot2)
## Loading required package: ggplot2
require(reshape2)
## Loading required package: reshape2
require(urca)
## Loading required package: urca
require(MSBVAR)
## Loading required package: MSBVAR
## ##
## ## MSBVAR Package v.0.9-2
## ## Build date: Wed Jul 15 16:49:58 2015
## ## Copyright (C) 2005-2015, Patrick T. Brandt
## ## Written by Patrick T. Brandt
```

Date: July 15, 2015.

```
## ##
## ## Support provided by the U.S. National Science Foundation
## ## (Grants SES-0351179, SES-0351205, SES-0540816, and SES-0921051)
## ##
##
##
## Attaching package: 'MSBVAR'
##
## The following object is masked from 'package:urca':
##
##      summary
```

```
load("vardata2")
```

I keep variables from Q1 1998 :

```
data<-subset(vardataframe[117:180,])
```

First difference of all series :

```
difdata <- tail(data,-1) - head(data,-1)
```

## SHOCK RESPONSE

Function for computing the IRF :

```
forecast<-function(data,lag,horizon,choc){
  fore<-matrix(0,nrow=dim(data)[2],ncol=horizon)
  lv<-lassovar(dat=data,lags=lag, ic="BIC")
  coeff<-lv$coefficients[-1,]
  for (i in 1:horizon){
    fore[,i]<-coeff^i*%choc
  }
  return(t(fore))
}
```

## CONDITIONAL SHOCK

Function to generate RW for exogenous variables. I fixed the variance of the shock equal to the variance of oil price index on the sample. The reason is that, in the following IRF I only use oil price as exogenous variable. I need to modify the function.

```
rw<-function(t,x){
  y<-matrix(0,dim(x)[2],t)
  y[,1]<-t(x)
  for(i in 2:t){
    y[,i] <- y[,i-1] + matrix(rnorm(dim(x)[2],0,0.5),dim(x)[2],1)
  }
  return(y[, -1])
}
```

Function to compute IRF to RW evolution of exogenous variables

```
conditional<-function(exogen,data,lag,horizon,preforecast){
  all=data.frame(exogen,data)
  fore<-matrix(0,nrow=dim(data)[2]+dim(exogen)[2],ncol=horizon+preforecast)
  fore[,1:(preforecast)]<-t(all[(dim(all)[1]-preforecast+1):dim(all)[1],])
  fore[1:dim(exogen)[2],-c(1:(preforecast))]<-rw(horizon+1,as.matrix(exogen[dim(exogen)[1]:dim(exogen)[2],]))
  lv<-lassovar(dat=all,lags=lag, ic="BIC")
  coeff<-as.matrix(t(lv$coefficients[-1,]),26,26)
  intercept<-as.matrix(lv$coefficients[1,],26,1)
  for (i in (preforecast+1):(horizon+preforecast)){
    fore[-dim(exogen)[2],i]<-intercept[-dim(exogen)[2],]+(coeff%*%fore[,i-1])[-dim(exogen)[2],]
  }
  rownames(fore)<-names(all)
  return(t(fore))
}
```

### APPLICATION WITH OIL PRICE AS EXOGENOUS VARIABLE

I transform the HICP index in an inflation rate (year-on-year)

```
HICP<-vardataframe[113:180,6]

inflation<-NULL
for (i in seq(from=5,to=length(HICP),by=1)){
  inflation[i]==-1+exp(HICP[i])/exp(HICP[i-4])
}

inflation<-inflation[5:68]*100

data$HICP<-NULL
data$inflation<-inflation
```

I choose oil price as exogenous variable.

```
exo<-data.frame(data$POILU)
colnames(exo)<- 'POILU'
end<-subset(data,~which(names(data) %in% c("POILU")))
```

I iterate 100 random walk scenario for oil price, and I keep the prediction of inflation.

```
iter<-100
preforecast<-16
horizon<-12

HICPpred<-matrix(0,horizon+preforecast,iter)
for (i in 1:iter){
  HICPpred[,i]<-matrix(conditional(exo,end,1,horizon,preforecast)[,"inflation"])
}
```

```
HICPpred<-data.frame(HICPpred)
HICPpred$time<-seq(as.Date("2010/01/01"), as.Date("2016/12/31"), by = "quarter")
var <- melt(HICPpred, id = 'time', variable.name = 'series')
```

```
ggplot(var, aes(time,value, col=series)) + geom_point()
```

