Análisis Avanzado de Datos

Nicolás López

Primer semestre de 2026

Introducción a la clase

Detalles del curso:

- Análisis avanzado de datos (AAD).
- 8 sesiones. Sede Claustro UR. Salón BOOLE.
- Horario: sábados de 7 a. m. a 10 a. m.
- Modalidad: presencial.
- Profesor: Nicolás López.

Prerrequisitos

- Fundamentos de programación en R.
- Fundamentos de AED (Análisis Estadístico de Datos): estadística introductoria.
- Contenido AED.
- Modelos lineales: RLS y RLM.
- Análisis en componentes principales.
- Distribución normal multivariada.
- Métodos de clasificación: k-means y aglomeramiento jerárquico.

Programa

Disponible en eaulas.

Principios de R

Principios de R

Herramienta de gran importancia en el análisis de datos, particularmente en el contexto de AAD:

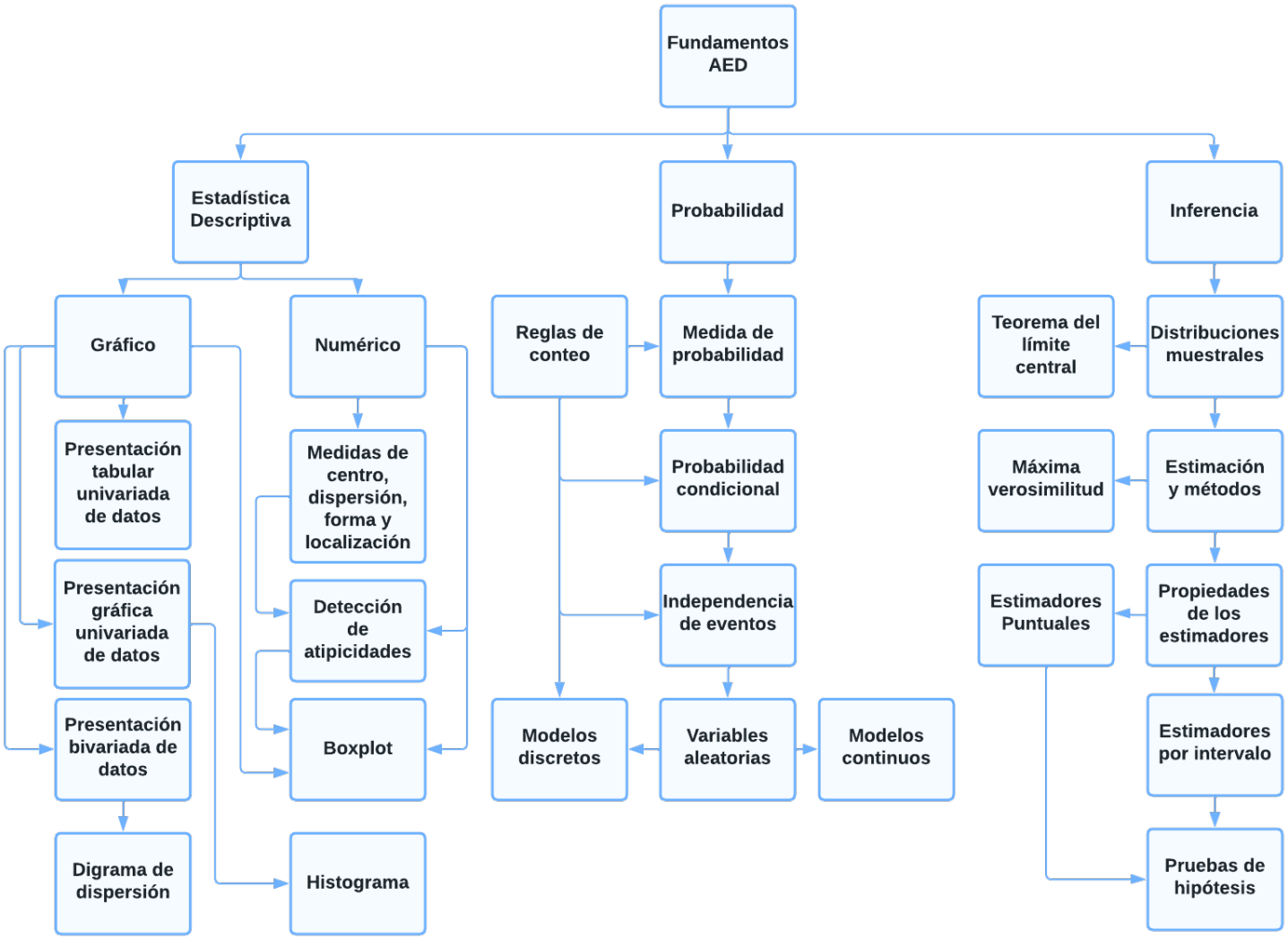
- Innovación en investigación y modelamiento: ej. BTM.
- Gran trayectoria en visualización efectiva y manipulación de datos: ej. tidyverse y ggplot.
- Modelamiento avanzado altamente documentado: gamlss.

Instrucciones de instalación local disponibles en eaulas. También existe opción remota (antiguo RStudio Cloud).

Laboratorios introductorios disponibles en eaulas. Se recomienda realizar los laboratorios introductorios 0 y 1.

Estadística introductoria

Estadística introductoria



Aprendizaje estadístico

Introducción

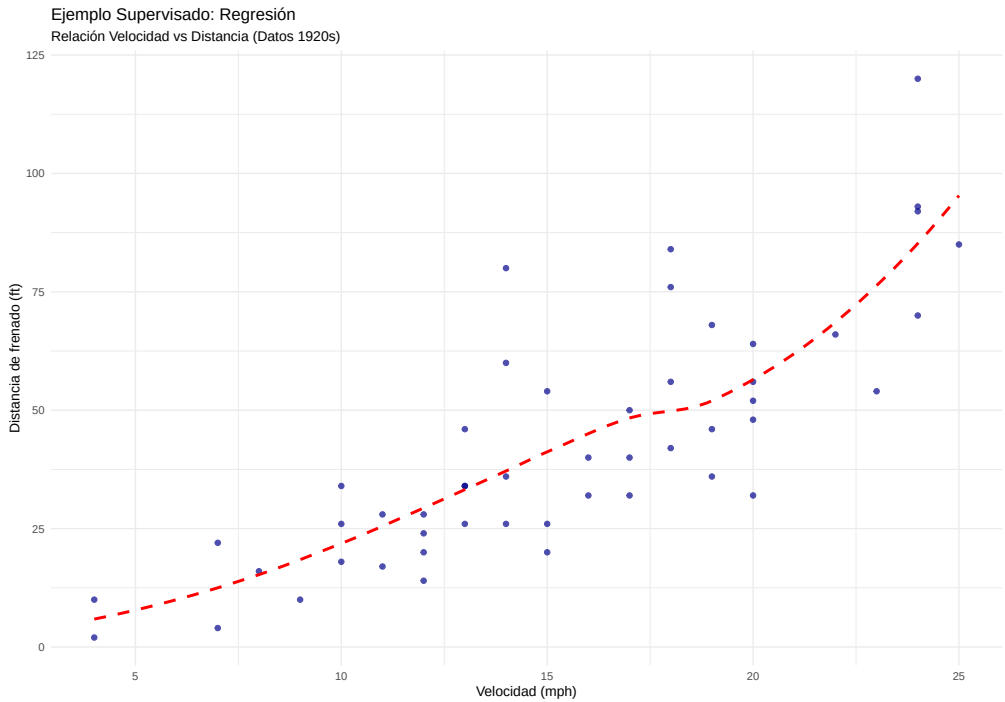
Aprender el proceso subyacente generador de datos. El proceso es formalizado matemáticamente en el aprendizaje estadístico y se clasifica en dos grupos:

- Aprendizaje supervisado: se tiene un resultado (*outcome*) que guía el proceso de aprendizaje (ej. identificación de dígitos).
- Aprendizaje no supervisado: no se tiene una medición de un resultado para guiar el aprendizaje (ej. clasificación de carros basada en características).

En ambos escenarios se cuenta con un conjunto de covariables (*features*) que permiten el aprendizaje.

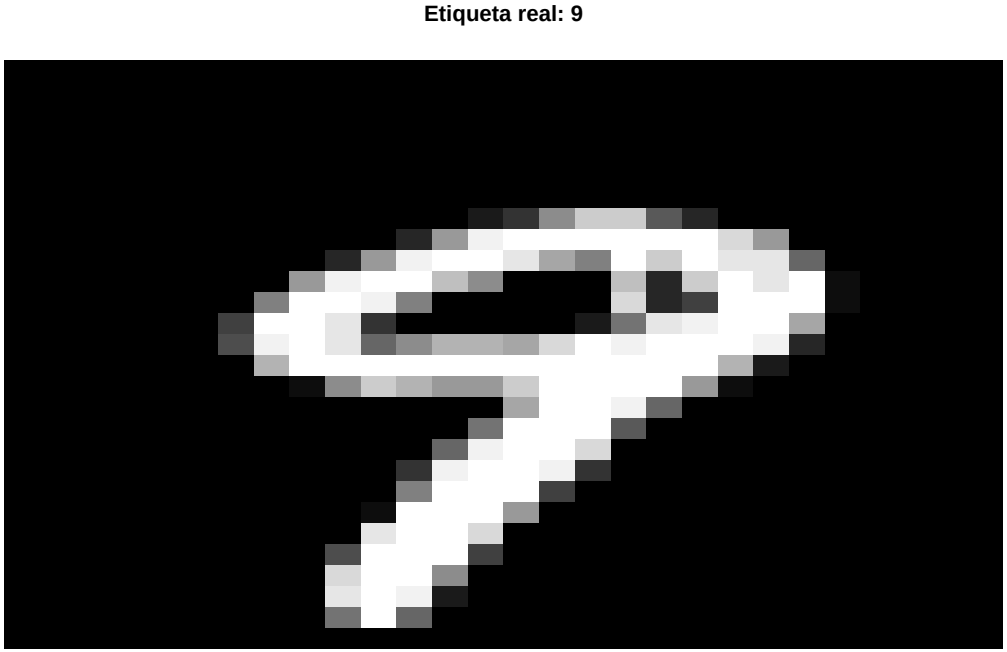
¿Aprendizaje supervisado o no supervisado?

Ejemplo 1



Función f tal que $Y \approx f(X)$.

Ejemplo 2



Tipos de variables

Dependiendo de la naturaleza de las variables (cuantitativas vs. cualitativas), seleccionamos el método:

- Cuantitativas.
- Cualitativas.

Esto aplica tanto para las covariables como para la variable respuesta.

Existe un mayor refinamiento en la categorización, pero por ahora basta entender que **el mismo problema de aprendizaje (supervisado o no) puede darse para diferente naturaleza de las variables**:

- Análisis de regresión lineal (simple/múltiple): supervisado con respuesta cuantitativa.
- Análisis de componentes principales: no supervisado con covariables cuantitativas.
- Análisis de correspondencias (simple/múltiple): no supervisado con covariables cualitativas.
- Árbol de decisión: supervisado con respuesta cualitativa.

Recurso recomendado:
YouTube: Supervised vs Unsupervised Learning

Definiciones importantes

Se destacan 4 elementos fundamentales en el aprendizaje estadístico dada la revisión anterior:

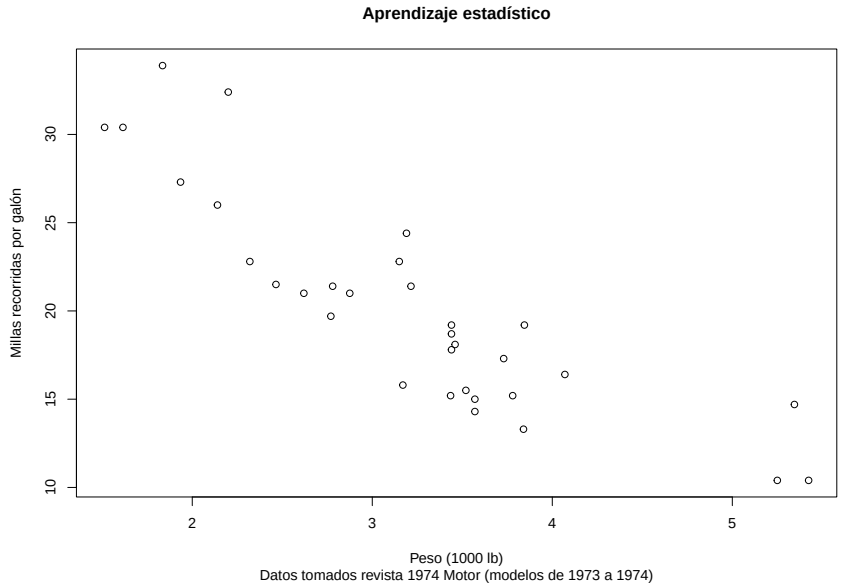
- Proceso generador P.
- Variable de entrada/covariable/input: X (uni/multivariada).
- Variable de salida/variable respuesta/output: Y (usualmente univariada).
- Observaciones/realizaciones/mediciones: $(x_1, y_1), ..., (x_n, y_n)$.

Estas mediciones son arregladas en una matriz diseño $\mathbf{X}$.

RLS

Introducción

Si analizamos con detenimiento la gráfica de dispersión de los datos de velocidad, podemos establecer con claridad una relación entre estas dos variables.



Si Y representa "Millas recorridas por galón" y X es igual a "Peso (1000 lb)", podemos representar una relación **determinística** entre las variables de la siguiente forma:

$$Y = \beta_0 + \beta_1 X$$

Por lo tanto, una relación **aleatoria** que ajusta por el error está formulada por:

$$Y = \beta_0 + \beta_1 X + \epsilon$$

La ecuación anterior se llama *Modelo de Regresión Lineal Simple*.

El modelo aleatorio está completamente especificado cuando definimos las características aleatorias del error; suponemos que $\epsilon \sim N(0, \sigma^2)$.

Tanto el error como la respuesta son aleatorios. Bajo normalidad de ϵ, Y también es normal (¿por qué?).

Entonces la respuesta esperada de Y dado X es:

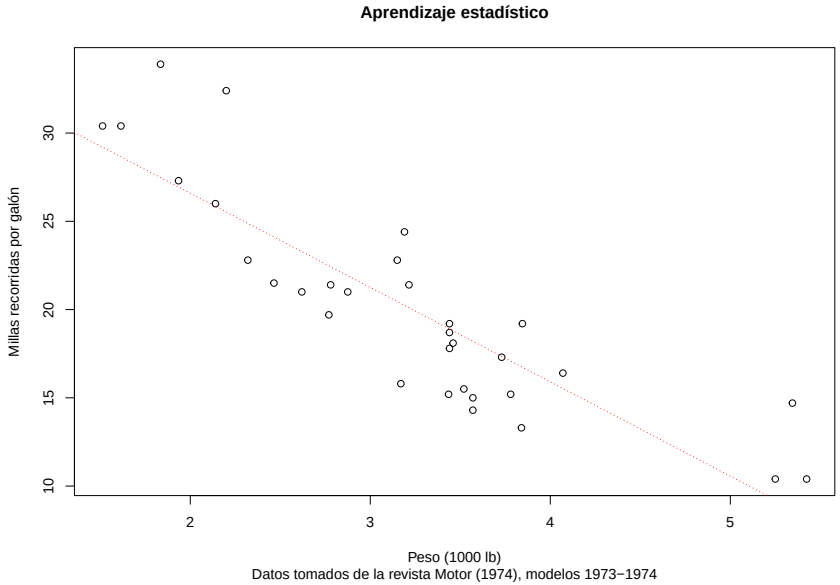
$$E(Y|X) = \mu_{Y|X} = E(\beta_0 + \beta_1 X + \epsilon) = \beta_0 + \beta_1 X$$

Con una varianza igual a

$$V(Y|X) = \sigma_{Y|X} = V(\beta_0 + \beta_1 X + \epsilon) = V(\epsilon) = \sigma^2$$

El modelo de regresión $\mu_{Y|X}$ es una línea recta de valores promedio; esto es, la altura de la línea de regresión en X es el valor esperado $\beta_0 + \beta_1 X$.

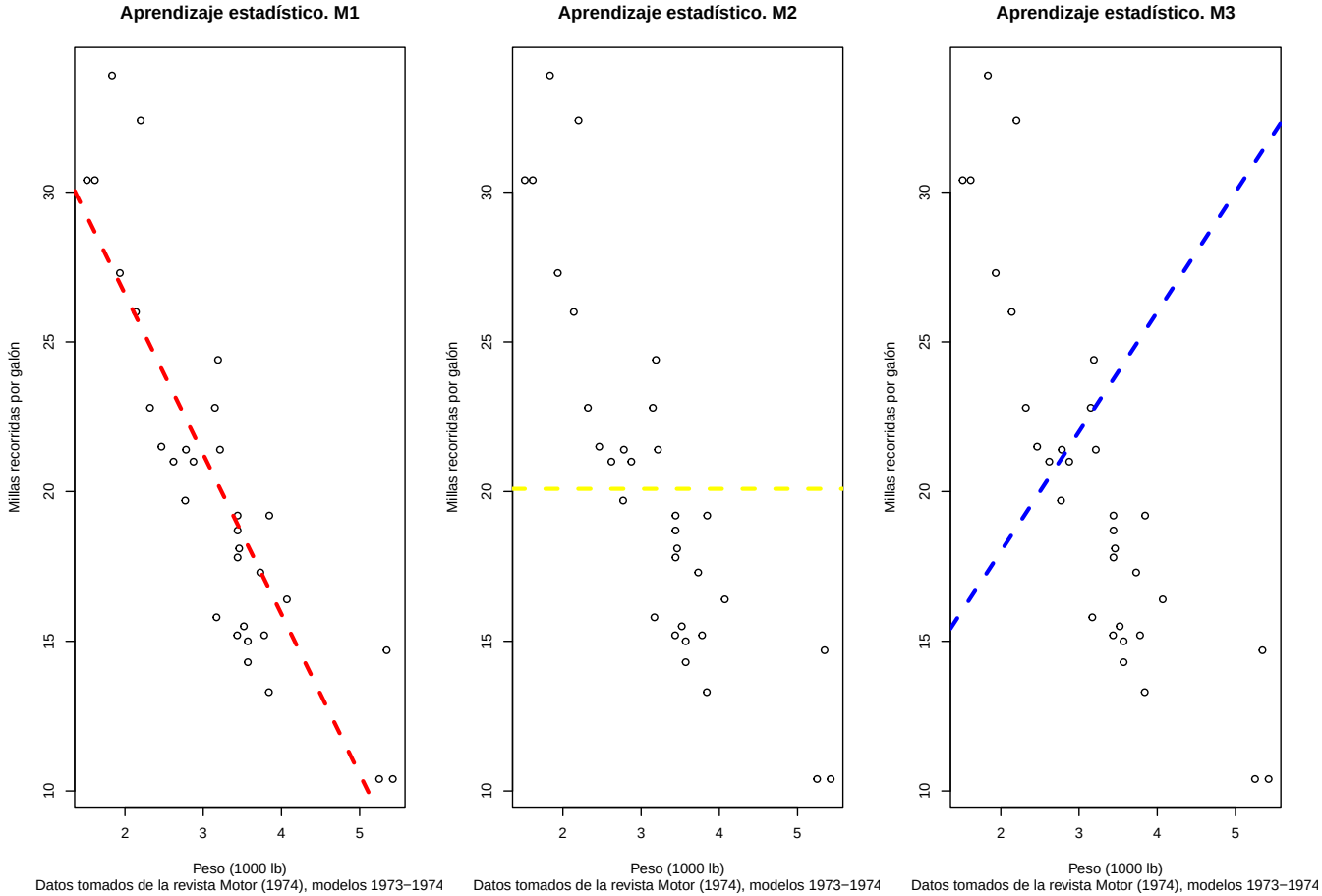
Esto implica que hay una distribución de valores de Y para cada X, y que la varianza σ^2 de esta distribución es igual en cada X.



Con una realización de n observaciones $(x_1, y_1), ..., (x_n, y_n)$ se estiman los parámetros del modelo: $\hat{\beta}_0$, $\hat{\beta}_1$ y $\hat{\sigma}$.

Ajuste por mínimos cuadrados

Para encontrar la línea que mejor se ajusta a los datos, necesitamos una medida de calidad del ajuste. Entre los tres candidatos siguientes es claro cuál resulta en una menor **suma de cuadrados**:



Es claro que:

$$SC(M1) < SC(M2) < SC(M3)$$

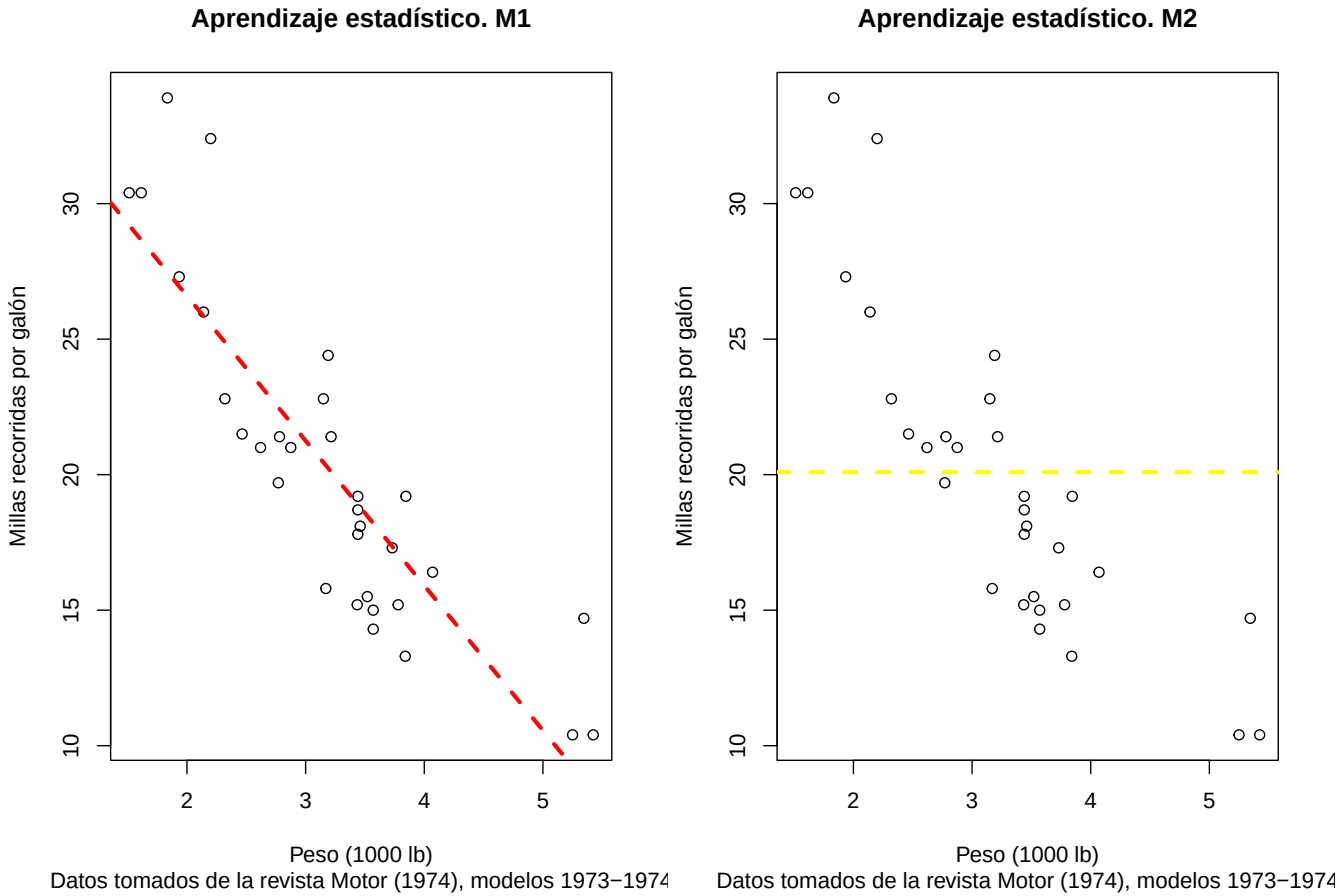
con

$$SC(M_j) = \sum_i (y_i - \hat{y}_{i,M_j})^2$$

En donde:

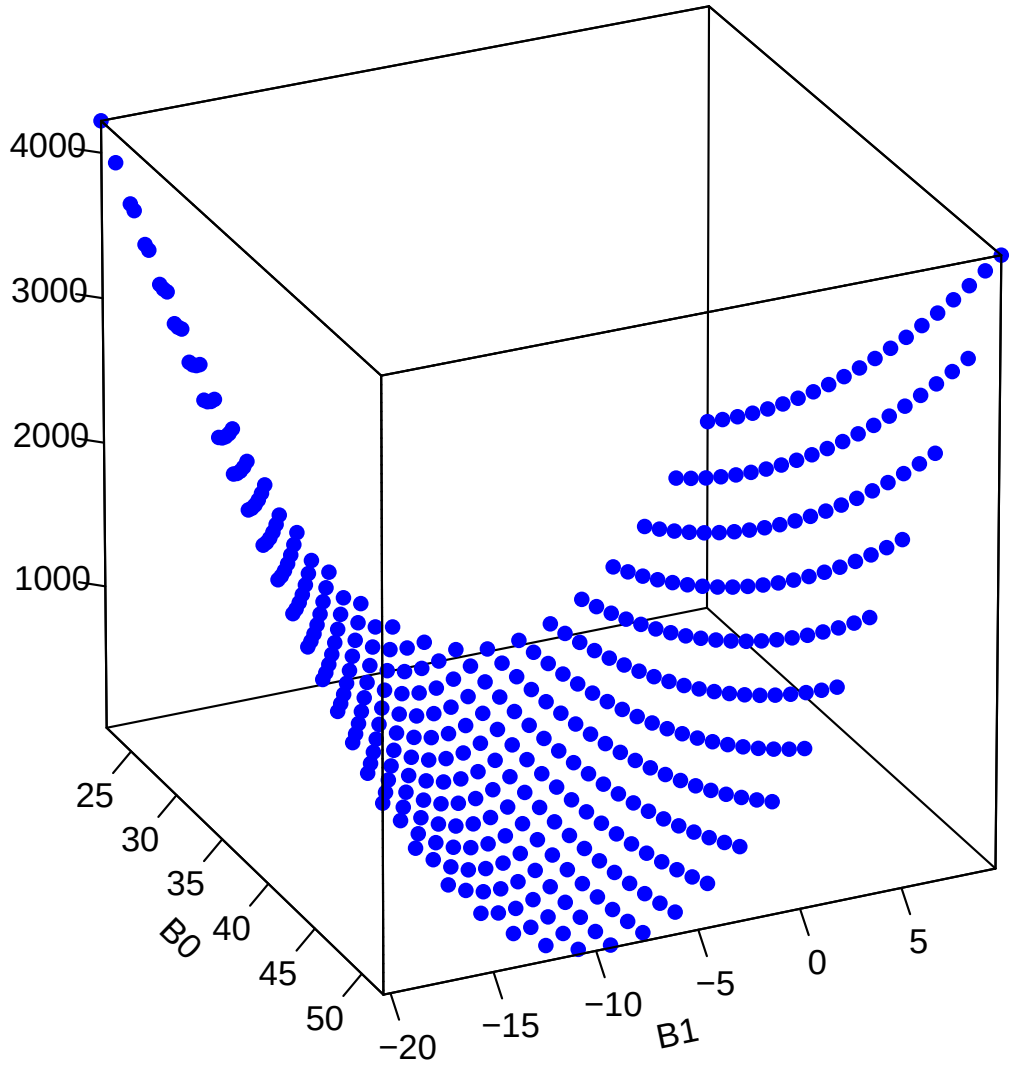
- El **modelo simple o reducido** (M2) no utiliza información de X para encontrar el valor de Y; asume el valor promedio de Y como modelo marginal. Este es considerado la línea base (*baseline*).
- La estimación por mínimos cuadrados consiste en encontrar β_0 y β_1 de tal forma que minimicen la suma de cuadrados, es decir, los **residuales al cuadrado** (¿qué sucedió con σ en la estimación?).

Es evidente que hay menor variabilidad alrededor de $M1$ que alrededor de $M2$; es decir, que la variación de las millas recorridas es explicada por el peso del vehículo. ¿Cómo formalizar esta noción?



La SC del modelo simple ($SC(M2) = SC_T$) cuantifica la variabilidad total de Y respecto a su media. Por su parte, $SC(M1) = SC_E$ mide la variación remanente al ajustar el modelo mediante mínimos cuadrados.

Fíjese que para cada modelo se tiene un intercepto y una pendiente (β_0, β_1) y, con ello, se obtiene SC_E, es decir, $SC_E(\beta_0, \beta_1)$. Los valores estimados $(\hat{\beta}_0, \hat{\beta}_1)$ por mínimos cuadrados minimizan la función de error:



En este caso $(\hat{\beta}_0 = 37.3, \hat{\beta}_1 = -5.3)$.
¿Cómo se verán las curvas de nivel de la función $SC_E(\beta_0, \beta_1)$?

Para cualquier pareja de parámetros, comparar SC_T con SC_E cuantifica la reducción de la variabilidad bajo el modelo lineal en X. La reducción de la varianza en Y explicada por X bajo el modelo es igual a:

$$SC_M = SC_T - SC_E$$

Así, SC_M cuantifica la reducción en la variación total al ajustar el modelo lineal en X. Al igual que SC_E, SC_M es función de (β_0, β_1), es decir, $SC_M(\beta_0, \beta_1)$, la cual es maximizada en $(\hat{\beta}_0, \hat{\beta}_1)$ por mínimos cuadrados (¿qué unidades tiene SC_M?).

La SC_M es estandarizada como:

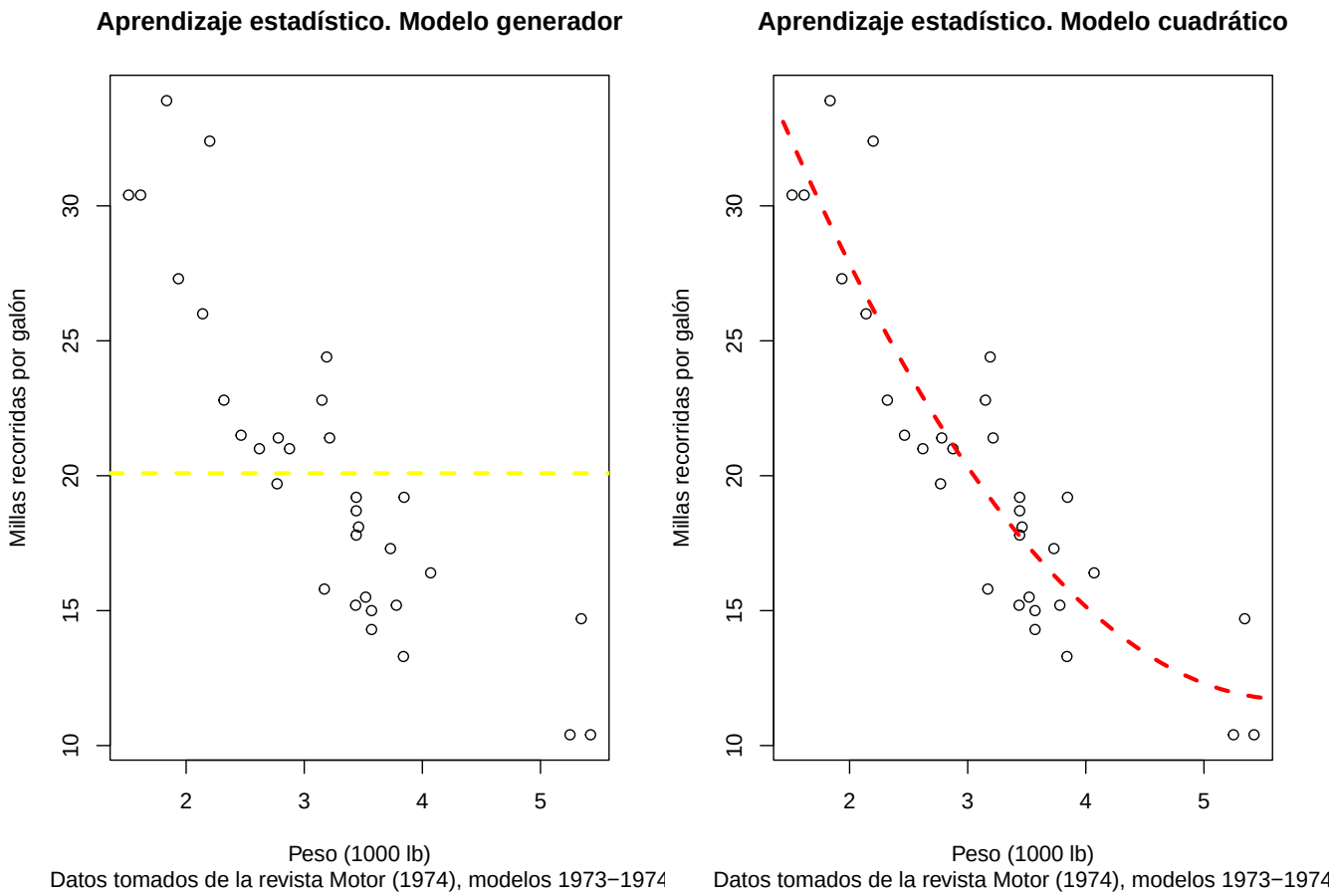
$$R^2 = \frac{SC_M}{SC_T}$$

- Con R^2 cuantificamos la **proporción** de la varianza en Y explicada por el regresor X.
- Si R^2 es cercano a 1, SC_M se acerca a SC_T, es decir, el *modelo explica* la variabilidad en Y (¿cómo lo medimos objetivamente?).
- Si R^2 es cercano a 0, SC_M se aleja de SC_T, es decir, el *modelo no explica* la variabilidad en Y (¿cómo lo medimos objetivamente?).

En nuestro ejemplo $R^2 = 0.75$, con lo cual hay una reducción del 75% en la varianza de las millas recorridas al considerar linealmente el peso del vehículo.

Note que esta definición de R^2 aplica para situaciones más generales, por ejemplo, para un modelo con componente lineal y **cuadrático** en X:

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2$$

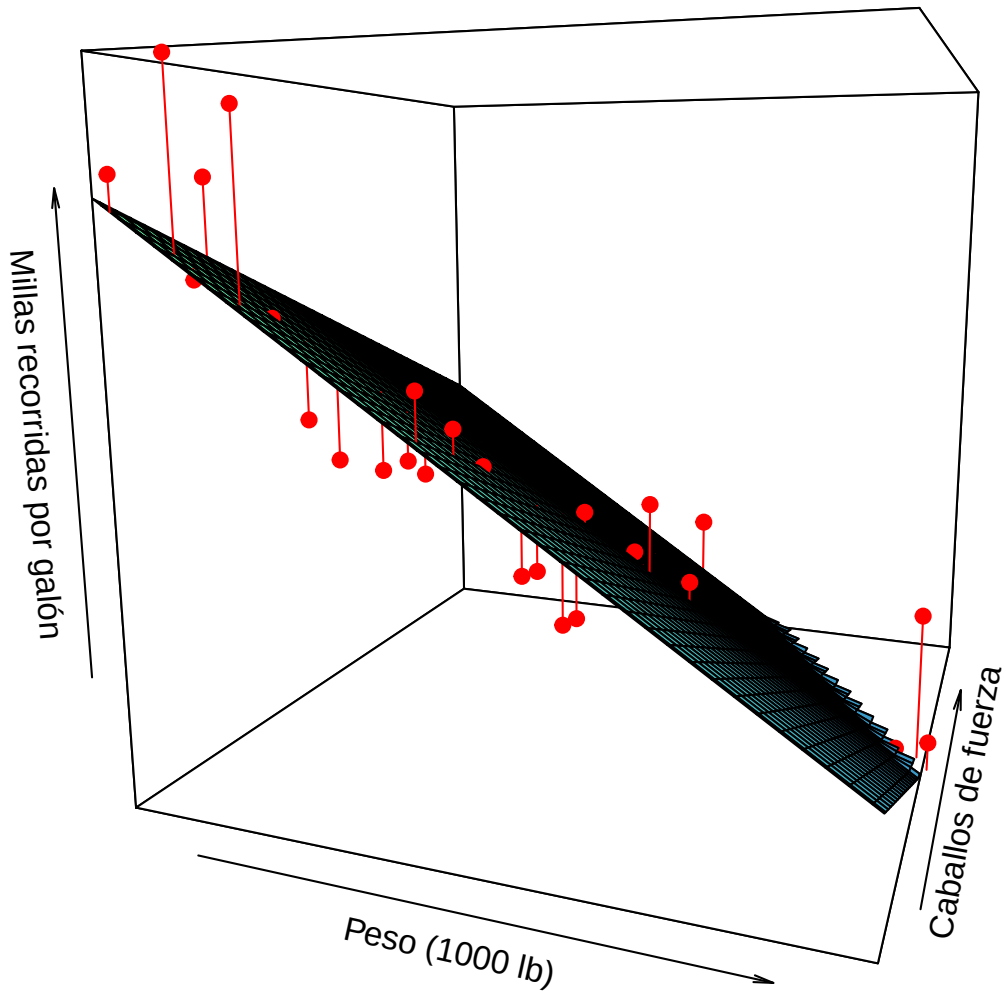


Bajo este modelo, tenemos un $R^2 = 0.81$.

Y bajo este modelo, tenemos un $R^2 = 0.81$.

También para el escenario multivariado, con Z igual a los caballos de fuerza del vehículo:

$$Y = \beta_0 + \beta_1 X + \beta_2 Z$$



Con un $R^2 = 0.82$.

Note que incorporar una variable al modelo puede o no ser relevante para explicar la variabilidad en Y.

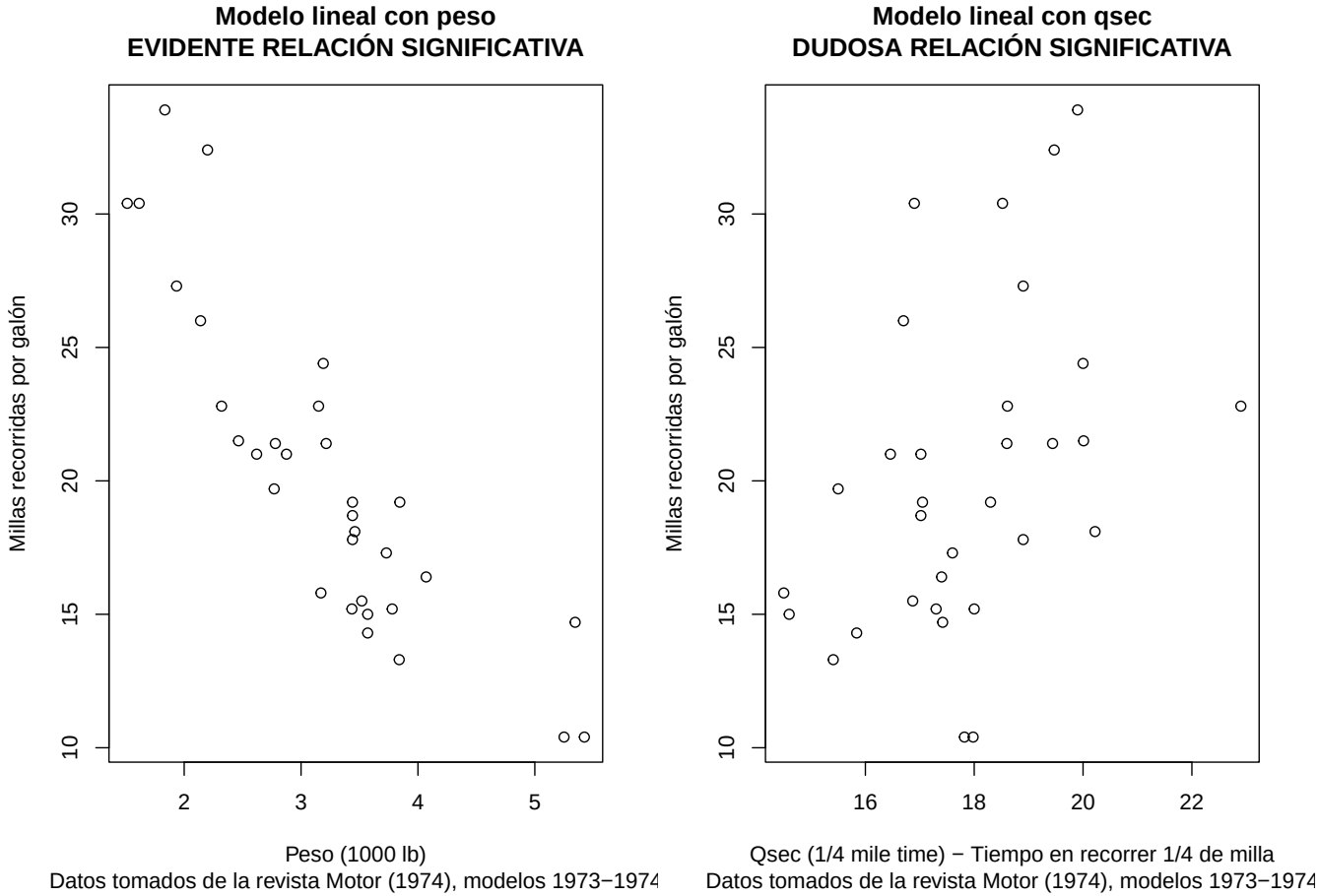
Si la variable Z no tiene efecto en la respuesta:

- Al minimizar SC_E, se obtiene $\beta_2 = 0$ y así $Y = \beta_0 + \beta_1 X + 0Z = \beta_0 + \beta_1 X$, es decir, el modelo de RLS.
- SC_M es el mismo bajo los dos modelos; es decir, añadir Z no mejora ni empeora R^2.
- Añadir variables mantiene igual o incluso **mejora** R^2, aunque no sean útiles para explicar Y.
- Variables irrelevantes pueden estar correlacionadas con Y por coincidencia.
- Más variables irrelevantes aumentan la probabilidad de que esto ocurra, mejorando artificialmente R^2.

En la práctica se reporta usualmente un R^2 ajustado por el número de variables.

Significancia de la regresión

Volviendo a RLS, no es claro cómo determinar o medir qué tan significativo resulta el valor de R^2. ¿En qué punto de R^2 el modelo es realmente mejor con o sin la covariable?



Recuerde que:

- La variación total en los datos, SC_T, corresponde a la variación total al asumir el modelo reducido.
- Al considerar la covariable mediante el modelo lineal disminuimos este error; la nueva variación la llamamos SC_E.
- La diferencia entre SC_T y SC_E corresponde a la variabilidad explicada por el modelo, es decir, SC_M.

SC_T es una suma de cuadrados, por lo cual tiene **grados de libertad** (gl) asociados.

- Para el cálculo de SC_T se requieren $gl(SC_T) = n - 1$ unidades.
- $y'_1 = y_1 - \bar{y}$
- ...
- $y'_{n-1} = y_{n-1} - \bar{y}$
- Como $\sum_i y'_i = 0$, se tiene $y'_n = -\sum_{i=1}^{n-1} y'_i$
- Por tanto, $SC_T = \sum_i (y_i - \bar{y})^2 = \sum_i y_i'^2$ tiene $n - 1$ grados de libertad.

SC_E también es una suma de cuadrados, con **grados de libertad** asociados.

- En RLS se estiman $p = 2$ parámetros (intercepto y pendiente).
- Por tanto, $gl(SC_E) = n - p$.

SC_M es igualmente una suma de cuadrados.

- Como $SC_M = SC_T - SC_E$,
  $gl(SC_M) = gl(SC_T) - gl(SC_E) = (n-1) - (n-p) = p - 1$.
- En RLS, $p - 1 = 1$.

De manera semejante a R^2, podemos definir:

$$F' = \frac{SC_M}{SC_E}$$

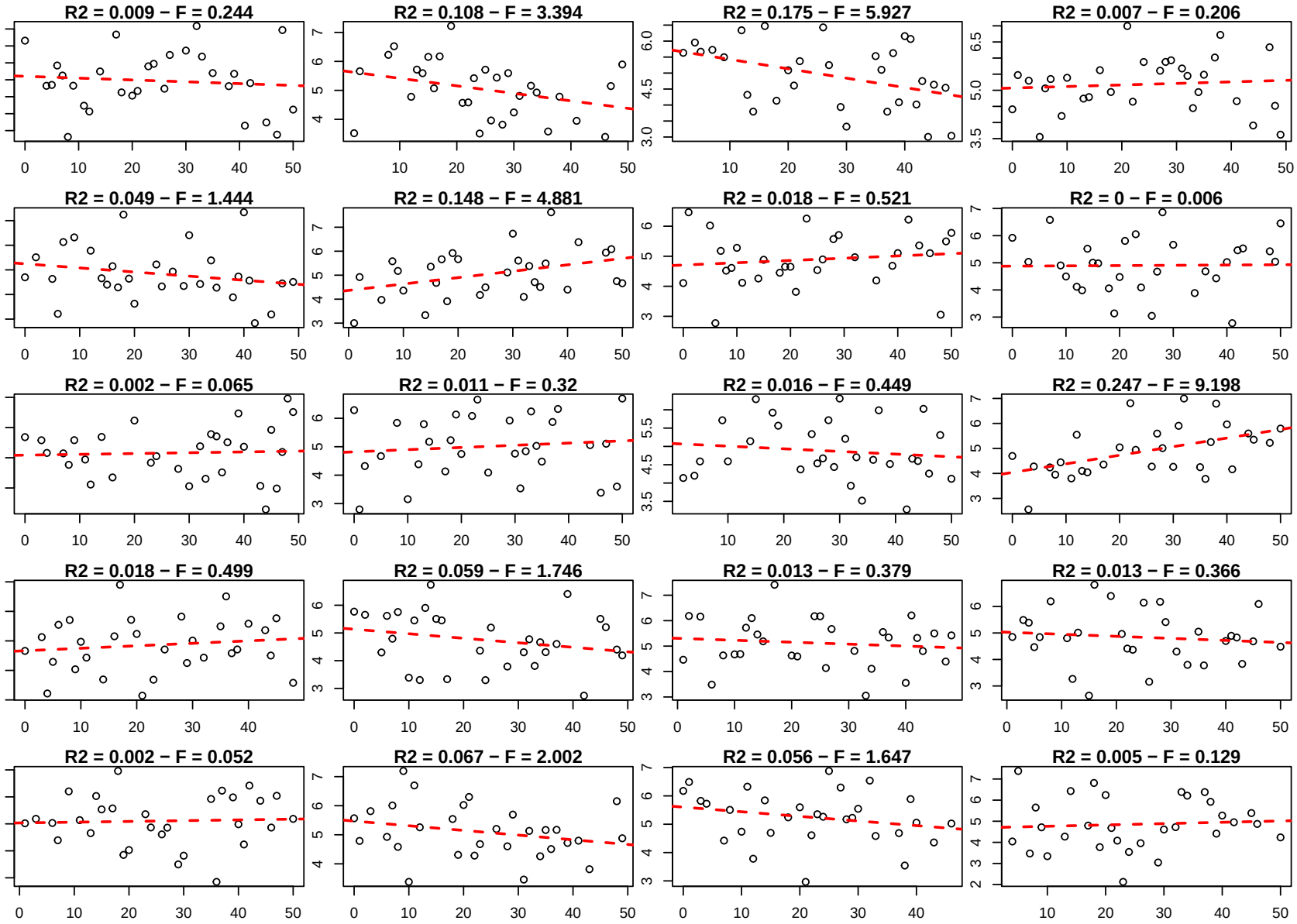
- A mayor F', mayor variabilidad explicada respecto a la no explicada.
- Es una razón, no una proporción, pero comparte interpretación con R^2.

Al normalizar por los grados de libertad:

$$F = \frac{SC_M/gl(SC_M)}{SC_E/gl(SC_E)}$$

- Si la regresión no es informativa ($\beta_i = 0$), F sigue una distribución F.
- Bajo la hipótesis nula, los parámetros son $gl(SC_M)$ en el numerador y $gl(SC_E)$ en el denominador.

Si la regresión no es informativa, los datos podrían verse como:



Finalmente, se destaca que el modelo lineal requiere las siguientes premisas:

- Relación (aproximadamente) lineal.
- Error con media cero.
- Error con varianza constante.
- Errores no correlacionados.
- Errores normalmente distribuidos (necesario para pruebas como la de F).

Estas condiciones no son detectadas mediante R^2 o F, ya que son propiedades globales del modelo.

De la Regresión Lineal a Ridge

Solución del Modelo Lineal Clásico

Mediante mínimos cuadrados, la solución al problema de MCO está dada por

$$\hat{\beta}_{MCO} = (X^T X)^{-1} X^T y$$

Que en la práctica puede presentar algunos inconvenientes al invertir $(X^T X)^{-1}$.

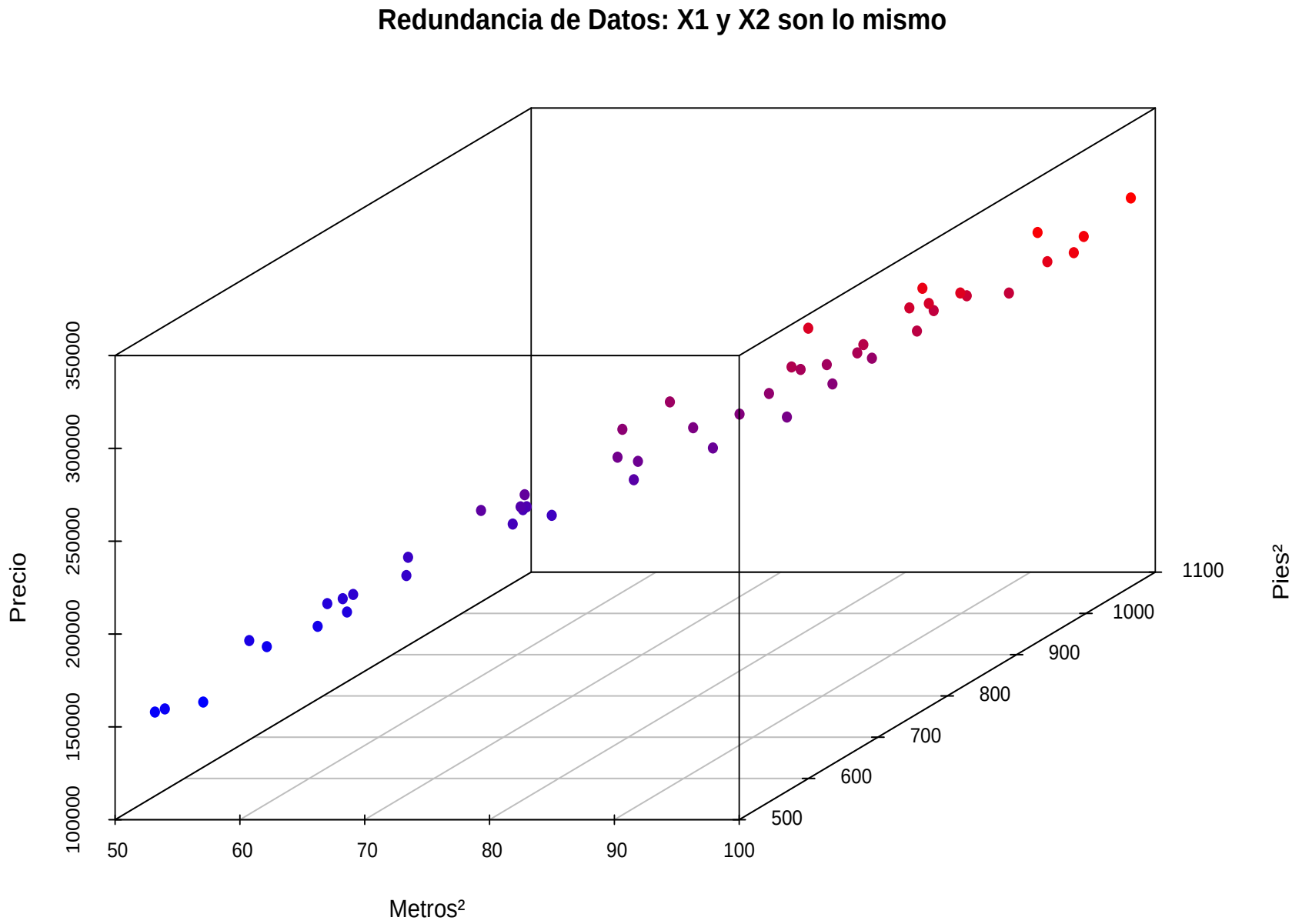
Los Problemas del Mundo Real

Problema 1: Multicolinealidad Perfecta

Supongamos que se predice el precio de casas usando:

- $X_1 =$ metros cuadrados
- $X_2 =$ pies cuadrados (conversión directa: $X_2 = 10.7639 \times X_1$)
- $X^T X$ tiene determinante a cero. Su inversa no existe.

```
# Simulación: Precio de casas con metros
# y pies cuadrados (conversión exacta)
n <- 50
set.seed(1)
metros <- runif(n, 50, 100)  # 50 a 200 metros cuadrados
pies <-  10.7639 * metros     # conversión exacta a pies cuadrados
precio <- 2000 + 3000 * metros + rnorm(n, 0, 10000)
```


Redundancia de Datos: X1 y X2 son lo mismo
Precio
100000
150000
200000
250000
300000
350000
Metros²
50
60
70
80
90
100
Pies²
500
600
700
800
900
1000
1100

Imaginemos que queremos sostener una mesa (el plano de regresión) sobre patas (cada uno de los datos).

- Si las variables X_1 (metros) y X_3 (otra variable como "antigüedad") son independientes, los datos están "esparcidos" por el suelo.
- Si las variables son una línea perfecta (pies son metros por una constante), es como si intentaras sostener la mesa usando solo una cuerda tensa.

Por mas patas que tenga la mesa, si están sobre una misma recta, no tendremos estabilidad y obtenemos infinitas soluciones.

¿Cómo se ve esto en las ecuaciones?

Si la predicción del precio de una casa de 100 $Metros^2$ es 302,000 y solo depende de los metros:

$$Precio = \beta_0 + \beta_1 \times Metros^2 = 2000 + 3000 \times Metros^2$$

Tenemos una única solución una vez ajustado el modelo de regresión por MCO.

Pero si introducimos los Pies (con $1\text{Metro}^2 = 10.764\text{Pies}^2$), el modelo tiene que resolver esto:

$$Precio = \beta_0 + \beta_1 \times Metros^2 + \beta_2 \times Pies^2$$

Aquí es donde aparecen las infinitas soluciones. Para las covariables:

$$\beta_1 = 3000, \beta_2 = 0$$

$$\beta_1 = 0, \beta_2 = 278.71$$

$$\beta_1 = 1500, \beta_2 = 139.4$$

Tenemos infinitas soluciones de MCO.

Geométricamente, X_1 y X_2 forman una línea. Para que un modelo de regresión lineal sea robusto, necesitamos que las variables independientes tengan suficiente variabilidad y no estén perfectamente correlacionadas.

- Si $\det(X'X)$ es alto: Variables con dispersión saludable y no son redundantes. Esto permite al modelo "distinguir" el efecto individual de cada variable sobre la dependiente (Y).
- Si $\det(X'X)$ es cercano a cero: Multicolinealidad. Las variables están tan relacionadas entre sí que el modelo no sabe a cuál asignarle los cambios en Y.

Problema 2: Varianza Inflada (Caso Cercano a la Multicolinealidad Perfecta)

Imaginemos ahora las mismas dos variables X_1 y X_2, pero ahora casi perfectamente correlacionadas (correlación $= 0.99$). Con el ejemplo anterior, añadiendo un pequeñísimo error a la conversión de pies a metros. Además tenemos nuevamente la tercera variable X_3 sin el problema de multicolinealidad.

```r
# 1. Simulación de datos
n <- 100
set.seed(123)
metros <- runif(n, 50, 200)

# --- CASO 1: SANO (Variables independientes) ---
# Generamos una variable 'Antigüedad' que no depende de los metros
antiguedad <- runif(n, 0, 50)
precio_sano <-  3000*metros - 1500*antiguedad + rnorm(n, 0, 5000)
fit_sano <- lm(precio_sano ~ metros + antiguedad)
fit_sano
```

```
##
## Call:
## lm(formula = precio_sano ~ metros + antiguedad)
##
## Coefficients:
## (Intercept)       metros   antiguedad
##      -866.1       3003.2      -1503.0
```

```r
# --- CASO 2: MULTICOLINEAL (Variables redundantes) ---
# Pies es casi igual a metros (con un ruido mínimo de 0.1)
pies <- (metros * 10.764) + rnorm(n, 0, 0.1)
precio_multi <- 3000*metros + rnorm(n, 0, 5000)
fit_multi <- lm(precio_multi ~ metros + pies)
fit_multi
```

```
##
## Call:
## lm(formula = precio_multi ~ metros + pies)
##
## Coefficients:
## (Intercept)       metros         pies
##        1177        19445        -1529
```

Con los coeficientes del fit_multi:

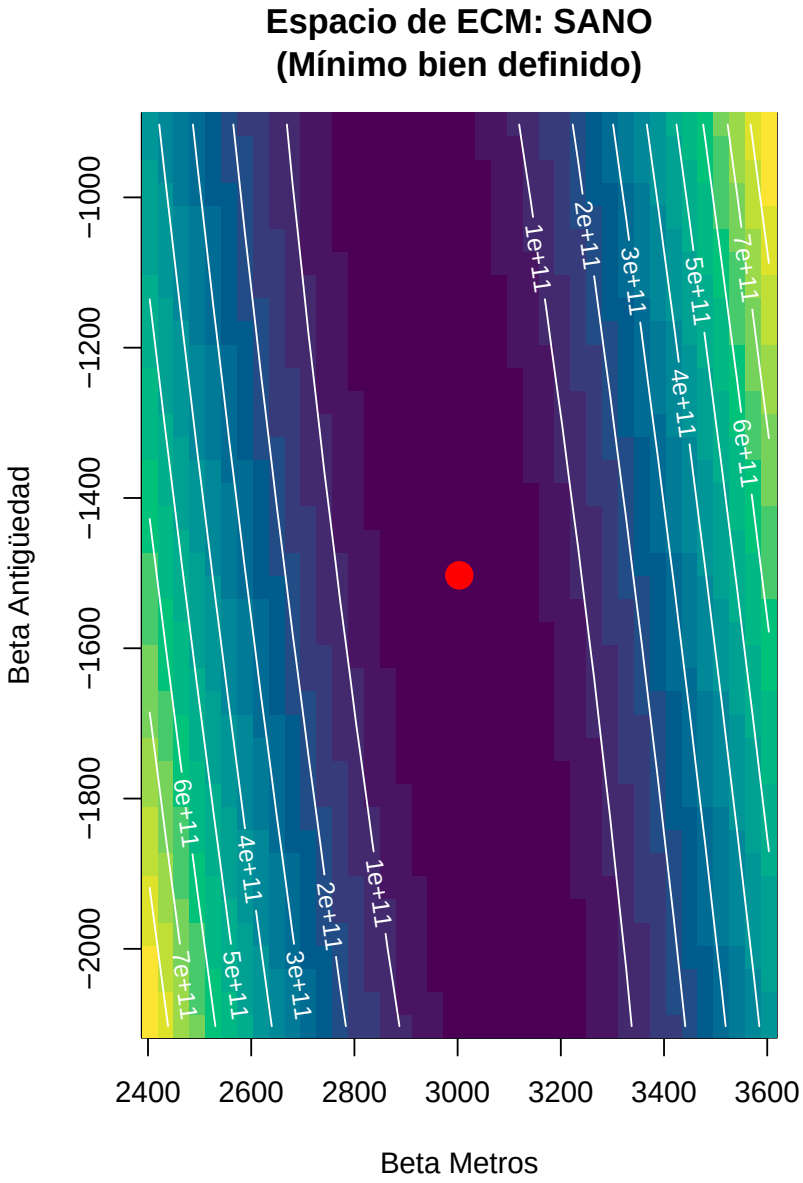
- El modelo dice que cada metro extra suma: $+19.445$.
- El modelo dice que cada pie extra resta: -1.529.

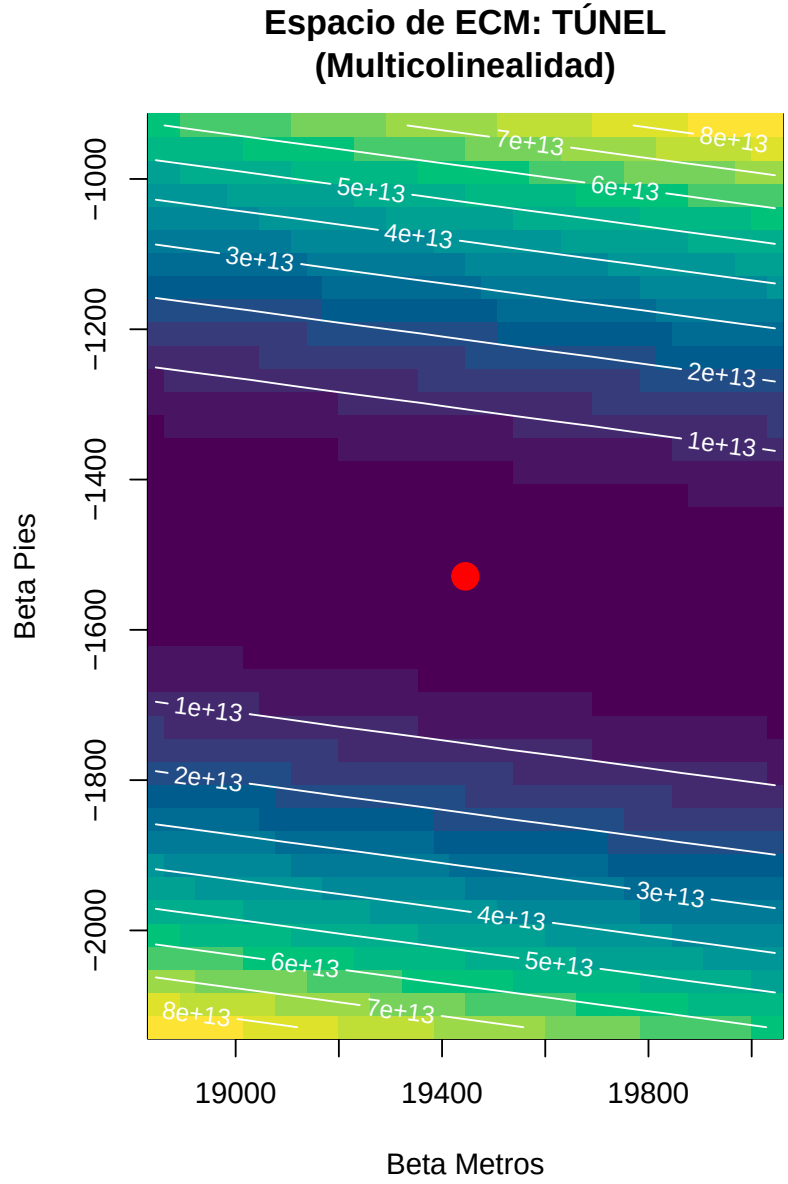
Pero como un $metro^2$ contiene 10.765 $pies^2$, para ver el efecto real de un $metro^2$ según el modelo, debemos compensar:

$$Efecto\ Total = 19,445 + (-1.529 \times 10.765)$$

$$Efecto\ Total = 19,439 - 16.459 = \mathbf{2985.3}$$

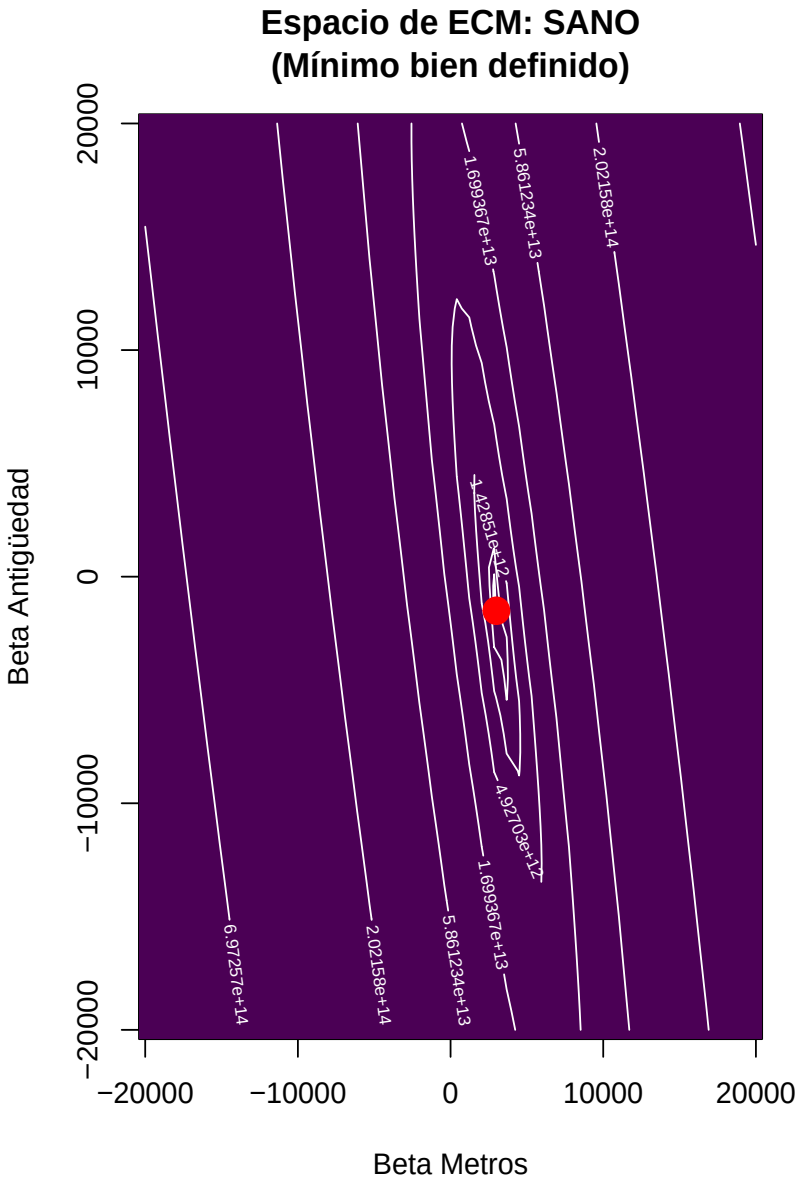
El efecto existe y es cercano al esperado, pero obtuvimos un resultado extraño y contraintuitivo.

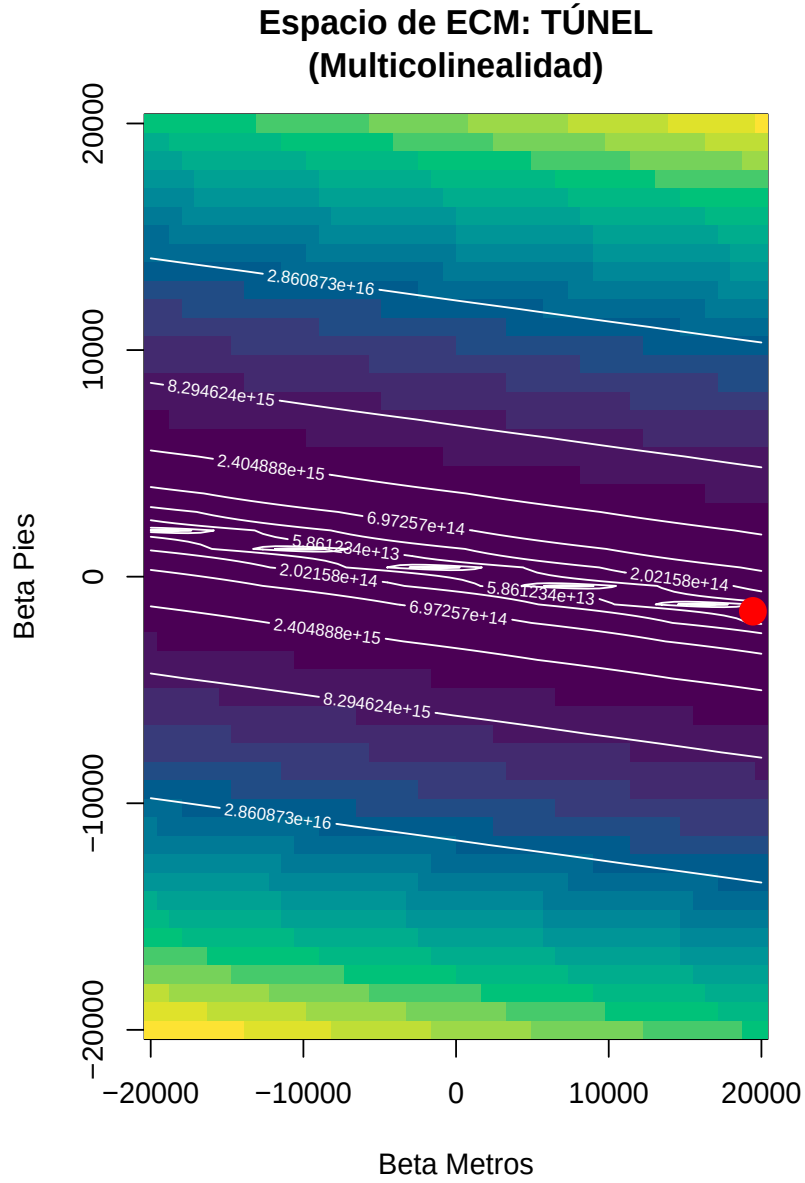

Espacio de ECM: SANO
(Mínimo bien definido)
Beta Antigüedad
Beta Metros
−1000
−1200
−1400
−1600
−1800
−2000
2400
2600
2800
3000
3200
3400
3600
1e+11
2e+11
3e+11
4e+11
5e+11
6e+11
7e+11


Espacio de ECM: TÚNEL
(Multicolinealidad)
Beta Pies
Beta Metros
−1000
−1200
−1400
−1600
−1800
−2000
19000
19400
19800
1e+13
2e+13
3e+13
4e+13
5e+13
6e+13
7e+13
8e+13

Para revelar la inestabilidad, fijamos los ejes de búsqueda en un rango amplio (± 20000).

- Mientras que el modelo sano (izquierda) encuentra una solución única y confinada, el modelo con multicolinealidad (derecha) muestra un 'valle' donde coeficientes absurdamente grandes tienen prácticamente el mismo error que los correctos.


Espacio de ECM: SANO
(Mínimo bien definido)
Beta Antigüedad
Beta Metros
20000
10000
0
−10000
−20000
1.699367e+13
5.861234e+13
2.02158e+14
1.428516e+12
4.927038e+12
6.97257e+14
2.02158e+14
5.861234e+13
1.699367e+13


Espacio de ECM: TÚNEL
(Multicolinealidad)
Beta Pies
Beta Metros
20000
10000
0
−10000
−20000
2.860873e+16
8.294624e+15
2.404888e+15
6.97257e+14
5.861234e+13
2.02158e+14
2.02158e+14
5.861234e+13
6.97257e+14
2.404888e+15
8.294624e+15
2.860873e+16

Al usar el mismo rango (-20000 a 20000):

- En el gráfico sano, el área de "error bajo" es un circulito: el área de confianza es un círculo o una elipse casi circular.
- En el multicolineal, el área es una 'pista de aterrizaje': el área de error bajo se convierte en esa elipse extremadamente alargada.

Bajo multicolinealidad OLS encuentra una solución de la 'pista de aterrizaje'

Si la verdadera relación con Y es que el metraje suma $+3000$, el modelo OLS tiene infinitas formas de sumar 3000:

- Opción A (Estable):

$$\beta_1 = 3000, \beta_2 = 0$$

$$\text{Total} = 3000 + (0 \times 10,765) = 3000$$

- Opción B (Inflada):

$$\beta_1 = 19439, \beta_2 = -1529$$

$$\text{Total} = 19,439 + (-1,529 \times 10,765) = 3000$$

Tanto la combinación estable $(3000, 0)$ como la combinación inflada $(19439, -1529)$ están prácticamente "al mismo nivel" en el fondo de la pista.

Es muy problable que las respuestas bajo multicolinealidad sean las infladas y compensatorias en lugar de estables y con sentido práctico:

- Los coeficientes se calculan $\hat{\beta} = (X'X)^{-1}X'Y$ y

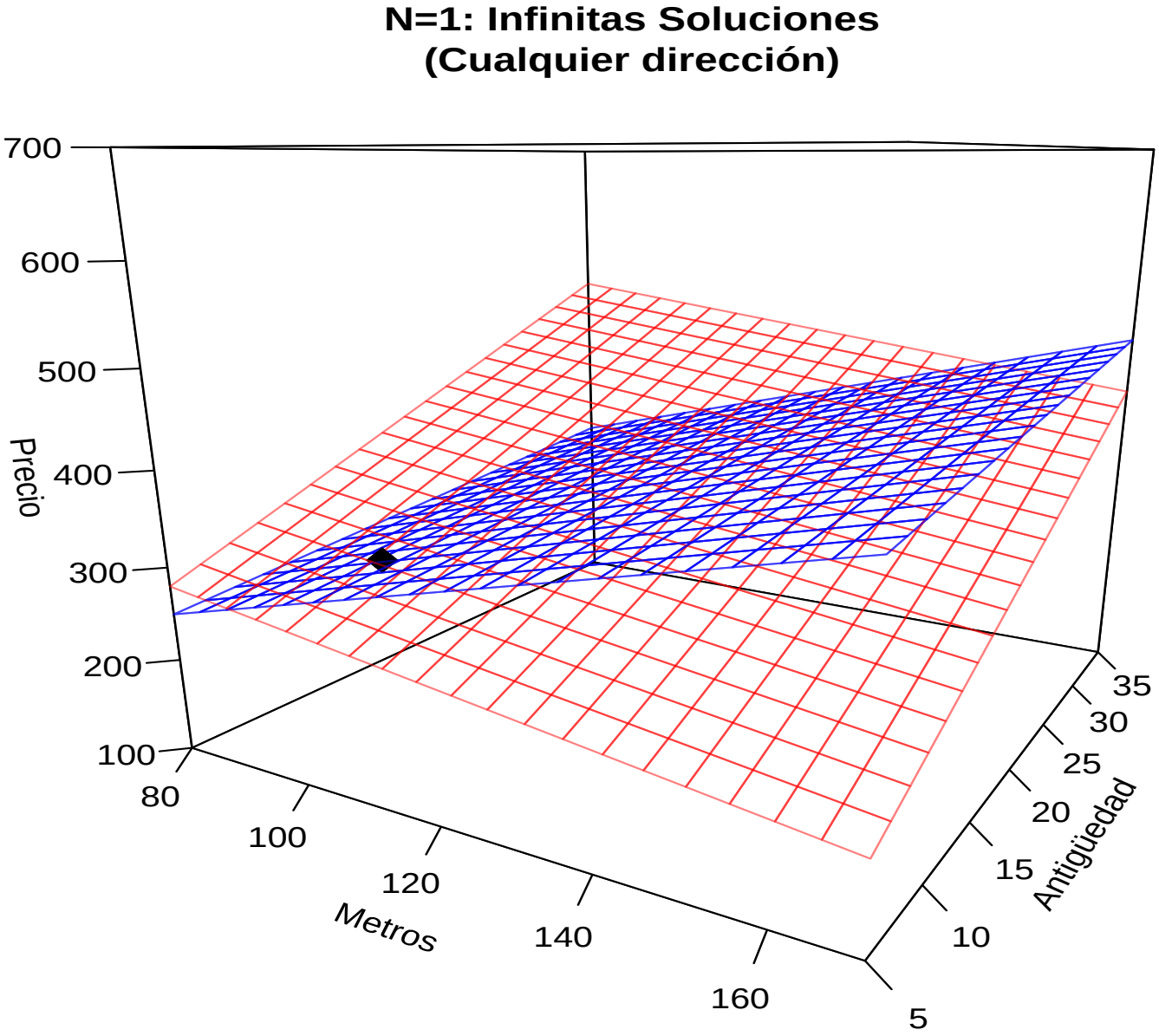
$$(X'X)^{-1} = \frac{1}{det(X^TX)} \cdot adj(X^TX)$$

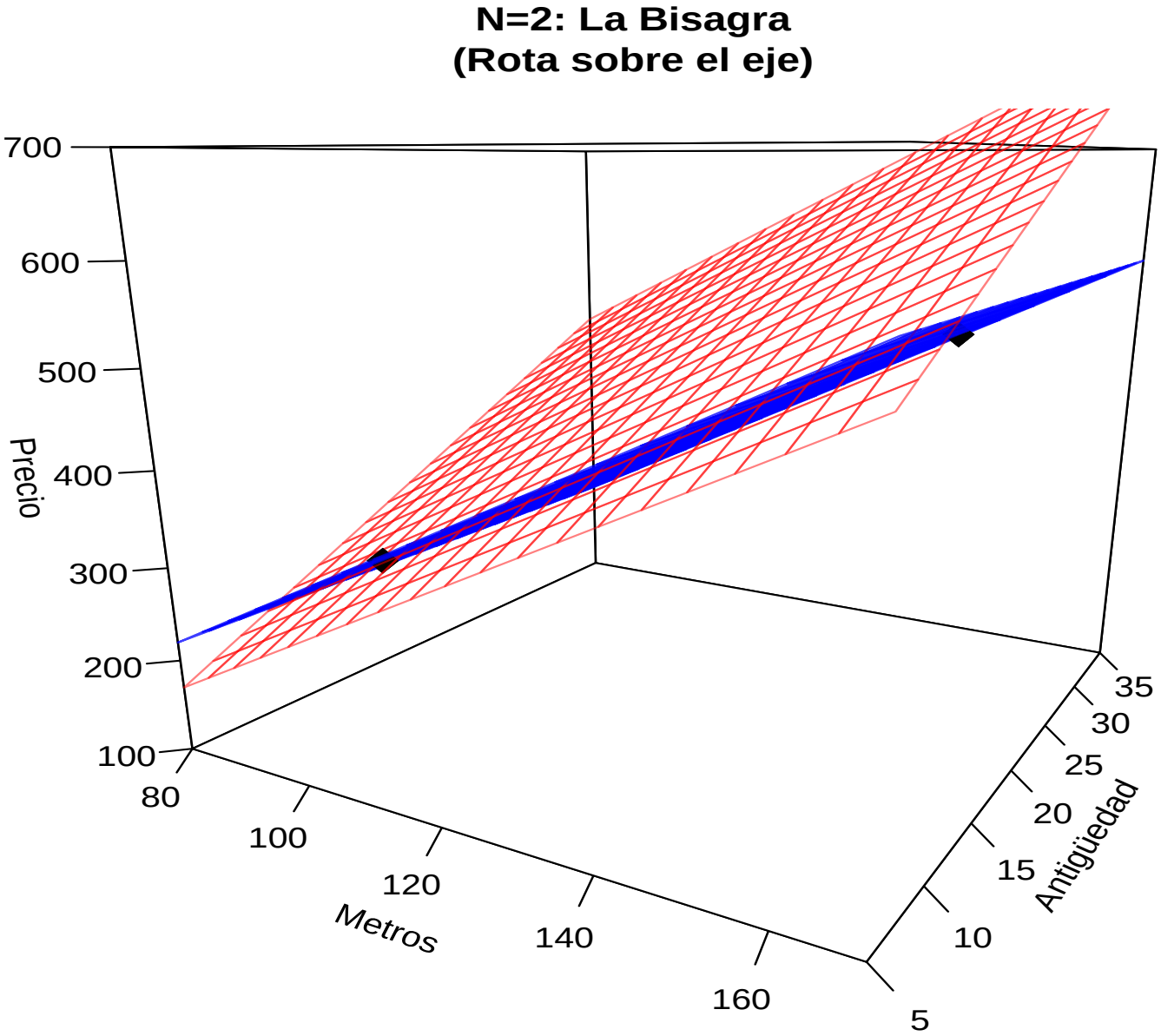
- Si el determinante es 0.00001, $\hat{\beta} = 100,000 adj(X^TX)X'Y$.
- En términos matemáticos, por la definición de la matriz de varianza-covarianza de los coeficientes:

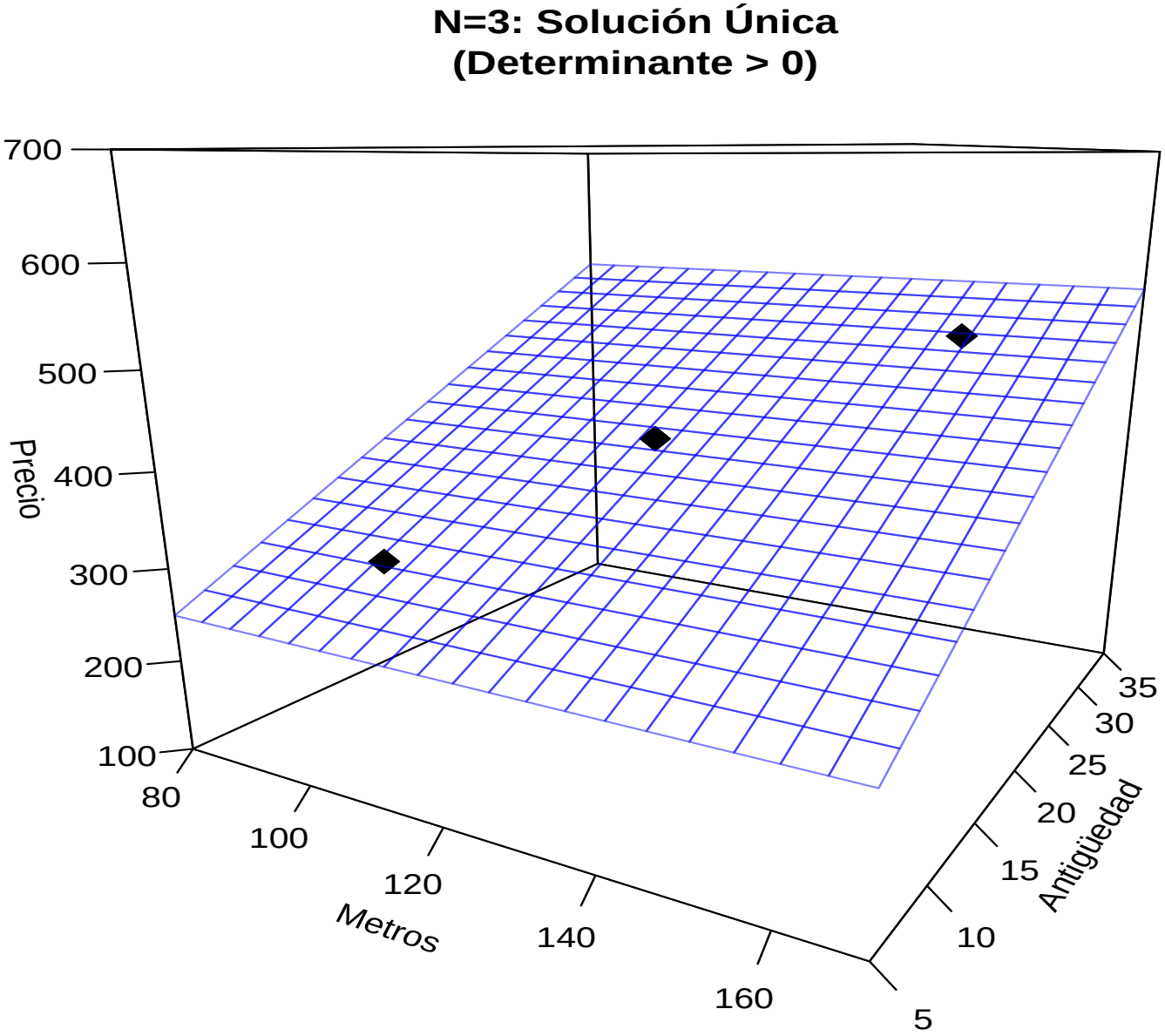
$$Var(\hat{\beta}) = \sigma^2(X^TX)^{-1} = \sigma^2 \frac{1}{det(X^TX)} \cdot adj(X^TX)$$

Donde:σ^2 es la varianza del error (el ruido del modelo). Se tiene una alta varianza en las estimaciones, aun optimizando por MCO.

Problema 3: Alta Dimensionalidad ($p > n$)




N=2: La Bisagra
(Rota sobre el eje)
Precio
700
600
500
400
300
200
100
Metros
80
100
120
140
160
Antigüedad
5
10
15
20
25
30
35



Necesitamos al menos 3 patas de la mesa para que esta se sostenga.

Para ajustar el modelo de RLS con p parámetros, necesitamos al menos p datos. Puede que contemos con miles de variables para unos cuantos sujetos:

Genómica y Bioinformática

- Registros (n): Tienes 50 pacientes (es difícil y caro conseguir más personas con esa condición específica).
- Variables (p): El genoma humano tiene aproximadamente 20,000 genes que se expresan.
- Problema: Tienes 20,000 posibles "explicaciones" para solo 50 resultados.

Marketing Digital de Productos

- Registros (n): Solo 100 usuarios han visitado la sección de relojes hoy.
- Variables (p): Por cada usuario, la plataforma rastrea 5,000 variables: ubicación exacta, dispositivo, historial de clics previos, tiempo de permanencia, edad estimada, ...
- El problema: Hay más "etiquetas" sobre el comportamiento del usuario que usuarios reales.

Diagnóstico por Imágenes Médicas de tumores en resonancias magnéticas

- Registros (n): Un hospital aporta 200 imágenes de resonancia.
- Variables (p): Cada imagen tiene una resolución de 512x512 píxeles.
- El problema: Cada píxel es una variable de entrada. Hay un cuarto de millón de variables para solo 200 casos.

- Con $n < p$, $(X'X)^{-1}$ no puede calcularse, pues su determinante es 0. Es un problema aún peor que el de la multicolinealidad, el la que el determinante es casi 0.
- Aún con $n = p$, el modelo corre el riesgo de "memorizar" a los pacientes/usuarios/... en lugar de aprender un patrón general
- El modelo dice: "No puedo darte una respuesta porque hay infinitas combinaciones de β que explican perfectamente estos pocos datos".

Regresión Ridge

El Problema

El ajuste de mínimos cuadrados da una gran libertad al ajuste buscando el mínimo error de predicción en el conjunto de datos

$$\min_{\beta} \|y - X\beta\|_2^2$$

- En Multicolinealidad o $n < p$: La superficie de error tiene un "valle plano". No hay un único punto mínimo, sino una línea (o hiperplano) de soluciones.
- Consecuencia: El algoritmo elige coeficientes gigantescos ($\beta \to \infty$) que se cancelan entre sí para ajustar el ruido. O inexistentes en $n < p$.

Para evitar que los β crezcan desmesuradamente, les ponemos un "techo":

$$\min_{\beta} \|y - X\beta\|_2^2 \quad \text{s.t.} \quad \sum \beta_j^2 \leq t$$

La Intuición de Lagrange (problema equivalente):

$$\min_{\beta} \underbrace{\|y - X\beta\|_2^2}_{\text{Pérdida (Ajuste)}} + \underbrace{\lambda \|\beta\|_2^2}_{\text{Penalización (Estabilidad)}}$$

- $\lambda = 0$: Volvemos a MCO.
- $\lambda \to \infty$: El modelo se vuelve ultra-conservador ($\beta \to 0$).

Al añadir $\lambda \sum \beta^2$, se está dando al modelo una regla adicional: "Ya que no tiene suficiente información para decidir, elija la combinación que tenga los menores betas posibles".

¿Por qué la Norma L_2?

Efecto Matemático y Computacional

$$\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - x_i' \beta)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

- Estabilización del Determinante: La solución analítica es

$$\hat{\beta}_{Ridge} = (X^T X + \lambda I)^{-1} X^T Y$$

- Sumar λ a la diagonal asegura que la matriz sea siempre invertible (valores propios también se les suma λ).
- Suavizado de la Curvatura: Transforma un valle plano y errático en un "tazón" con un único mínimo global claro.

De $\hat{\beta}_{Ridge}$ a $\hat{\beta}_{MCO}$

Podemos ver como Ridge es un caso general del MCO. Inciando con la factorización de la base:

$$(X^T X + \lambda I) = (X^T X)[I + \lambda (X^T X)^{-1}]$$

Inversión del producto $(A \cdot B)^{-1} = B^{-1} A^{-1}$:

$$(X^T X + \lambda I)^{-1} = [I + \lambda (X^T X)^{-1}]^{-1} (X^T X)^{-1}$$

Sustitución en la ecuación original:

$$\hat{\beta}_{Ridge} = [I + \lambda (X^T X)^{-1}]^{-1} (X^T X)^{-1} X^T Y$$

Sustitución del estimador MCO ($\hat{\beta}_{MCO} = (X^T X)^{-1} X^T Y$):

$$\hat{\beta}_{Ridge} = [I + \lambda (X^T X)^{-1}]^{-1} \hat{\beta}_{MCO} \longrightarrow Sesgo(\hat{\beta}_{Ridge}) = -\lambda (X^T X + \lambda I)^{-1} \beta$$

2. El Dilema Sesgo-Varianza (Bias-Variance Tradeoff)

En una regresión lineal, nuestra predicción es $\hat{f}(x) = x^T \hat{\beta}$. Por lo tanto, el comportamiento del error de predicción depende directamente de las propiedades estadísticas de $\hat{\beta}$. El error de predicción se descompone como:

$$\underbrace{\mathbb{E}[(y - \hat{f}(x))^2]}_{\text{Error Total}} = \underbrace{\text{Bias}^2[\hat{f}(x)]}_{\text{Sesgo}^2} + \underbrace{\text{Var}[\hat{f}(x)]}_{\text{Varianza}} + \underbrace{\sigma^2}_{\text{Ruido irreducible}}$$

El Sesgo en el Error de Predicción

El sesgo del modelo es la diferencia entre el valor esperado de la predicción y la realidad:

$$Bias[\hat{f}(x)] = \mathbb{E}[x^T\hat{\beta}] - x^T\beta = x^T(\mathbb{E}[\hat{\beta}] - \beta)$$

En RLS: Como $\mathbb{E}[\hat{\beta}_{MCO}] = \beta$:

$$Bias[\hat{f}(x)]_{RLS} = 0$$

En Ridge: Como $\mathbb{E}[\hat{\beta}_{Ridge}] = [I + \lambda(X^TX)^{-1}]^{-1}\beta$:

$$Bias[\hat{f}(x)]_{Ridge} = x^T\left([I + \lambda(X^TX)^{-1}]^{-1} - I\right)\beta \neq 0$$

Conclusión: Ridge aumenta el primer término de la ecuación ($Bias^2$).

La Varianza en el Error de Predicción

La varianza del modelo mide cuánto cambia la predicción si cambiamos los datos de entrenamiento:

$$Var[\hat{f}(x)] = Var(x^T\hat{\beta}) = x^T Var(\hat{\beta})x$$

En RLS:

$$Var[\hat{f}(x)]_{RLS} = \sigma^2 \underbrace{x^T(X^TX)^{-1}x}_{\text{Explota si } \det \to 0}$$

En Ridge:

$$Var[\hat{f}(x)]_{Ridge} = \sigma^2 \underbrace{x^T(X^TX+\lambda I)^{-1}X^TX(X^TX+\lambda I)^{-1}x}_{\text{Estable por } \lambda}$$

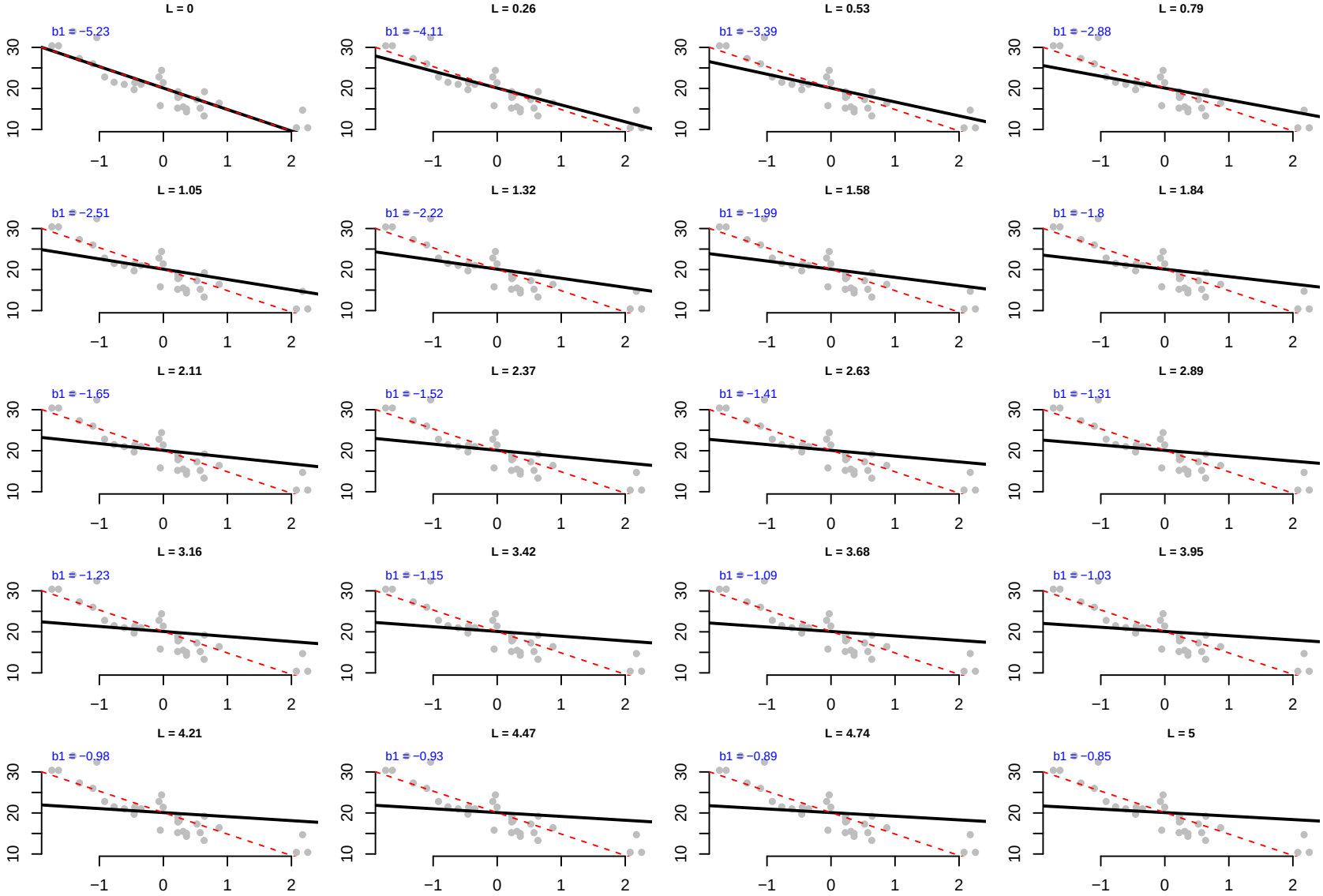
Conclusión: Ridge reduce drásticamente el segundo término de la ecuación (Var).

Si sumamos todo en tu fórmula de Error Total:

$$MSE = \underbrace{\|x^T(Bias(\hat{\beta}))\|^2}_{\text{Crece con } \lambda} + \underbrace{\sigma^2 x^T Var(\hat{\beta})x}_{\text{Decrece con } \lambda} + \underbrace{\sigma^2}_{\text{Constante}}$$

- Qué tanto sesgo es necesario?. Aquél que resulte en una menor varianza.

Efecto de Encogimiento (Shrinkage) en Ridge



La importancia de la estandarización

La penalización Ridge ($\lambda \sum \beta_j^2$) no considera a las unidades de medida.

```
# 1. Creamos dos variables con la misma información pero distinta escala
y  <- mtcars$mpg
peso_ton  <- mtcars$wt           # Escala original (aprox 2 a 5)
peso_gramos <- mtcars$wt * 10^6  # Escala gigante (millones)

# 2. Función Ridge
ridge_analitico <- function(X, y, lambda) {
  # IMPORTANTE: No penalizamos la primera columna (intercepto)
  I <- diag(ncol(X))
  I[1,1] <- 0
  solve(t(X) %*% X + lambda * I) %*% t(X) %*% y
}

# 3. Preparamos las matrices de diseño
X_ton <- cbind(1, peso_ton)
X_gr  <- cbind(1, peso_gramos)

# 4. Aplicamos el mismo Lambda a ambos
l_val <- 10

b_ton <- ridge_analitico(X_ton, y, lambda = l_val)
b_gr  <- ridge_analitico(X_gr, y, lambda = l_val)

# 5. RESULTADOS
cat("Pendiente con Toneladas:", round(b_ton[2], 6), "\n")
```

```
## Pendiente con Toneladas: -3.997536
```

```
cat("Pendiente con Gramos:   ", round(b_gr[2], 6), "\n")
```

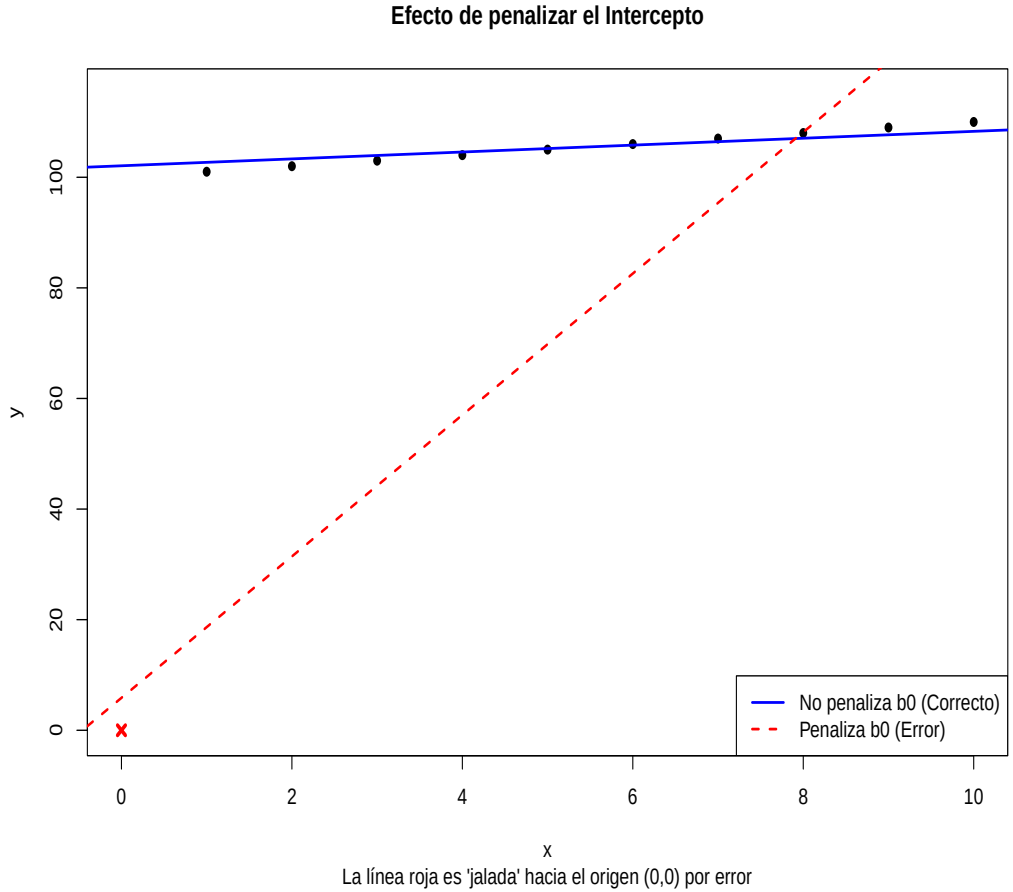
```
## Pendiente con Gramos:    -5e-06
```

Ridge "castigará" mucho más fuerte a la variable que tenga la escala más pequeña (porque su β es naturalmente más grande)

La importancia de no penalizar el intercepto

```r
# 1. Datos simples: Una relación lineal clara pero desplazada del origen
x <- 1:10
y <- x + 100  # El intercepto real es 100

# 2. Función Ridge que permite elegir si penalizar o no el intercepto
ridge_demo <- function(x, y, lambda, penalizar_intercepto = FALSE) {
  X <- cbind(1, x)
  I <- diag(2)
  if (!penalizar_intercepto) I[1,1] <- 0 # Quitamos penalización al intercepto

  # Solución analítica: (XtX + lambda*I)^-1 %*% XtY
  beta <- solve(t(X) %*% X + lambda * I) %*% t(X) %*% y
  return(beta)
}

# 3. Probamos con un lambda fuerte
L <- 50

beta_ok  <- ridge_demo(x, y, L, penalizar_intercepto = FALSE)
beta_bad <- ridge_demo(x, y, L, penalizar_intercepto = TRUE)
```

El intercepto es un parámetro de posición, no de complejidad: penalizarlo sesga la ubicación del modelo sin reducir su varianza



La regularización busca controlar la fuerza de las variables (pendientes), no el nivel donde flotan los datos (intercepto).

Uso básico de glmnet

```
# 1. Preparación de datos (Matriz X y vector y)
X <- as.matrix(mtcars[, c("wt", "disp")])
y <- mtcars$mpg

# 2. Ajuste del modelo Ridge (alpha = 0)
fit_ridge <- glmnet(X, y, alpha = 0)

# 3. Comparación: OLS vs Ridge (con lambda casi 0)
fit_ols <- lm(y ~ wt + disp, data = mtcars)

cat("Coeficientes OLS originales:\n")
```

```
## Coeficientes OLS originales:
```

```
print(round(coef(fit_ols), 3))
```

```
## (Intercept)          wt        disp
##      34.961      -3.351      -0.018
```

```
cat("Coeficientes Ridge con lambda muy pequeño")
```

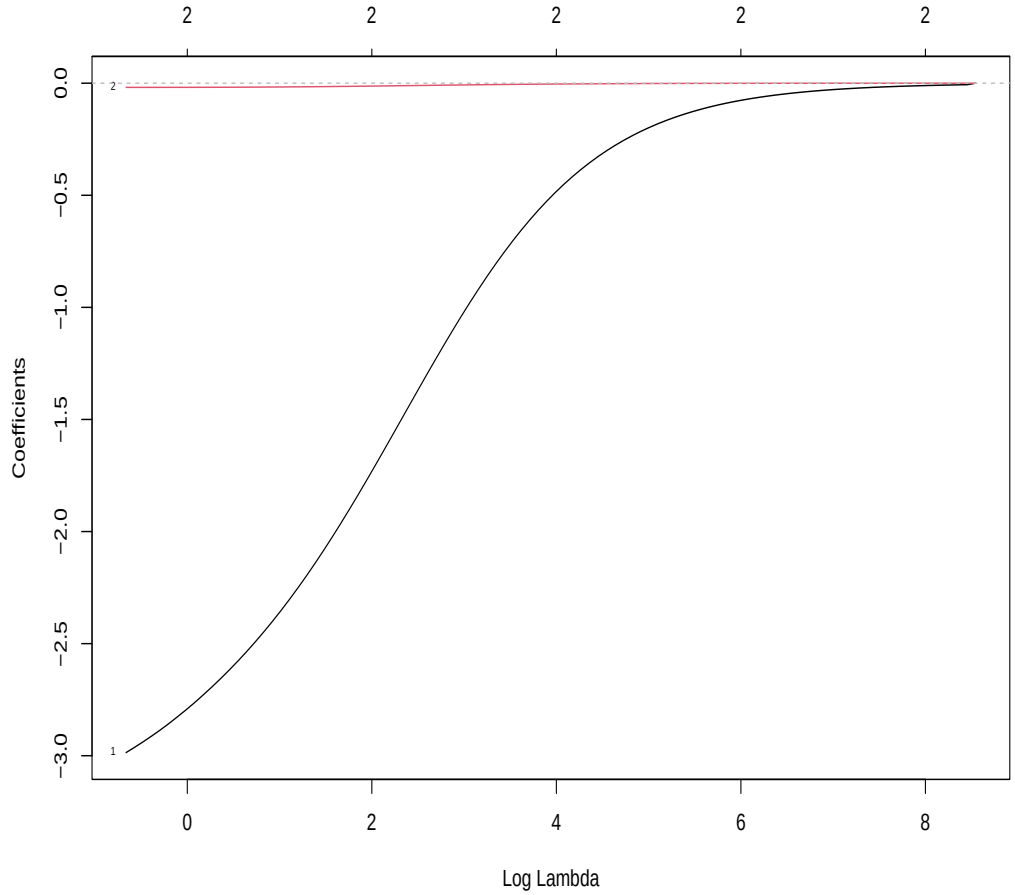
```
## Coeficientes Ridge con lambda muy pequeño
```

```
print(round(as.matrix(coef(fit_ridge, s = 0.01)), 3))
```

```
##                  s1
## (Intercept) 34.003
## wt          -2.986
## disp        -0.019
```

En la traza de glmnet, el eje X suele ser el $log(\lambda)$.

- A la izquierda (log-lambda pequeño), los coeficientes son libres: MCO.
- A la derecha (log-lambda grande), Los coeficientes se encogen.



glmnet escala internamente las variables ($media = 0, sd = 1$)

Regresión LASSO

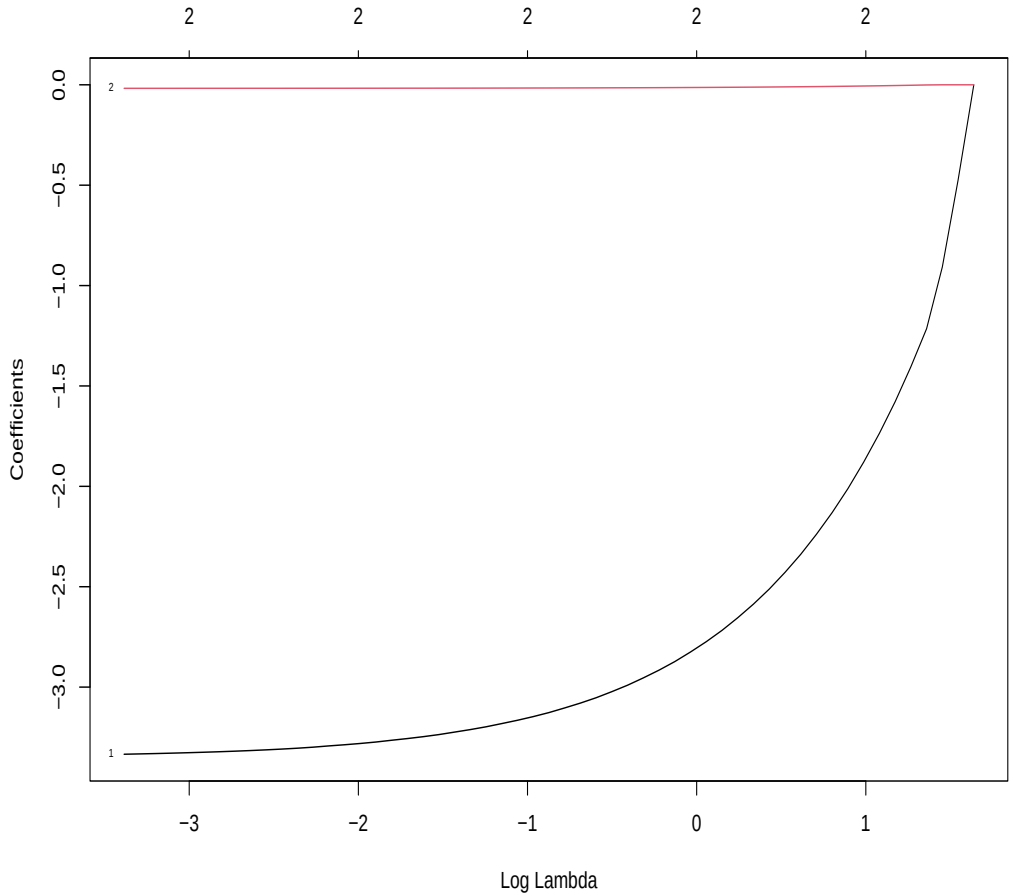
Regresión LASSO

(Norma L_1) Si Ridge es un ancla que estabiliza los coeficientes, LASSO es un filtro que simplifica el modelo eliminando variables irrelevantes.

$$\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - x_i' \beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

A diferencia de Ridge, la geometría de LASSO tiene "esquinas". Esto produce una riqueza de ceros: Mientras que Ridge fuerza a los β a ser pequeños, LASSO los obliga a ser cero.

```r
# LASSO (alpha = 1)
fit_lasso <- glmnet(X, y, alpha = 1)

# Visualiza la diferencia?
plot(fit_lasso, xvar = "lambda", label = TRUE)
```



¿Cuándo usar cuál?

¿Cuándo usar cuál?

- Usa Ridge: Si crees que todas tus variables tienen algún efecto (aunque sea pequeño) y solo quieres manejar la multicolinealidad.
- Usa LASSO: Si tienes muchísimas variables ($n < p$) y sospechas que solo unas pocas son realmente importantes. LASSO te entrega un modelo más simple y fácil de interpretar.