

Sesión 11 (A) — MCO con Gretl: el ejemplo de la lección 7 y verificación con datos simulados

Marcos Bujosa

Descripción de la práctica

En las lecciones 7 y 8 obtuvimos, con lápiz y papel, las fórmulas de $\hat{\beta}_1$ y $\hat{\beta}_2$ (a partir de dos condiciones de ortogonalidad) y la descomposición $STC = SEC + SRC$ con su cociente R^2 . Hoy usamos por primera vez el comando `ols` de Gretl para comprobar que reproduce exactamente esos cálculos: primero con el mismo ejemplo pequeño ($n = 5$) ya resuelto a mano en la lección 7, y después con una muestra simulada más grande, para distinguir qué es una *identidad algebraica* (se cumple siempre) de qué es una *estimación* (no coincide, pero se aproxima al valor verdadero que generó los datos).

Nota importante: al pedir `ols` a Gretl, la salida incluye columnas adicionales (error estándar, valor t , p -valor) que *todavía no sabemos interpretar*: son objeto de estudio más adelante en el curso, cuando lleguemos a inferencia. Por ahora, ignórelas deliberadamente y fíjese solo en los coeficientes estimados y en las cantidades ya conocidas ($\hat{e}_i$, R^2).

Objetivo

1. Usar `ols` por primera vez y comprobar que reproduce exactamente el cálculo manual de la lección 7.
2. Verificar las dos condiciones de ortogonalidad ($\sum \hat{e}_i = 0$, $\sum x_i \hat{e}_i = 0$) usando los residuos que guarda Gretl.
3. Verificar $STC = SEC + SRC$ y R^2 (lección 8), comparando el cálculo manual con el que reporta Gretl.
4. Verificar $\hat{\beta}_2 = \rho_{xy} \sigma_y / \sigma_x$ y $R^2 = \rho_{xy}^2$.
5. Distinguir, con una muestra simulada más grande, qué se cumple *siempre* (ortogonalidad) de qué es solo una *aproximación* (recuperar el coeficiente generador).
6. Ver qué se pierde si se ajusta la recta *sin constante*: la primera condición de ortogonalidad desaparece, y con ella todo lo que la lección 7 dedujo de ella.

Actividad 1 - El ejemplo de la lección 7, ahora con Gretl

Recuperamos exactamente el ejemplo de la lección 7: $\mathbf{x} = (1, 2, 3, 4, 5)$, $\mathbf{u} = (1, -2, 0, 2, -1)$, $\mathbf{y} = 2 + 3\mathbf{x} + \mathbf{u}$.
en línea de comandos:

```
nulldata 5 --preserve
setobs 1 1 --cross-section

matrix mx = {1;2;3;4;5}
matrix mu = {1;-2;0;2;-1}
series x = mx
series u = mu
series y = 2 + 3*x + u
```

Licencia: Creative Commons Attribution-ShareAlike 4.0 International (CC BY-SA 4.0).

Ahora pedimos la regresión:

en línea de comandos:

```
ols y const x
```

- GUI equivalente: **Modelo** -> **Mínimos cuadrados ordinarios**, seleccionando **y** como variable dependiente y **const, x** como regresores.

Antes de ejecutar: ¿qué valores espera obtener para el coeficiente de **const** y para el de **x**? (Repase, si hace falta, el cálculo manual de la lección 7.)

Modelo 1: MCO, usando las observaciones 1-5

Variable dependiente: y

	coeficiente	Desv. típica	Estadístico t	valor p
const	2,00000	1,91485	1,044	0,3730
x	3,00000	0,577350	5,196	0,0138 **

Media de la vble. dep.	11,00000	D.T. de la vble. dep.	5,000000
Suma de cuad. residuos	10,000000	D.T. de la regresión	1,825742
R-cuadrado	0,900000	R-cuadrado corregido	0,866667
F(1, 3)	27,00000	Valor p (de F)	0,013847
Log-verosimilitud	-8,827561	Criterio de Akaike	21,65512
Criterio de Schwarz	20,87400	Crit. de Hannan-Quinn	19,55866

Lo que debe observar

Los coeficientes son exactamente 2 y 3, los mismos que calculamos a mano en la lección 7: Gretl no hace otra cosa que resolver las dos condiciones de ortogonalidad. Fíjese en **R-cuadrado**, 0,9, y en **Suma de cuad. residuos**, 10: son el R^2 y la SRC de la lección 8 con estos mismos datos. El resto de la tabla (columnas **Desv. típica**, **Estadístico t** y **valor p**, y las líneas del pie) lo dejamos para más adelante.

Actividad 2 - Residuos y las dos condiciones de ortogonalidad

Gretl guarda, tras cada regresión, el vector de residuos en el accesor **\$uhat** y el vector ajustado en **\$yhat**.

en línea de comandos:

```
series ehat = $uhat
series yhat = $yhat

print x u ehat --byobs

printf "\nSuma de residuos          sum(ehat)  = %.10f\n", sum(ehat)
printf "Suma de x*residuos        sum(x*ehat) = %.10f\n", sum(x*ehat)
printf "Diferencia maxima |ehat-u|  = %.10f\n", max(abs(ehat-u))
```

	x	u	ehat
1	1	1	1
2	2	-2	-2
3	3	0	0
4	4	2	2
5	5	-1	-1

Suma de residuos	sum(ehat)	= 0,0000000000
Suma de x*residuos	sum(x*ehat)	= 0,0000000000
Diferencia maxima ehat-u		= 0,0000000000

Recuerde la advertencia general sobre vectores de diseño: aquí $\hat{e}$ coincide EXACTAMENTE con u porque este ejemplo se construyó a propósito para que así fuera (u ya era ortogonal a los regresores). Esto no es lo habitual; lo veremos en la Actividad 5.

Actividad 3 - Bondad de ajuste: $STC = SEC + SRC$ y R^2

en línea de comandos:

```
scalar mu_y = mean(y)
scalar STC = sum((y - mu_y)^2)
scalar SEC = sum((yhat - mu_y)^2)
scalar SRC = sum(ehat^2)

printf "STC = %.4f   SEC = %.4f   SRC = %.4f   (SEC+SRC = %.4f)\n", STC, SEC, SRC, SEC+SRC
printf "R2 (a mano)           = %.6f\n", SEC/STC
printf "R2 de Gretl ($rsq)    = %.6f\n", $rsq
```

```
STC = 100,0000   SEC = 90,0000   SRC = 10,0000   (SEC+SRC = 100,0000)
R2 (a mano)           = 0,900000
R2 de Gretl ($rsq)    = 0,900000
```

El ajuste preserva la media, y la recta pasa por el punto de las medias

Fíjese en que, al calcular SEC, hemos escrito $\text{sum}((\hat{y} - \mu_y)^2)$ usando la media de $\mathbf{y}$, cuando lo que pide la definición es la media de $\hat{\mathbf{y}}$. No es un descuido: la lección 7 demostró que ambas coinciden. Comprobémoslo, junto con las otras dos consecuencias de la primera condición de ortogonalidad:

en línea de comandos:

```
scalar mu_x = mean(x)
scalar mu_yhat = mean(yhat)

printf "mu_y = %.10f   mu_yhat = %.10f   (el ajuste preserva la media)\n", mu_y, mu_yhat
printf "beta1 + beta2*mu_x = %.10f   (la recta pasa por el punto de las medias)\n", $coeff(const) + $coeff(x)*mu_x
printf "Diferencia maxima |(yhat-mu_yhat) - beta2*(x-mu_x)| = %.10f\n", max(abs((yhat-mu_yhat) - $coeff(x)*(x-mu_x)))
```

```
mu_y = 11,0000000000   mu_yhat = 11,0000000000   (el ajuste preserva la media)
beta1 + beta2*mu_x = 11,0000000000   (la recta pasa por el punto de las medias)
Diferencia maxima |(yhat-mu_yhat) - beta2*(x-mu_x)| = 0,0000000000
```

Las tres líneas son la misma condición de ortogonalidad vista de tres maneras. La primera: $\mu_y = \mu_{\hat{y}}$ —que es lo que legitima la fórmula de SEC que acabamos de usar—. La segunda: evaluada en $x = \mu_x$, la recta ajustada devuelve exactamente μ_y . La tercera es la identidad que la lección 7 pidió guardar para la lección 8: $\hat{\mathbf{y}} - \bar{\mathbf{y}} = \hat{\beta}_2(\mathbf{x} - \bar{\mathbf{x}})$ —no dos vectores de la misma longitud, sino literalmente el mismo vector, componente a componente—.

Actividad 4 - La pendiente como correlación reescalada

Recordando la lección 7 ($\hat{\beta}_2 = \rho_{xy} \sigma_y / \sigma_x$) y la lección 8 ($R^2 = \rho_{xy}^2$ en regresión simple):

en línea de comandos:

```
scalar rho_xy = corr(x,y)
scalar beta2_formula = rho_xy * sd(y)/sd(x)

printf "rho_xy           = %.6f\n", rho_xy
printf "beta2 (formula rho*sy/sx) = %.6f\n", beta2_formula
printf "beta2 ($coeff(x) de Gretl) = %.6f\n", $coeff(x)
printf "rho_xy^2         = %.6f\n", rho_xy^2
printf "R2 ($rsq de Gretl)       = %.6f\n", $rsq
```

```
rho_xy          = 0,948683
beta2 (formula rho*sy/sx) = 3,000000
beta2 ($coeff(x) de Gretl) = 3,000000
rho_xy^2        = 0,900000
R2 ($rsq de Gretl) = 0,900000
```

Actividad 5 - ¿Y si la muestra es más grande y el residuo no está “hecho a medida”?

Generamos una muestra nueva, mayor ($n = 60$), con una perturbación genuinamente aleatoria (no diseñada para ser ortogonal a $\mathbf{x}$).

en línea de comandos:

```
nulldata 60 --preserve
set seed 24680

series x2 = uniform(0, 10)
series u2 = normal(0, 2)
series y2 = 2 + 3*x2 + u2

ols y2 const x2
series ehat2 = $uhat
```

en línea de comandos:

```
printf "Suma de residuos      sum(ehat2)      = %.10f      (debe ser 0, exactamente)\n", sum(ehat2)
printf "Suma de x2*residuos sum(x2*ehat2)    = %.10f      (debe ser 0, exactamente)\n", sum(x2*ehat2)
printf "beta2 estimado (pendiente) = %.4f      (valor generador = 3, pero NO se recupera exacto)\n", $coeff(x2)
printf "beta1 estimado (constante) = %.4f      (valor generador = 2, pero NO se recupera exacto)\n", $coeff(const)
```

```
Suma de residuos      sum(ehat2)      = 0,0000000000      (debe ser 0, exactamente)
Suma de x2*residuos sum(x2*ehat2)    = -0,0000000000      (debe ser 0, exactamente)
beta2 estimado (pendiente) = 3,0601      (valor generador = 3, pero NO se recupera exacto)
beta1 estimado (constante) = 1,8220      (valor generador = 2, pero NO se recupera exacto)
```

Compare: la ortogonalidad (primeras dos líneas) se cumple exactamente, sea cual sea la muestra –es álgebra pura, consecuencia de cómo se define MCO–. En cambio, $\hat{\beta}_1, \hat{\beta}_2$ solo se aproximan a los valores 2 y 3 que usamos para generar los datos –son una estimación, no una identidad–.

Actividad 6 - ¿Y si quitamos la constante?

La lección 7 sostuvo que sin constante no puede hablarse de regresión. Comprobemos qué se pierde, con la muestra de la Actividad 5: pedimos a Gretl la recta que pasa por el origen (solo el regresor $\mathbf{x}_2$, sin `const`).

en línea de comandos:

```
ols y2 const x2 --quiet
scalar b2_con = $coeff(x2)
scalar mu_yhat_con = mean($yhat)

ols y2 x2                                     # sin constante
series ehat_sc = $uhat
printf "\nsuma de residuos      sum(ehat)      = %.4f      (ya NO es cero)\n", sum(ehat_sc)
printf "suma de x2*residuos      sum(x2*ehat) = %.10f     (esta sigue siendo cero)\n", sum(x2*ehat_sc)
printf "media de y2 = %.4f      media del ajuste sin constante = %.4f      (con constante: %.4f)\n", mean(y2), mean($yhat),
↪ mu_yhat_con
printf "pendiente sin constante = %.4f      (con constante: %.4f; generador: 3)\n", $coeff(x2), b2_con
```

Modelo 2: MCO, usando las observaciones 1-60
Variable dependiente: y2

	coeficiente	Desv. típica	Estadístico t	valor p
x2	3,35500	0,0523912	64,04	3,13e-56 ***
Media de la vble. dep.	16,42120	D.T. de la vble. dep.	8,246837	
Suma de cuad. residuos	286,3910	D.T. de la regresión	2,203199	
R-cuadrado no centrado	0,985817	R-cuadrado centrado	0,928627	
F(1, 59)	4100,788	Valor p (de F)	3,13e-56	
Log-verosimilitud	-132,0267	Criterio de Akaike	266,0534	
Criterio de Schwarz	268,1478	Crit. de Hannan-Quinn	266,8726	

suma de residuos sum(ehat) = 24,8978 (ya NO es cero)
suma de x2*residuos sum(x2*ehat) = -0,0000000000 (esta sigue siendo cero)
media de y2 = 16,4212 media del ajuste sin constante = 16,0062 (con constante: 16,4212)
pendiente sin constante = 3,3550 (con constante: 3,0601; generador: 3)

Lo que debe observar

Sin la constante solo queda una condición de ortogonalidad, la del regresor: $\sum x_i \hat{e}_i$ sigue siendo cero, pero $\sum \hat{e}_i$ vale 24,9. Con ello se pierde todo lo que la lección 7 dedujo de la primera condición: el ajuste ya no tiene la media de los datos (16,01 frente a 16,42), la recta ya no pasa por el punto de las medias, y la pendiente sube de 3,06 a 3,36: obligada a pasar por el origen, la recta se inclina para compensar la constante que le falta. Fíjese también en que Gretl ya no imprime un **R-cuadrado**, sino dos: sin constante no existe la descomposición $STC = SEC + SRC$ de la lección 8, y el programa ofrece dos sustitutos (“no centrado” y “centrado”) que no son comparables con el R^2 de los modelos con constante.

Actividad 7 - Síntesis

Hoy hemos comprobado con Gretl, en vez de a mano, tres cosas ya demostradas en las lecciones 7 y 8: (i) las fórmulas de $\hat{\beta}_1, \hat{\beta}_2$; (ii) las dos condiciones de ortogonalidad que, por construcción de MCO, se cumplen *siempre* (no dependen de si el ejemplo está “diseñado” o no); (iii) $STC = SEC + SRC$ y $R^2 = \rho_{xy}^2$. Y hemos visto la diferencia esencial entre una *identidad algebraica* (ortogonalidad: exacta siempre) y una *estimación* (recuperar el coeficiente generador: solo aproximada, y solo exacta en el ejemplo “con truco” de la lección 7).

Preguntas de interpretación para la clase

1. En la Actividad 1, ¿por qué Gretl reproduce exactamente $\hat{\beta}_1 = 2, \hat{\beta}_2 = 3$, y no una aproximación?
2. ¿Por qué la ortogonalidad ($\sum \hat{e}_i = 0, \sum x_i \hat{e}_i = 0$) se cumple exactamente en la Actividad 5, mientras que $\hat{\beta}_2$ solo se aproxima a 3?
3. ¿Qué diferencia hay entre el vector generador $\mathbf{u}$ y el vector de residuos $\hat{\mathbf{e}}$? ¿En qué caso particular, de los vistos hoy, coinciden?
4. ¿Por qué el signo de $\hat{\beta}_2$ coincide siempre con el signo de ρ_{xy} ?
5. Si en la Actividad 5 usáramos $n = 600$ en lugar de $n = 60$ (manteniendo el mismo proceso generador), ¿qué esperaríamos que ocurriera con la cercanía entre $\hat{\beta}_2$ y 3?
6. En la Actividad 6, ¿por qué $\sum x_i \hat{e}_i$ sigue siendo cero y $\sum \hat{e}_i$ no? ¿Qué subespacio es ahora el de los regresores?

Para profundizar:

- Véanse las lecciones 7 y 8 ([lección 7](#), [lección 8](#)) para las demostraciones completas de lo que hoy verificamos con Gretl.
- Cottrell, A. y Lucchetti, R. (2023). *Gretl User's Guide*. Sección sobre el comando `ols` y los accesorios tras la estimación (`$uhat`, `$yhat`, `$coeff`, `$rsq`).

Código completo de la práctica

```
# -----
# Copyright (C) 2026 Marcos Bujosa
# Licencia: GNU General Public License v3.0 o posterior
# Este código es software libre y puede ser redistribuido y/o modificado bajo los términos de la GPL.
# Ver el archivo LICENSE del repositorio para más detalles.
# -----

# Los dos primeros comandos son necesarios para que Gretl guarde los resultados de la práctica en el
↳ directorio de trabajo
# al ejecutar lo siguiente desde un terminal (use los nombres y ruta que correspondan)
#
#   DIRECTORIO="Nombre_Directorio_trabajo" gretlcli -b ruta/nombre_fichero_de_la_practica.inp
#
# Si esto no le funciona en su sistema, comente las siguientes dos líneas y sitúese en el directorio de
↳ trabajo de gretl
# que corresponda (configure dicho directorio de trabajo desde la ventana principal de Gretl).

string directory = getenv("DIRECTORIO")
set workdir "@directory"

nulldata 5 --preserve
setobs 1 1 --cross-section

matrix mx = {1;2;3;4;5}
matrix mu = {1;-2;0;2;-1}
series x = mx
series u = mu
series y = 2 + 3*x + u

outfile --quiet olsEjemploN5.txt
    ols y const x
end outfile

outfile --quiet residuosYortogonalidad.txt
    series ehat = $uhat
    series yhat = $yhat

    print x u ehat --byobs

    printf "\nSuma de residuos          sum(ehat)   = %.10f\n", sum(ehat)
    printf "Suma de x*residuos          sum(x*ehat) = %.10f\n", sum(x*ehat)
    printf "Diferencia maxima |ehat-u|      = %.10f\n", max(abs(ehat-u))
end outfile

outfile --quiet stcSecSrcR2.txt
    scalar mu_y = mean(y)
    scalar STC = sum((y - mu_y)^2)
    scalar SEC = sum((yhat - mu_y)^2)
    scalar SRC = sum(ehat^2)

    printf "STC = %.4f   SEC = %.4f   SRC = %.4f   (SEC+SRC = %.4f)\n", STC, SEC, SRC, SEC+SRC
    printf "R2 (a mano)           = %.6f\n", SEC/STC
    printf "R2 de Gretl ($rsq)      = %.6f\n", $rsq
end outfile

outfile --quiet mediasYpuntoDeLasMedias.txt
    scalar mu_x = mean(x)
    scalar mu_yhat = mean(yhat)

    printf "mu_y = %.10f   mu_yhat = %.10f   (el ajuste preserva la media)\n", mu_y, mu_yhat
    printf "beta1 + beta2*mu_x = %.10f   (la recta pasa por el punto de las medias)\n",
↳ $coeff(const) + $coeff(x)*mu_x
    printf "Diferencia maxima |(yhat-mu_yhat) - beta2*(x-mu_x)| = %.10f\n", max(abs((yhat-mu_yhat) -
↳ $coeff(x)*(x-mu_x)))
```

```

end outfile

outfile --quiet pendienteComoCorrelacion.txt
  scalar rho_xy = corr(x,y)
  scalar beta2_formula = rho_xy * sd(y)/sd(x)

  printf "rho_xy                = %.6f\n", rho_xy
  printf "beta2 (formula rho*sy/sx) = %.6f\n", beta2_formula
  printf "beta2 ($coeff(x) de Gretl) = %.6f\n", $coeff(x)
  printf "rho_xy^2                = %.6f\n", rho_xy^2
  printf "R2 ($rsq de Gretl)       = %.6f\n", $rsq
end outfile

nulldata 60 --preserve
set seed 24680

series x2 = uniform(0, 10)
series u2 = normal(0, 2)
series y2 = 2 + 3*x2 + u2

ols y2 const x2
series ehat2 = $uhat

outfile --quiet verificacionMuestraGrande.txt
  printf "Suma de residuos      sum(ehat2)    = %.10f    (debe ser 0, exactamente)\n", sum(ehat2)
  printf "Suma de x2*residuos sum(x2*ehat2) = %.10f    (debe ser 0, exactamente)\n", sum(x2*ehat2)
  printf "beta2 estimado (pendiente) = %.4f    (valor generador = 3, pero NO se recupera\n"
  ↪ exacto)\n", $coeff(x2)
  printf "beta1 estimado (constante) = %.4f    (valor generador = 2, pero NO se recupera\n"
  ↪ exacto)\n", $coeff(const)
end outfile

outfile --quiet sinConstante.txt
  ols y2 const x2 --quiet
  scalar b2_con = $coeff(x2)
  scalar mu_yhat_con = mean($yhat)

  ols y2 x2                                     # sin constante
  series ehat_sc = $uhat
  printf "\nsuma de residuos              sum(ehat)    = %.4f    (ya NO es cero)\n", sum(ehat_sc)
  printf "suma de x2*residuos            sum(x2*ehat) = %.10f    (esta sigue siendo cero)\n",
  ↪ sum(x2*ehat_sc)
  printf "media de y2 = %.4f    media del ajuste sin constante = %.4f    (con constante: %.4f)\n",
  ↪ mean(y2), mean($yhat), mu_yhat_con
  printf "pendiente sin constante = %.4f    (con constante: %.4f; generador: 3)\n", $coeff(x2), b2_con
end outfile

```

Enlace al guión: [S11-Prct-A-mco-simulado.inp](#)

Respuestas

1. ¿Por qué Gretl reproduce exactamente $\hat{\beta}_1 = 2, \hat{\beta}_2 = 3$? Porque `ols` no hace más que resolver, internamente, las mismas dos condiciones de ortogonalidad que resolvimos a mano en la lección 7. Como este ejemplo se construyó a propósito con un vector $\mathbf{u}$ cuya covarianza muestral con $\mathbf{x}$ es cero (y de media cero), el sistema de ortogonalidad recupera exactamente los coeficientes generadores 2 y 3, puesto que el vector de residuos es necesariamente igual a $\mathbf{u}$ —no es una propiedad general de MCO, sino consecuencia del diseño deliberado del ejemplo.
2. ¿Por qué la ortogonalidad es exacta y $\hat{\beta}_2$ solo se aproxima? Porque la ortogonalidad ($\sum \hat{e}_i = 0$, $\sum x_i \hat{e}_i = 0$) es una consecuencia *algebraica* de cómo MCO define $\hat{\beta}_1, \hat{\beta}_2$ (las dos condiciones que se imponen para construirlos): se cumple exactamente para *cualquier* conjunto de datos, sin excepción. En cambio, que $\hat{\beta}_2$ coincida con el valor 3 usado para generar los datos es una cuestión distinta: depende de que la perturbación u_2 sea ortogonal a los regresores, es decir, de media nula y que tenga, para esa muestra concreta, covarianza exactamente nula con x_2 —algo que ocurre por diseño en la Actividad 1, pero que una perturbación genuinamente aleatoria (Actividad 5) solo cumple *aproximadamente*, con un error que depende de la muestra concreta.
3. Diferencia entre $\mathbf{u}$ y $\hat{\mathbf{e}}$. $\mathbf{u}$ es el vector que *nosotros* usamos para *generar* los datos (una elección de diseño, no observable en la práctica real); $\hat{\mathbf{e}}$ es el vector de residuos que MCO *calcula* a partir de los datos observados, sin conocer $\mathbf{u}$. Coinciden exactamente solo en el caso especial de la Actividad 1, porque ahí $\mathbf{u}$ se construyó deliberadamente ortogonal a los regresores —la misma situación exacta que ya advertimos en la lección 7 y que aquí volvemos a subrayar—. En la Actividad 5, con perturbación genuinamente aleatoria, $\hat{\mathbf{e}} \neq \mathbf{u}_2$ en general.
4. Signo de $\hat{\beta}_2$ y de ρ_{xy} . Porque $\hat{\beta}_2 = \rho_{xy} \sigma_y / \sigma_x$ (lección 7), y tanto σ_y como σ_x son siempre no negativas (son normas). Por tanto, el cociente σ_y / σ_x nunca cambia el signo: el signo de $\hat{\beta}_2$ es siempre el mismo que el de ρ_{xy} .
5. Con $n = 600$ en lugar de $n = 60$. Cabe esperar que $\hat{\beta}_2$ se acerque *todavía más* a 3: con muestras más grandes, la covarianza muestral entre la perturbación u_2 y el regresor x_2 tiende a acercarse a cero (si, como es el caso, u_2 se generó independientemente de x_2), de modo que el “desajuste” entre $\hat{\beta}_2$ y el valor generador 3 tiende a reducirse. Esta es una primera intuición, sin demostrar todavía, de la propiedad de *consistencia* de un estimador, que se estudiará con más detalle más adelante en el curso.
6. *Sin constante*. Al quitar $\mathbf{1}$, el subespacio sobre el que se proyecta es la recta $\mathcal{L}(\mathbf{x})$, con un solo generador; el residuo es ortogonal a ese único generador —de ahí $\sum x_i \hat{e}_i = 0$ — pero nada le obliga a ser ortogonal a $\mathbf{1}$, que ya no está en el subespacio. Por eso los residuos no suman cero, el ajuste no preserva la media y la recta no pasa por el punto de las medias: todas esas propiedades eran consecuencias de la primera condición de ortogonalidad, y esa condición ha desaparecido.