Tarea 4 - Modelos de series temporales

Jordi Vanrell Forteza

30/6/2021

library(tidyverse)  
library(vars)  
library(cowplot)  
library(scales)

La tarea requiere escoger 5 archivos VAR\*.txt de entre la lista de 10 ofrecidos, pertenecientes a activo del IBEX-35. Cada uno de ellos contiene datos de cotizaciones, asociadas o no con transacciones, de todas las sesiones del mes de diciembre de 2011. Se escogen los activos al azar con la semilla 603, se leen con un bucle y se almacenan en una lista.

set.seed(603)  
idx <- sort(as.character(sample(1:10, size = 5, replace = F)), decreasing = F)  
  
VAR <- vector(mode = "list", length = length(idx))  
for (i in idx){ # Bucle de lectura de los archivos; se almacenan en lista  
 VAR[[which(idx==i)]] <- read.table(paste0("VAR", i,".txt"), header = T)  
 rm(i)  
}

La lista consiste en 5 tablas de datos ordenadas según el índice de posición del vector idx = (“4”, “5”, “7”, “8”, “9”).

# 1. Estadística descriptiva

Se requieren algunos indicadores descriptivos de los archivos escogidos. Concretamente:

* El número de cambios del punto medio por día (QC), que se calcula como la media diaria de veces que el cambio del punto medio no es 0.
* El número de transacciones (NT), que se halla sobre la cantidad total de veces que el signo de la transacción es distinto de 0. La magnitud que se ofrece es absoluta; la diaria podría obtenerse dividiendo la absoluta entre el número de sesiones. En todo caso, como se dispone de datos de todos los activos para las mismas sesiones, la magnitud absoluta es igualmente válida para comparación.
* El tamaño medio de las transacciones (MS), que se calcula sobre todos los datos del mes como la suma del volumen de todas las transacciones entre el número de registros cuya variable *sign* reporta un valor distinto de 0.
* El volumen medio negociado diario (MV), que se determina como la media aritmética de la suma diaria del volumen de todas las operaciones.

tabla\_i <- data.frame()  
for (d in 1:length(VAR)){  
 day <- unique(VAR[[d]]$day)  
 pre\_subtabla\_i <- data.frame()  
 for (t in day){  
 VAR\_day <- VAR[[d]] %>%  
 filter(day == t) %>%  
 mutate(dqmp = qmp - lag(qmp))  
 row = data.frame(QC = sum(table(VAR\_day$dqmp[VAR\_day$dqmp != 0])),  
 MV = sum(VAR\_day$size[VAR\_day$size != 0]))  
 pre\_subtabla\_i <- rbind(pre\_subtabla\_i, row)  
 }  
 pre\_subtabla\_i <- pre\_subtabla\_i %>%  
 summarise(QC = mean(QC, na.rm = T),  
 MV = mean(MV, na.rm = T)) %>%  
 # NT son las transacciones del mes, no la media diaria.  
 # Como NT, MS se calcula sobe todo el mes.  
 mutate(NT = sum(table(VAR[[d]]$sign[VAR[[d]]$sign != 0])),  
 MS = mean(VAR\_day$size[VAR\_day$size != 0], na.rm = T))  
 tabla\_i <- rbind(tabla\_i, pre\_subtabla\_i)  
 rm(d, t, row, pre\_subtabla\_i, VAR\_day)  
}  
rownames(tabla\_i) <- paste0("VAR",idx)

# Por cuestiones de compilación la tabla (esta y las siguientes) se ha generado   
# de forma separada, de acuerdo con el código que sigue. Este es un código que   
# NO puede compilarse en Rmd.  
library(officer)  
library(flextable)  
library(magrittr)  
# Para exportar tablas a formato Word  
ft <- flextable(data = tabla\_i %>% add\_rownames()) %>%   
 theme\_zebra %>% autofit  
ft <- set\_caption(ft, caption = "Tabla I", style = "Table Caption")  
# Crea un archivo temp  
tmp <- tempfile(fileext = ".docx")  
# Crea un documento docx  
read\_docx() %>% body\_add\_flextable(ft) %>% print(target = tmp)  
browseURL(tmp) # abre el documento

**Tabla I**

|  | **QC** | **MV** | **NT** | **MS** |
| --- | --- | --- | --- | --- |
| VAR4 | 3,439.05 | 311,389.4 | 17,985 | 384.72 |
| VAR5 | 5,368.38 | 202,776.1 | 12,261 | 348.61 |
| VAR7 | 10,147.43 | 935,002.4 | 82,499 | 193.15 |
| VAR8 | 12,300.91 | 4,225,807.7 | 108,984 | 602.95 |
| VAR9 | 36,790.19 | 30,250,125.7 | 215,919 | 2,412.90 |

# 2. Estimación del VAR estructural de Hasbrouck (1991a)

En este apartado se requiere la estimación para cada activo de un VAR estructural de Hasbrouck (1991a) utilizando, para la representación de la negociación, (a) el signo de las transacciones y (b) el tamaño de las transacciones (multiplicado por su signo correspondiente). Al respecto, se indican una serie de consideraciones a tener en cuenta:

* Se requiere que los cambios en el punto medio se calculen como rentabilidades en capitalización continua.
* Debe escogerse el orden del VAR de manera óptima, utilizando un criterio como el de Akaike, con un máximo de 10 retardos.
* Debe intentarse evitar que las primeras observaciones de una sesión se expliquen con las últimas de la sesión anterior. Se señala la eliminación de tantas observaciones al principio de la sesión como orden tenga el VAR para conseguirlo.

Para evitar duplicidad de código se define una función marco dentro de la cual, para cada activo y representación de la negociación:

* Se hace la manipulación de los datos. Por un lado, se genera una nueva variable para el cambio del punto medio (otorgando valor faltante, NA, a la primera observación de cada día), por otro se crea también la variable de la representación de la negociación de la especificación b.
* Se extraen las variables que entrarán en el VAR según la especificación.
* Se escoge internamente el orden del VAR, según el criterio de Akaike, con la función VARselect del paquete vars. Cabe señalarse que, en el momento en que se elige el orden, no es posible todavía evitar el efecto de causalidad entre días mencionado anteriormente. Sin embargo, como solamente se trata de un paso previo a la estimación, se considera que el efecto de esa limitación no será muy importante, aun asumiendo que la instrucción arroje un orden por encima del óptimo.
* Inicialmente se consideró la implementación del requisito de independencia entre sesiones directamente y como se especifica el enunciado. No obstante, al usar la función VAR del paquete vars, se juzgó que la eliminación de las primeras observaciones de la sesión no solucionaba el problema: borrar esos registros solamente se traducía en una dependencia de los primeros datos no borrados respecto los últimos del día anterior. Seguidamente se especuló con la posibilidad de borrar los datos, pero conservar los registros como datos faltantes o, incluso, con el fin de no perder información, introducir registros con tantos datos faltantes como orden del VAR entre los últimos datos de un día y los primeros del siguiente, pero ambas aproximaciones no tenían visos de funcionar desde el momento en que la función VAR del paquete vars no admite valores faltantes en las series temporales. Por otra parte, interpolar esos valores faltantes de la serie introduciría registros artificiales en el análisis que podrían desvirtuar los resultados. Como ninguno de los métodos introducidos solucionaba el problema inicial para el modo en que estima las ecuaciones del VAR la función en cuestión del paquete vars, se decidió aproximarse al problema prescindiendo de esta, usando regresiones lineales.
* Con anterioridad a la efectuación de las regresiones se ha considerado necesario armar un bucle interno que, a partir de una función interna previa, genera tantas variables para los retardos del VAR como orden se ha preseleccionado. El método alinea en cada fila los valores actuales y pasados de ambas variables corriendo tantas filas hacia abajo las columnas correspondientes a los retardos como índice tenga el retardo. Recuérdese que la primera observación de cada sesión de la variable del cambio del punto medio es un dato faltante. Por tanto, al eliminar de las tablas todos los registros con algún dato faltante, se eliminan los primeros registros de cada día (tantos como orden tenga el VAR) pero se conserva la información eliminada en las columnas correspondientes a los retardos. Las regresiones lineales buscan en las variables de retardo como si fueran variables distintas, de forma que se consigue que cada registro conste solamente de datos del mismo día.
* Tanto la función interna como el bucle asociado incluyen funciones condicionales que permiten bautizar correctamente las variables/retardos que va generando.
* La primera regresión hace depender el cambio del punto medio de todas las demás variables retardadas, incluyendo su relación contemporánea con la otra variable (el signo o el tamaño multiplicado por el signo, dependiendo de la especificación deseada, a o b); la ecuación de la segunda se hace depender de la especificación, el signo o el tamaño multiplicado por el signo dependen de todas las demás variables excepto del cambio contemporáneo del punto medio. Los coeficientes de ambas regresiones son los coeficientes del VAR estructural.
* La función marco también contiene todos los cálculos necesarios para el cálculo de las FIR que se requieren en los apartados 3 y 4. Los cálculos se efectúan instrumentalizando dos matrices con valor inicial de cero con dimensiones de columnas análogas al número de variables explicativas de las dos regresiones del VAR y tantas filas como número de periodos hacia el futuro requeridos, más 2.
* Se asignan valores de 1 a todos los valores de la primera columna de ambas matrices, correspondientes al intercepto de las regresiones. A continuación, se inicializa la matriz 1, correspondiente a la primera ecuación del modelo, con el shock contemporáneo especificado por el parámetro init; el mismo valor se traslada como retardo en las siguientes filas (periodos hacia el futuro), tanto en la matriz 1 como en la 2. Luego se multiplica esa primera fila de la matriz 1 por los coeficientes de la primera regresión; el resultado se asigna al valor retardado del cambio del punto medio en los sucesivos periodos hacia el futuro, en las dos matrices. Seguidamente se toma la primera fila de la matriz 2 y se multiplica por los coeficientes de la segunda regresión; el valor resultante se asigna a los retardos del signo en los sucesivos periodos hacia el futuro, en ambas matrices. Luego se toma la segunda fila de la primera matriz y se repite el proceso, en bucle. El resultado son dos matrices que contienen, salvo por sendos triángulos superiores de ceros, repeticiones de los coeficientes de las FIR.
* De todo lo estimado dentro de la función marco se extraen las dos matrices y, por lo que respecta a las regresiones, por limitaciones de memoria que se han puesto de manifiesto durante la elaboración de la tarea, solamente se extraen los coeficientes y la varianza de los errores. Las matrices contienen la información necesaria para representar las FIR en los apartados 3 y 4, los coeficientes de las regresiones son imprescindibles para la obtención de la VMA en el apartado 6 y la varianza de los errores lo es para la descomposición de la volatilidad que se requiere en el mismo apartado.

manualVAR <- function(d, vars, crit, ahead, init){  
 # Función marco para la estimación manual del VAR y los coeficientes de   
 # las FIR.  
 # INPUTS:  
 # d: orden del ativo en la lista VAR  
 # vars: columnas del VAR (dqmp + sign o sign\*size); vector de posic.   
 # de longitud 2. En primera posición hay que situar sign o signxze,   
 # en segunda dqmp. Tal como se ha programado:  
 # sign es la columna 4  
 # dqmp es la columna 6  
 # signxze (sign \* size) es la columna 7  
 # crit: criterio de elección del orden del VAR (Akaike: 1)  
 # ahead: Pasos hacia el futuro  
 # init: valor de inicialización del FIR  
 # OUTPUTS:  
 # mat1 y mat2: matrices instrumentales; incluyen los valores de FIR.  
 # mco1coef y mco2coef: coeficientes de las regresiones.  
 # res1 y res2: varianzas de los residuos de las regresiones.  
   
 # Generación de cambios en el punto medio (cambio de día: NA)  
 VAR[[d]]$dqmp <- ifelse(VAR[[d]]$day == lag(VAR[[d]]$day),   
 log(VAR[[d]]$qmp) - log(lag(VAR[[d]]$qmp)), NA)  
 # Generación de signo\*tamaño  
 VAR[[d]]$signxze <- VAR[[d]]$sign \* VAR[[d]]$size  
 # Selección de las variables que entran en el VAR:  
 VAR2VAR <- subset(VAR[[d]], select = vars)  
 # Selección del orden del VAR  
 var.ord = VARselect(na.omit(VAR2VAR), lag.max = 10, type = "none")  
   
 narowsfunct <- function(num){  
 # Función de generación de subtablas de NA para añadir al  
 # principio y/o al final de las columnas de retardos en preparación  
 # de los datos para regresiones lineales  
 if (vars[1] == 4){  
 narows <- data.frame(sign = rep(NA, num), dqmp = rep(NA, num))  
 }  
 else  
 {  
 narows <- data.frame(signxze = rep(NA, num), dqmp = rep(NA, num))  
 }  
 return(narows)  
 }  
   
 # Se inicializa la nueva tabla con dos series originales con tantos NA  
 # al final como orden del VAR.  
 VAR2VARexp <- rbind(VAR2VAR, narowsfunct(as.numeric(var.ord$selection[crit])))  
 for (r in 1:as.numeric(var.ord$selection[crit])){  
 # El bucle va añadiendo columnas desplazadas tantas filas hacia abajo como  
 # índice del retardo. Para que case con las columnas adyacentes es  
 # necesario rellenar los huecos con NA.  
 lagVAR2VAR <- rbind(rbind(narowsfunct(r), VAR2VAR),   
 narowsfunct(as.numeric(var.ord$selection[crit])-r))  
 if (vars[1] == 4){ # Condicional para bautizar las columnas que  
 # se van generando  
 colnames(lagVAR2VAR) <- c(paste0("sign.l",r), paste0("dqmp.l",r))  
 }  
 else  
 {  
 colnames(lagVAR2VAR) <- c(paste0("signxze.l",r), paste0("dqmp.l",r))  
 }  
 # Van yuxtaponiéndose las columnas:  
 VAR2VARexp <- cbind(VAR2VARexp, lagVAR2VAR)  
 }  
   
 # Regresiones lineales para el VAR:  
 # La primera incluye relación contemporánea  
 mco1 <- lm(dqmp~., data = na.omit(VAR2VARexp))  
 if (vars[1] == 4){ # La segunda no  
 # Se hace depender de la especificación escogida  
 mco2 <- lm(sign~.-dqmp, data = na.omit(VAR2VARexp))  
 }  
 else  
 {  
 mco2 <- lm(signxze~.-dqmp, data = na.omit(VAR2VARexp))  
 }  
   
 # Se rescatan los nombres de los coeficientes  
 coef1 <- attr(mco1$coefficients, "names")  
 # Se crea unamatriz de ceros.  
 # Número de columnas: tantas como coeficientes  
 # Número de filas: periodos hacia el futuro +2  
 mat1 <- matrix(0, nrow = ahead+2, ncol = length(coef1))  
 colnames(mat1) <- coef1  
 mat1[,1] <- 1 # se asigna 1 a la columna del intercepto.  
   
 # Proceso análogo al anterior para mat2  
 coef2 <- attr(mco2$coefficients, "names")  
 mat2 <- matrix(0, nrow = ahead+2, ncol = length(coef2))  
 colnames(mat2) <- coef2  
 mat2[,1] <- 1  
   
 # Inicializamos mat1 y mat2:  
 mat1[1,2] <- init # Se inicializa el proceso  
 for (c in 1:(ahead+1)){  
 if (c == 1){ # Para el primer momento (relación contemporánea)  
 for (i in 1:as.numeric(var.ord$selection[crit])){  
 # En un primer momento se traslada el valor de   
 # inicialización a los retardos de sus periodos futuros.  
 mat1[i+1,i\*2+1] <- init  
 mat2[i+1,2\*i] <- init  
 }  
 for (i in 1:as.numeric(var.ord$selection[crit])){  
 # Primera implementación sobre mat1 con coef1  
 mat1[i+c,i\*2+2] <- sum(as.numeric(mco1$coefficients)\*mat1[c,])  
 mat2[i+c,i\*2+1] <- sum(as.numeric(mco1$coefficients)\*mat1[c,])  
 }  
 }  
 else  
 { # Para momentos posteriores a la relación contemporánea  
 for (i in 1:as.numeric(var.ord$selection[crit])){  
 if (i+c <= ahead+2){ # Condicional para evitar intentar situar  
 # valores fuera de las dimensiones de mat2  
 mat1[c,2] <- sum(as.numeric(mco2$coefficients)\*mat2[c,])  
 mat1[i+c, i\*2+1] <- sum(as.numeric(mco2$coefficients)\*mat2[c,])  
 mat2[i+c, 2\*i] <- sum(as.numeric(mco2$coefficients)\*mat2[c,])  
 }  
 else  
 {next} # Si el valor queda fuera de los límites, pasa  
 }  
 for (i in 1:as.numeric(var.ord$selection[crit])){  
 if (i+c <= ahead+2){ # Proceso análogo para mat1  
 mat1[i+c, i\*2+2] <- sum(as.numeric(mco1$coefficients)\*mat1[c,])  
 mat2[i+c, i\*2+1] <- sum(as.numeric(mco1$coefficients)\*mat1[c,])  
 }  
 else  
 {next}  
 }   
 }  
 }  
 # Se completa la tabla con el cálculo de la celda que falta:  
 mat1[ahead+2,2] <- sum(as.numeric(mco2$coefficients)\*mat2[ahead+2,])  
 # Captura de resultados que interesan:  
 results <- list(mco1coef = mco1$coefficients, mco2coef = mco2$coefficients,   
 mat1 = mat1, mat2 = mat2, res1 = var(mco1$residuals),  
 res2 = var(mco2$residuals))  
 return(results)  
}

Una vez definida la función se aplica en bucle con las dos especificaciones señaladas (a y b; también sacando las FIR con los tres valores de inicialización señalados en el punto 3) y los resultados se almacenan en una “matriz de listas”. Este tipo de objetos permiten referenciar los resultados como si fuera una matriz (añadiendo [[1]] a una posición[x, y]) o sencillamente como si fuera una lista. Se prefiere la referenciación como matriz porque permite localizar los modelos de acuerdo con cómo se han generado: los distintos activos se localizan en las columnas, en las filas se disponen los distintos modelos/valores de inicialización. Por supuesto, los coeficientes de los modelos correspondientes a distintos valores de inicialización son idénticos para las filas de la 2 a la 4 (todos son el modelo b pero con distintos valores de inicialización)

results <- matrix(vector(mode = "list", length = 1), nrow = 4, ncol = 5)  
init <- c(1, 100, 500, 1000)  
# Los resultados se clasifican por activo en las columnas  
# Por especificaciones del modelo en las filas  
  
for (d in 1:length(VAR)){  
 for (i in 1:length(init)){  
 if (init[i] == 1)  
 {  
 results[i,d][[1]] <- manualVAR(d, c(4, 6), 1, 20, init[i])  
 }  
 else  
 {  
 results[i,d][[1]] <- manualVAR(d, c(7, 6), 1, 20, init[i])  
 }  
 }  
 rm(d, i)  
}

# 3. FIR de un *shock* unitario de compra (modelo a)

Como se ha dicho en el punto 2, se dispone de los valores de la FIR en los resultados de aplicar la función marco con los parámetros adecuados.

Se pretende representar la correspondiente al activo de nombre 4. Se rescatan los resultados para introducirlos en vectores y se introducen en un *data frame* con una variable t que representan los sucesivos periodos hacia el futuro.

# FIR de sign0 = 1 sobre los siguientes 20 adelantos de dqmp  
FIRdqmp <- results[1,1][[1]]$mat1[2:22,4]  
# FIR de sign0 = 1 sobre los siguientes 20 adelantos de sign  
FIRsign <- results[1,1][[1]]$mat1[1:21,2]  
data <- data.frame(t = seq(0,20), FIRdqmp = FIRdqmp, FIRsign = FIRsign)

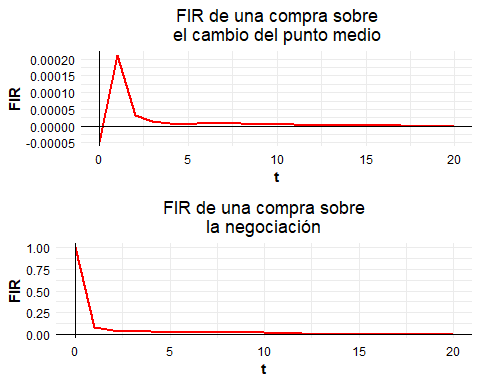
Seguidamente se define una función para la generación de gráficos FIR a partir de las prestaciones de ggplot (incluidas en tidyverse). En los gráficos que genera esta función, el eje x representa los periodos hacia delante; el eje y es uno de los parámetros de la función, junto con el origen de los datos y una cadena de caracteres con la que completar el título del gráfico.

FIRgraf <- function(dataframe, column, title){  
 ggplot(data = dataframe, aes(x = t)) +  
 geom\_line(aes(y = column), color = "red", size = 1) +  
 geom\_hline(aes(yintercept = 0)) +  
 geom\_vline(aes(xintercept = 0)) +  
 theme\_minimal() +  
 theme(axis.title.x = element\_text(color = "black", face = "bold"),  
 axis.title.y = element\_text(color = "black", face = "bold"),  
 axis.text.x = element\_text(color = "black"),  
 axis.text.y = element\_text(color = "black"),  
 plot.title = element\_text(color = "black", hjust = .5)) +  
 ggtitle(paste0("FIR de una compra sobre\n", title)) +  
 ylab("FIR")  
}

Los gráficos para cada ecuación se anidan en una estructura de gráficos de cowplot.

plot\_grid(FIRgraf(data, FIRdqmp, "el cambio del punto medio"),  
 FIRgraf(data, FIRsign, "la negociación"), nrow = 2)

**Figura 1**



rm(FIRdqmp, FIRsign, data)

Como puede observarse, el efecto contemporáneo sobre el cambio del punto medio es ligeramente negativo y tiene un pico en el primer periodo hacia adelante. Luego decrece progresivamente hasta tender a cero. En cuanto al efecto sobre el *signo*, en el primer periodo hacia adelante ya ha decrecido fuertemente y, en los siguientes, la probabilidad de compra sigue decreciendo hasta converger a cero.

# 4. FIR de un *shock* de compra de 100, 500 y 1000 acciones (modelo b)

En este punto vuelve a utilizarse la función creada en el punto 3. El proceso es idéntico al seguido entonces. Todos los gráficos generados también se refieren al activo de nombre 4.

## 4.1. FIR de un *shock* de compra de 100 acciones

# FIR de sign0 = 1 sobre los siguientes 20 adelantos de dqmp  
FIRdqmp <- results[2,1][[1]]$mat1[2:22,4]  
# FIR de sign0 = 1 sobre los siguientes 20 adelantos de sign  
FIRsignxze <- results[2,1][[1]]$mat1[1:21,2]  
data <- data.frame(t = seq(0,20), FIRdqmp = FIRdqmp, FIRsignxze = FIRsignxze)

plot\_grid(FIRgraf(data, FIRdqmp, "el cambio del punto medio"),  
 FIRgraf(data, FIRsignxze, "la negociación (vol = 100)"), nrow = 2)

**Figura 2 (Parte 1)**

Gráfico, Tabla

Descripción generada automáticamente

## 4.2. FIR de un *shock* de compra de 500 acciones

# FIR de sign0 = 1 sobre los siguientes 20 adelantos de dqmp  
FIRdqmp <- results[3,1][[1]]$mat1[2:22,4]  
# FIR de sign0 = 1 sobre los siguientes 20 adelantos de signxze  
FIRsignxze <- results[3,1][[1]]$mat1[1:21,2]  
data <- data.frame(t = seq(0,20), FIRdqmp = FIRdqmp, FIRsignxze = FIRsignxze)

plot\_grid(FIRgraf(data, FIRdqmp, "el cambio del punto medio"),  
 FIRgraf(data, FIRsignxze, "la negociación (vol = 500)"), nrow = 2)

**Figura 2 (Parte 2)**

Gráfico, Tabla

Descripción generada automáticamente

## 4.3. FIR de un *shock* de compra de 1000 acciones

# FIR de sign0 = 1 sobre los siguientes 20 adelantos de dqmp  
FIRdqmp <- results[4,1][[1]]$mat1[2:22,4]  
# FIR de sign0 = 1 sobre los siguientes 20 adelantos de signxze  
FIRsignxze <- results[4,1][[1]]$mat1[1:21,2]  
data <- data.frame(t = seq(0,20), FIRdqmp = FIRdqmp, FIRsignxze = FIRsignxze)

plot\_grid(FIRgraf(data, FIRdqmp, "el cambio del punto medio"),  
 FIRgraf(data, FIRsignxze, "la negociación (vol = 1000)"), nrow = 2)

**Figura 2 (Parte 3)**

Gráfico, Tabla

Descripción generada automáticamente

rm(FIRdqmp, FIRsignxze, data)

Todas las FIR del modelo (b) siguen un esquema muy parecido al de (a). La principal conclusión que se extrae es que la magnitud del *shock* depende fuertemente del tamaño de la compra de acciones. A mayor tamaño, la magnitud del *shock* es mayor.

# 5. Impacto Permanente en Precios medio estimado

El impacto permanente en precios viene a ser la incidencia a largo plazo de un *shock* sobre los precios. De acuerdo con las FIR calculadas, esto se traduce, para cada modelo y valor del *shock*, en el efecto acumulado del cambio del punto medio en el último periodo hacia adelante que se ha calculado.

Su cálculo se implementa mediante un bucle a través del cual se suman todos los cambios en el punto medio de las mismas matrices que han servido para la representación de la FIR.

tabla\_ii <- data.frame(aux = rep(NA, nrow(results)))  
for (d in 1:length(VAR)){  
 subtabla\_ii <- data.frame()  
 for (i in 1:nrow(results)){  
 cell <- sum(results[i,d][[1]]$mat1[2:nrow(results[i,d][[1]]$mat1), 4])  
 cell <- data.frame(cell)  
 subtabla\_ii <- rbind(subtabla\_ii, cell)  
 }  
 tabla\_ii <- cbind(tabla\_ii, subtabla\_ii)  
 rm(i, d, cell, subtabla\_ii)  
}  
tabla\_ii <- tabla\_ii[,2:6]  
colnames(tabla\_ii) <- paste0("VAR", idx)  
rownames(tabla\_ii) <- c("(a)\_+1", paste0("(b)\_+", init[2:4]))

# Para exportar tablas a formato Word  
ft <- flextable(data = format(tabla\_ii, scientific = T) %>% add\_rownames()) %>%   
 theme\_zebra %>% autofit  
ft <- set\_caption(ft, caption = "Tabla II", style = "Table Caption")  
# Crea un archivo temp  
tmp <- tempfile(fileext = ".docx")  
# Crea un documento docx  
read\_docx() %>% body\_add\_flextable(ft) %>% print(target = tmp)  
browseURL(tmp) # abre el documento

**Tabla II**

|  | **VAR4** | **VAR5** | **VAR7** | **VAR8** | **VAR9** |
| --- | --- | --- | --- | --- | --- |
| (a)\_+1 | 2.731354e-04 | 4.059690e-04 | 1.407687e-04 | 1.275756e-04 | 8.765560e-05 |
| (b)\_+100 | 3.747521e-05 | 3.869156e-05 | 2.441604e-05 | 6.416270e-06 | 9.943260e-07 |
| (b)\_+500 | 1.939586e-04 | 1.895518e-04 | 1.246691e-04 | 3.169779e-05 | 6.131475e-06 |
| (b)\_+1000 | 3.895628e-04 | 3.781271e-04 | 2.499854e-04 | 6.329968e-05 | 1.255291e-05 |

En la tabla II queda patente que, para la especificación b, la magnitud del impacto medio permanente en precios depende de la magnitud del *shock*.

# 6. Obtención de la forma VMA y aplicación de Hasbrouck (1991b)

En este apartado se piden los promedios de la descomposición del precio eficiente de los modelos (a) y (b) para todos los activos. Para ello es imprescindible obtener las formas VMA a partir de los VAR estimados en el punto 2. Se ha optado por derivar formalmente la forma de medias móviles; un paso previo para ello consiste en rescatar los coeficientes de los modelos estimados en el punto 2 para reconstruirlos en forma matricial.

## 6.1. Construcción de la forma matricial del VAR

Todos los coeficientes necesarios para construir la forma matricial del VAR están presentes en las partes mco1coef y mco2coef de los componentes de la lista results. En cada caso, la matriz es la de efectos contemporáneos y se representa, básicamente, como una matriz con 1’s en la diagonal y el coeficiente del efecto contemporáneo de *sign* sobre el cambio del punto medio en la posición [1,2] (y 0 en el resto).

Las matrices siguen la estructura siguiente:

* En la posición [1,1] se sitúa el coeficiente correspondiente al retardo *j* del cambio del punto medio sobre la misma variable en el momento actual.
* En la posición [2,2] se sitúa el coeficiente correspondiente al retardo *j* del signo (a) o del signo multiplicado por el volumen (b) sobre la misma variable en el momento actual.
* En la posición [1,2] se sitúa el coeficiente correspondiente al retardo *j* del signo (a) o del signo multiplicado por el volumen (b) sobre el cambio del punto medio en el momento actual.
* En la posición [2,1] se sitúa el coeficiente correspondiente al retardo *j* del cambio del punto medio sobre el signo (a) o el signo multiplicado por el volumen (b) en el momento actual.

Los resultados del bucle se depositan en una estructura de listas como la siguiente:

* Archivo *VARMod*
  + VARMod[[1]]: Modelos para el activo en posición 1 (“4”)
    - VARMod[[1]][[1]]: Matrices del modelo (a) para el activo en posición 1.
      * VARMod[[1]][[1]][[1]]: Matriz del modelo (a) para el activo en posición 1.
      * VARMod[[1]][[1]][[2]]: Matriz del modelo (a) para el activo en posición 1.
      * …
    - VARMod[[1]][[2]]: Matrices del modelo (b) para el activo en posición 1
    - …
  + VARMod[[2]]: Modelos para el activo en posición 2 (“5”)
  + …

# Se meterán los grupos de matrices en una triple lista anidada:  
VARMod <- vector(mode = "list", length = length(VAR))  
for (d in 1:length(VARMod)){ # Para cada activo  
 # Se crea una lista donde se almacenarán 2 listas de matrices, una  
 # correspondiente al VAR de las ecuaciones dqmp y sign, y la otra la  
 # de dqmp y signxze (sign\*size):  
 Mod <- vector(mode = "list", length = 2)  
 for (m in 1:2){  
 # Se crea una lista con tantos elementos como retardos del VAR:  
 Aj <- vector(mode = "list", length = (ncol(results[m,d][[1]]$mat1)/2))  
 # Genera la matriz A0 y la mete en la PRIMERA posición de la lista  
 Aj[[1]] <- matrix(c(1, as.numeric(results[m,d][[1]]$mco1coef[2]), 0, 1),   
 nrow = 2, byrow = T)  
 for (i in 1:(length(Aj)-1)){  
 # Generan el resto de matrices y las mete en la lista con índice = retardo:  
 Aj[[i+1]] = matrix(c(as.numeric(results[m,d][[1]]$mco1coef[2\*i+2]),  
 as.numeric(results[m,d][[1]]$mco1coef[2\*i+1]),  
 as.numeric(results[m,d][[1]]$mco2coef[2\*i+1]),  
 as.numeric(results[m,d][[1]]$mco2coef[2\*i])),   
 nrow = 2, byrow = T)  
 }  
 Mod[[m]] <- Aj  
 }  
 VARMod[[d]] <- Mod  
 rm(d, m, i, Aj, Mod)  
}

## 6.2. Cálculo de los coeficientes del VMA

Los coeficientes del VMA se han derivado de acuerdo con las notas de cálculo matricial proporcionadas. El algoritmo implementado emplea tres objetos depositarios de matrices:

* PSIc es una matriz de matrices en formato lista donde cada elemento representa un componente de la suma de matrices a partir de las cuales se calculan las matrices de coeficientes VMA (matrices PSI) para cada retardo. El algoritmo explota la propiedad de que cada uno de los *j* componentes a partir de los cuales se calcula PSI para el retardo *j* está compuesto por la multiplicación de una matriz Astar y una PSI cuyos índices suman siempre *j*.
* PSI es un objeto tipo lista donde va metiéndose la suma por filas de los diferentes componentes de PSIc.
* Astar es una lista en la que van introduciéndose los productos matriciales de las matrices del VAR (determinados en el punto 6.1.) por la matriz PSI en el retardo anterior.

Para cada activo y modelo el algoritmo se inicializa definiendo los coeficientes VMA contemporáneos (PSI[[1]] = PSIc[1,1][[1]]) como la inversa de la matriz del VAR. A partir de esta matriz se arma un bucle que calcula y acumula las Astar, que seguidamente sirven para calcular las matrices del PSIc. Luego se suman estas matrices y se acumulan en la lista PSI. Este nuevo PSI sirve para calcular una nueva matriz Astar en el siguiente retardo y el proceso se reinicia haciendo uso, progresivamente, de cada vez más valores pasados de PSI y Astar, explotándolo con las condiciones que se han mencionado.

La estructura resultante es un archivo con la siguiente estructura

* Archivo *VMA*
  + VMA[[1]]: Modelos para el activo en posición 1 (“4”)
    - VMA[[1]][[1]]: Matrices del modelo (a) para el activo en posición 1.
      * VMA[[1]][[1]]$PSI[[1]]: Matriz del modelo (a) para el activo en posición 1.
      * VMA[[1]][[1]]$PSI[[2]]: Matriz del modelo (a) para el activo en posición 1.
      * …
    - VMA[[1]][[2]]: Matrices del modelo (b) para el activo en posición 1
    - …
  + VMA[[2]]: Modelos para el activo en posición 2 (“5”)
  + …

VMA <- vector(mode = "list", length = length(VARMod))  
for (d in 1:length(VARMod)){  
 Mod <- vector(mode = "list", length = 2)  
 for (m in 1:2){  
 # Se define una "matriz de matrices" en formato lista, instrumental.  
 PSIc <- matrix(vector(mode = "list", length = 1),   
 nrow = length(VARMod[[d]][[m]]), ncol = length(VARMod[[d]][[m]]))  
 # Se define una lista para depositar los valores de A\*:  
 Astar <- vector(mode = "list", length = length(VARMod[[d]][[m]]))  
 # Se define una lista para depositar los valores de PSI:  
 PSI <- vector(mode = "list", length = length(VARMod[[d]][[m]]))  
   
 PSIc[1,1][[1]] <- PSI[[1]] <- solve(VARMod[[d]][[m]][[1]]) # PSI0  
 for (r in 2:length(Astar)){  
 Astar[[r]] <- VARMod[[d]][[m]][[r]] %\*% PSI[[r-1]]  
 for (i in 1:(r-1)){  
 PSIc[r,i][[1]] <- PSI[[i]] %\*% Astar[[r-i+1]]  
 }  
 # Generador de PSI cuando tiene 1 componente o más:  
 PSIdep <- matrix(rep(0, 4), ncol = 2)  
 for (i in 1:(r-1)){  
 PSIdep <- PSIdep + PSIc[r,i][[1]]  
 }  
 PSI[[r]] <- PSIdep  
 }  
 Mod[[m]] <- list(PSI = PSI)  
 }  
 VMA[[d]] <- Mod  
 rm(d, Mod, m, PSIc, Astar, PSI, r, PSIdep, i)  
}

## 6.3. Descomposición de la volatilidad a largo plazo

Las formas VAR y VMA son teóricamente equivalentes, con lo cual se presume que sus ecuaciones tienen el mismo ajuste y, por tanto, las mismas varianzas de los errores. Los valores de estas varianzas ya se capturaron en el punto 2, con lo cual solamente cabe rescatarlos para, junto con los coeficientes del VMA, aplicar la descomposición a la volatilidad a largo plazo en sus componentes debido a la negociación y a la información pública.

Los coeficientes del VMA relevantes para la descomposición son los que se encuentran en las posiciones:

* [1,1] para el componente debido a información pública.
* [1,2] para el componente debido a la negociación.

Las partes de la volatilidad correspondientes a cada componente se calculan como porcentajes.

tabla\_iii <- data.frame()  
for (d in 1:length(VMA)){  
 subtabla\_iii <- data.frame(aux = NA)  
 for (m in 1:2){  
 pinfo <- c()  
 neg <- c()  
 for (i in 1:length(VMA[[d]][[m]]$PSI)){  
 pinfo[i] <- (VMA[[d]][[m]]$PSI[[i]][1,1])^2  
 neg[i] <- (VMA[[d]][[m]]$PSI[[i]][1,2])^2  
 }  
 pinfo <- sum(pinfo)\*results[m,d][[1]]$res1  
 neg <- sum(neg)\*results[m,d][[1]]$res2  
 row <- data.frame(pinfo\_pc = (pinfo/(pinfo+neg)), neg\_pc = (neg/(pinfo+neg)))  
 subtabla\_iii <- cbind(subtabla\_iii, row)  
 }  
 tabla\_iii <- rbind(tabla\_iii, subtabla\_iii)  
 rm(d, subtabla\_iii, pinfo, neg, row, m, i)  
}  
tabla\_iii <- tabla\_iii[,2:5]  
colnames(tabla\_iii) <- c("P.info\_(a)", "Neg\_(a)", "P.info\_(b)", "Neg\_(b)")  
rownames(tabla\_iii) <- paste0("VAR",idx)

var <- colnames(tabla\_iii) # Se disponen los datos como % con 2 dec.  
tabla\_iii[, var] <- lapply(tabla\_iii[, var], percent\_format(accuracy = .01))  
rm(var)

# Para exportar tablas a formato Word  
ft <- flextable(data = tabla\_iii %>% add\_rownames()) %>%   
 theme\_zebra %>% autofit  
ft <- set\_caption(ft, caption = "Tabla III", style = "Table Caption")  
# Crea un archivo temp  
tmp <- tempfile(fileext = ".docx")  
# Crea un documento docx  
read\_docx() %>% body\_add\_flextable(ft) %>% print(target = tmp)  
browseURL(tmp) # abre el documento

**Tabla III**

|  | **Modelo (a)** | | **Modelo (b)** | |
| --- | --- | --- | --- | --- |
|  | **P.info** | **Neg** | **P.info** | **Neg** |
| VAR4 | 91.11% | 8.89% | 94.10% | 5.90% |
| VAR5 | 92.94% | 7.06% | 96.17% | 3.83% |
| VAR7 | 88.99% | 11.01% | 94.44% | 5.56% |
| VAR8 | 90.95% | 9.05% | 94.75% | 5.25% |
| VAR9 | 95.84% | 4.16% | 96.27% | 3.73% |

De los resultados se desprende que la información privada es responsable de una parte de la volatilidad que oscila entre el 3.73% y el 11.01%, dependiendo del activo y el modelo contemplado. Complementariamente, la información pública explica cerca del 89%, en el peor de los casos. Con todo, puede decirse que los agentes parecen estar operando en un entorno en el que los agentes informados son poco determinantes en la formación del precio de los activos.