

Diplomado de Econometría Financiera

Benjamin Oliva y Emiliano Pérez Caullieres

2022-09-06

Contents

1	Mínimos Cuadrados Ordinarios	5
1.1	El problema	5
1.2	Estimación R	6
1.3	Ejercicio	9
2	Máxima Verosimilitud	11
2.1	El problema	11
2.2	Estimación y simulación	12
3	Método Generalizado de Momentos (MGM)	15
3.1	El problema	15

Chapter 1

Mínimos Cuadrados Ordinarios

1.1 El problema

Recordando que el método de MCO resulta en encontrar la combinación de valores de los estimadores de los parámetros $\hat{\beta}$ que permita minimizar la suma de los residuales (estimadores de los términos de error ε) al cuadrado dada por:

$$\sum_{i=1}^N e_i^2 = \sum_{i=1}^N (y_i - \mathbf{X}_i' \hat{\beta})^2$$

Donde $\hat{\beta}$ denota el vector de estimadores $\hat{\beta}_1, \dots, \hat{\beta}_K$ y dado que $(e_1, e_2, \dots, e_n)'(e_1, e_2, \dots, e_n) = \mathbf{e}'\mathbf{e}$, el problema del método de MCO consiste en resolver el problema de optimización:

$$\begin{aligned} \text{Minimizar}_{\hat{\beta}} S(\hat{\beta}) &= \text{Minimizar}_{\hat{\beta}} \mathbf{e}'\mathbf{e} \\ &= \text{Minimizar}_{\hat{\beta}} (\mathbf{Y} - \mathbf{X}\hat{\beta})'(\mathbf{Y} - \mathbf{X}\hat{\beta}) \end{aligned}$$

Expandiendo la expresión $\mathbf{e}'\mathbf{e}$ obtenemos:

$$\mathbf{e}'\mathbf{e} = \mathbf{Y}'\mathbf{Y} - 2\mathbf{Y}'\mathbf{X}\hat{\beta} + \hat{\beta}'\mathbf{X}'\mathbf{X}\hat{\beta}$$

De esta forma obtenemos que las condiciones necesarias de un mínimo son:

$$\frac{\partial S(\hat{\beta})}{\partial \hat{\beta}} = -2\mathbf{X}'\mathbf{Y} + 2\mathbf{X}'\mathbf{X}\hat{\beta} = \mathbf{0}$$

Y se pueden despejar las *ecuaciones normales* dadas por:

Debido a que el objetivo es encontrar la matriz $\hat{\beta}$ despejamos:

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$$

$$\mathbf{X}'\mathbf{X}\hat{\beta} = \mathbf{X}'\mathbf{Y}$$

1.2 Estimación R

Para la estimación utilizaremos el paquete “BatchGetSymbols”. Este paquete nos permitirá descargar información acerca de la bolsa de valores internacional.

1.2.1 Dependencias

```
#install.packages("pacman")
#pacman nos permite cargar varias librerías en una sola línea
library(pacman)
pacman::p_load(tidyverse,BatchGetSymbols,ggplot2, lubridate)
```

1.2.2 Descarga de los valores

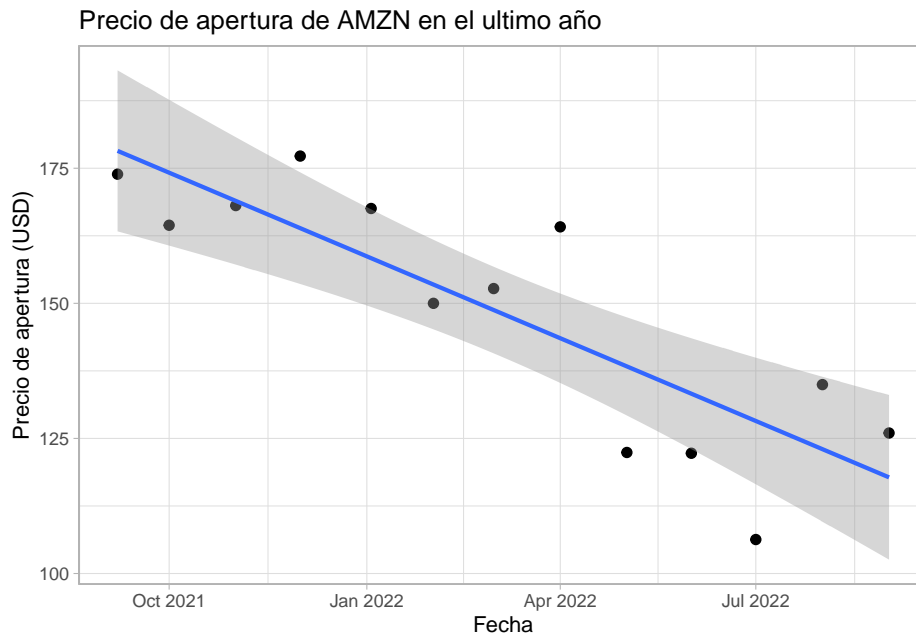
```
#Primero determinamos el lapso de tiempo
pd<-Sys.Date()-365 #primer fecha
pd
#> [1] "2021-09-06"
ld<-Sys.Date() #última fecha
ld
#> [1] "2022-09-06"
#Intervalos de tiempo
int<-"monthly"
#Datos a elegir
dt<-c("AMZN")

#Descargando los valores
?BatchGetSymbols()
data<- BatchGetSymbols(tickers = dt,
                        first.date = pd,
                        last.date = ld,
                        freq.data = int,
                        do.cache = FALSE,
                        thresh.bad.data = 0)

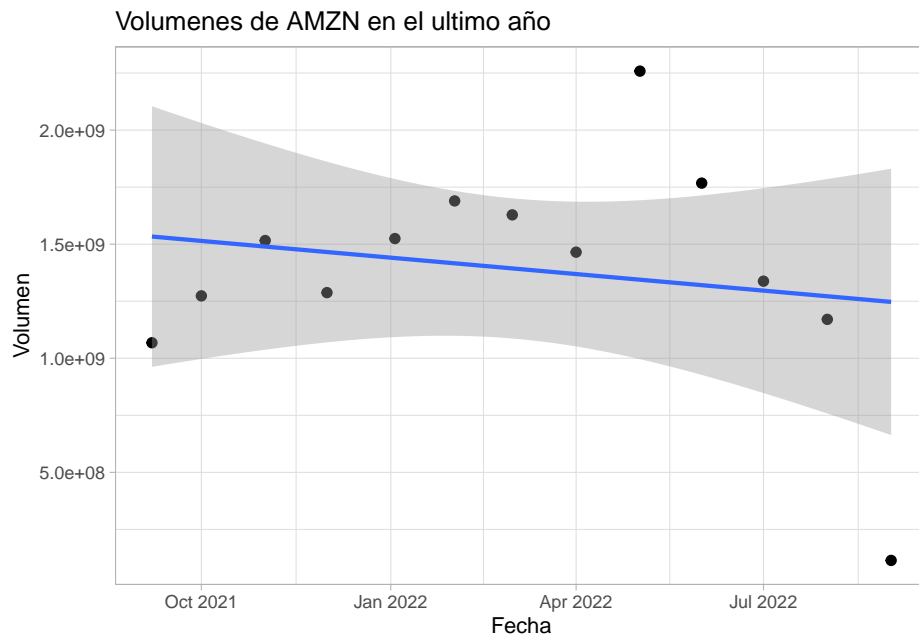
#Generando data frame con los valores
```

```
data_precio<-data$df.tickers
colnames(data_precio)
#> [1] "ticker"           "ref.date"
#> [3] "volume"           "price.open"
#> [5] "price.high"       "price.low"
#> [7] "price.close"      "price.adjusted"
#> [9] "ret.adjusted.prices" "ret.closing.prices"
```

```
sp_precio<-ggplot(data_precio, aes(x=ref.date, y=price.open))+geom_point(size =2, colour = "black")
sp_precio
```



```
sp_volumen<-ggplot(data_precio, aes(x=ref.date, y=volume))+geom_point(size =2, colour = "black")+
sp_volumen
```



1.2.4 Regresión lineal que optiene los coeficientes $\hat{\beta}$

```
#datos estadísticos
summary(data_precio[c("price.open","volume")])
#>   price.open      volume
#>   Min.   :106.3   Min.   :1.140e+08
#>   1st Qu.:126.0   1st Qu.:1.273e+09
#>   Median :152.7   Median :1.465e+09
#>   Mean   :148.5   Mean   :1.392e+09
#>   3rd Qu.:167.6   3rd Qu.:1.628e+09
#>   Max.   :177.2   Max.   :2.258e+09
#análisis de regresión lineal lm() y=precio,x=fecha
reg_tiempo_precio<-lm(price.open~ref.date, data=data_precio)
#¡Siempre se pone dentro de lm() la variable dependiente primero y luego la independiente
summary(reg_tiempo_precio)
#>
#> Call:
#> lm(formula = price.open ~ ref.date, data = data_precio)
#>
#> Residuals:
#>      Min       1Q   Median       3Q      Max
#> -21.9379  -9.7257  -0.8686   9.1948  20.6055
#>
#> Coefficients:
```



```
#>               Estimate Std. Error t value Pr(>|t|)
#> (Intercept) 3355.37814   615.12157   5.455 0.000199 ***
#> ref.date    -0.16831     0.03228  -5.214 0.000288 ***
#> ---
#> Signif. codes:
#> 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Residual standard error: 13.13 on 11 degrees of freedom
#> Multiple R-squared:  0.7119, Adjusted R-squared:  0.6857
#> F-statistic: 27.18 on 1 and 11 DF,  p-value: 0.0002884

#análisis de regresión lineal lm() y=volumen,x=fecha
reg_tiempo_volumen<-lm(volume~ref.date, data=data_precio)
summary(reg_tiempo_volumen)
#>
#> Call:
#> lm(formula = volume ~ ref.date, data = data_precio)
#>
#> Residuals:
#>      Min       1Q   Median       3Q      Max
#> -1.133e+09 -1.780e+08  4.137e+07  2.347e+08  9.141e+08
#>
#> Coefficients:
#>               Estimate Std. Error t value Pr(>|t|)
#> (Intercept)  1.659e+10  2.358e+10   0.704   0.496
#> ref.date    -7.978e+05  1.237e+06  -0.645   0.532
#>
#> Residual standard error: 503100000 on 11 degrees of freedom
#> Multiple R-squared:  0.03641, Adjusted R-squared: -0.05119
#> F-statistic: 0.4157 on 1 and 11 DF,  p-value: 0.5323
```

1.3 Ejercicio

El objetivo de este ejercicio es simplemente que indiquen y modifiquen los errores en el código. Así pues, deberán descomentar -quitar las #antes del código- para empezar el ejercicio.

1.3.1 1

El objetivo de este código es explicar la variable “**volume**” con la variable “**price.high**”.

```
#reg_tiempo_ej1<-lm(price.high~volume, data=data_precio)
#summary(reg_tiempo_ej1)
```

1.3.2 2

El objetivo de este código es explicar la variable “**volume**” con la variable “**price.low**”.

```
#reg_tiempo_ej2<-lm(price.low~volume, data=data_precio)
#summary(reg_tiempo_ej1)
```

1.3.3 3 (opcional)

El objetivo de este ejercicio es descargar los valores del stock de Tesla *BMV: TSLA* en los últimos *dos años*.

```
#dt_ej3<-("TSLA")
#pdej<-Sys.Date()-(365*3) #primer fecha
#pdej
#Descargando los valores
#dataej3<- BatchgetSymbols(tickers = dt_ej3,
                           #first.date = pdej,
                           #last.date = ld,
                           #freq.data = int,
                           #do.cache = FALSE,
                           #thresh.bad.data = 0)

#Generando data frame con los valores
#data_precio_ej2<-dataej3$df.tickers
#1colnames(data_precio_ej2)
```

Chapter 2

Máxima Verosimilitud

2.1 El problema

Recordemos que dado $f(y_i|\mathbf{x}_i)$ la función de densidad condicional de y_i dado \mathbf{x}_i . Sea θ un conjunto de parámetros de la función. Entonces la función de densidad conjunta de variables aleatorias independientes $\{y_i : y_i \in \mathbb{R}\}$ dados los valores $\{\mathbf{x}_i : \mathbf{x}_i \in \mathbb{R}^K\}$ estará dada por:

$$\Pi_{i=1}^n f(y_i|\mathbf{x}_i; \theta) = f(y_1, y_2, \dots, y_n | \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n; \theta) = L(\theta) \quad (2.1)$$

A la ecuación (2.1) se le conoce como ecuación de verosimilitud. El problema de máxima verosimilitud entonces será:

$$\max_{\theta \in \Theta} \Pi_{i=1}^n f(y_i|\mathbf{x}_i; \theta) = \max_{\theta \in \Theta} L(\theta) \quad (2.2)$$

Dado que el logaritmo natural es una transformación monótona, podemos decir que el problema de la ecuación (2.2) es equivalente a:

$$\max_{\theta \in \Theta} \ln L(\theta) = \max_{\theta \in \Theta} \ln \Pi_{i=1}^n f(y_i|\mathbf{x}_i; \theta) = \max_{\theta \in \Theta} \sum_{i=1}^n \ln f(y_i|\mathbf{x}_i; \theta) \quad (2.3)$$

Para solucionar el problema se tiene que determinar las condiciones de primer y segundo orden, las cuales serán:

$$\frac{\partial}{\partial \theta} \ln L(\theta) = \nabla \ln L(\theta) \quad (2.4)$$

$$\frac{\partial^2}{\partial \theta^2} \ln L(\theta) = \frac{\partial}{\partial \theta} \ln L(\theta) \cdot \frac{\partial}{\partial \theta} \ln L(\theta') = H(\theta) \quad (2.5)$$

La solución estará dada por aquel valor de $\hat{\theta}$ que hace:

$$\frac{\partial}{\partial \theta} \ln L(\hat{\theta}) = 0$$

A su vez, la varianza será aquella que resulta de:

$$\text{Var}[\hat{\theta}|\mathbf{X}] = (-\mathbb{E}_{\hat{\theta}}[H(\theta)])^{-1}$$

2.2 Estimación y simulación

2.2.1 Lanzar una moneda

```
set.seed(1234)#esto sirve para siempre generar los mismos numeros aleatorios
#rbinom(numero observaciones,numero de ensayos,probabilidad de exito en cada ensayo)
cara<-rbinom(1,100,0.5)
cara#esto nos dice de los 100 ensayos cuantos fueron cara
#> [1] 47
sol<-100-cara
sol
#> [1] 53

#Ahora definiremos la función que encontrará la función de verosimilitud para determinar
#
verosimilitud <- function(p){
  dbinom(cara, 100, p)
}

#si suponemos que la probabilidad sesgada de que caiga cara es 40%
prob_sesgada<-0.4
#es posible calcular la función de que salga cara
verosimilitud(prob_sesgada)
#> [1] 0.02919091
#ahora es posible generar una función de verosimilitud negativa
#para maximizar el valor de la verosimilitud
neg_verosimilitud <- function(p){
  dbinom(cara, 100, p)*-1
}
neg_verosimilitud(prob_sesgada)
#> [1] -0.02919091
# usamos la función nlm() para maximizar esta función no lineal
#?nlm()
nlm(neg_verosimilitud,0.5,stepmax=0.5)#se pone un parametro porque sabemos que hay un
#> $minimum
```

```
#> [1] -0.07973193
#>
#> $estimate
#> [1] 0.47
#>
#> $gradient
#> [1] 1.589701e-10
#>
#> $code
#> [1] 1
#>
#> $iterations
#> [1] 4
```

Si bien el ejercicio anterior es un tanto repetitivo debido a que sabemos que hay un 50% de que caiga una moneda de un lado o otro. Esto ejemplifica la manera en la que se utiliza el metodo de maximización de máxima verosimilitud.

Chapter 3

Método Generalizado de Momentos (MGM)

3.1 El problema

Retomemos el modelo de regresión lineal tal que:

$$y_i = X_i\beta + u_i \quad (3.1)$$

Tomando en cuenta los principios de ortogonalidad ($E(Z_i u_i) = 0$) y ($rank E(Z_i' X_i) = 0$) sabemos que β es el único vector de $N \times 1$ que resuelve las condiciones de momento de determinada población. En otras palabras, $E[z_i'(y_i - x_i\beta)] = 0$ es una solución y $E[z_i'(y_i - x_i\beta)] \neq 0$ NO es una solución. Debido a que la media muestral son estimadores consistentes de momentos de una población, se puede:

$$N^{-1} \sum_{i=1}^N z_i'(y_i - x_i\beta) = 0 \quad (3.2)$$

Asumiendo que la ecuación (3.2) tiene L ecuaciones lineales y K coeficientes β desconocidos y $K = L$, entonces la matriz $\sum_{i=1}^N z_i' x_i$ debe ser no singular para encontrar los coeficientes de la siguiente manera.

$$\hat{\beta} = N^{-1} \left[\sum_{i=1}^N z_i' x_i \right]^{-1} \left[\sum_{i=1}^N z_i' y_i \right] \quad (3.3)$$

Para simplificar (3.3) se puede nombrar Z juntando z_i N veces para crear una matriz de tamaño $NG \times L$. Lo mismo hacemos con X juntando x_i para obtener una de $NG \times K$ y Y obteniendo una $NG \times 1$. Obteniendo:

$$\hat{\beta} = [Z' X]^{-1} [Z' Y] \quad (3.4)$$

Es importante tomar en cuenta cuando el caso en el que hay más ecuaciones lineales que coeficientes β ; es decir, $L \geq K$. En estos casos es muy probable que no haya solución, por lo que mejor que se puede estimar es poner la ecuación (3.2), tan pequeña como sea posible. Por lo mismo el paso que nos lleva a la ecuación (3.3), debe eliminarse N^{-1} . El objetivo:

$$\min_{\beta} \left[\sum_{i=1}^N z'_i x_i \beta \right]^{-1} \left[\sum_{i=1}^N z'_i y_i \beta \right] \quad (3.5)$$

Así pues nombramos a W como una matriz simétrica de $W \times W$ donde se genera la variable b que debemos minimizar que sustituye a β creando una función cuadrática en la ecuación (3.3).

$$\min_b \left[\sum_{i=1}^N z'_i x_i b \right]^{-1} \left[\sum_{i=1}^N z'_i y_i b \right] \quad (3.6)$$

$$\therefore \hat{\beta} = [X' Z \hat{W} Z' X]^{-1} [X' Z \hat{W} Z' Y] \quad (3.7)$$

Sin embargo, $X' Z \hat{W} Z' X$ debe ser no singular para que haya una solución. Para esto se asume que \hat{W} tiene un límite de probabilidad no singular. Esto se describe como $\hat{W} \xrightarrow{p} W$ y $N \rightarrow W\infty$ donde W no es aleatorio, es una matriz positiva definida simétrica de $L \times L$.