



Aprendizaje Automático Profundo

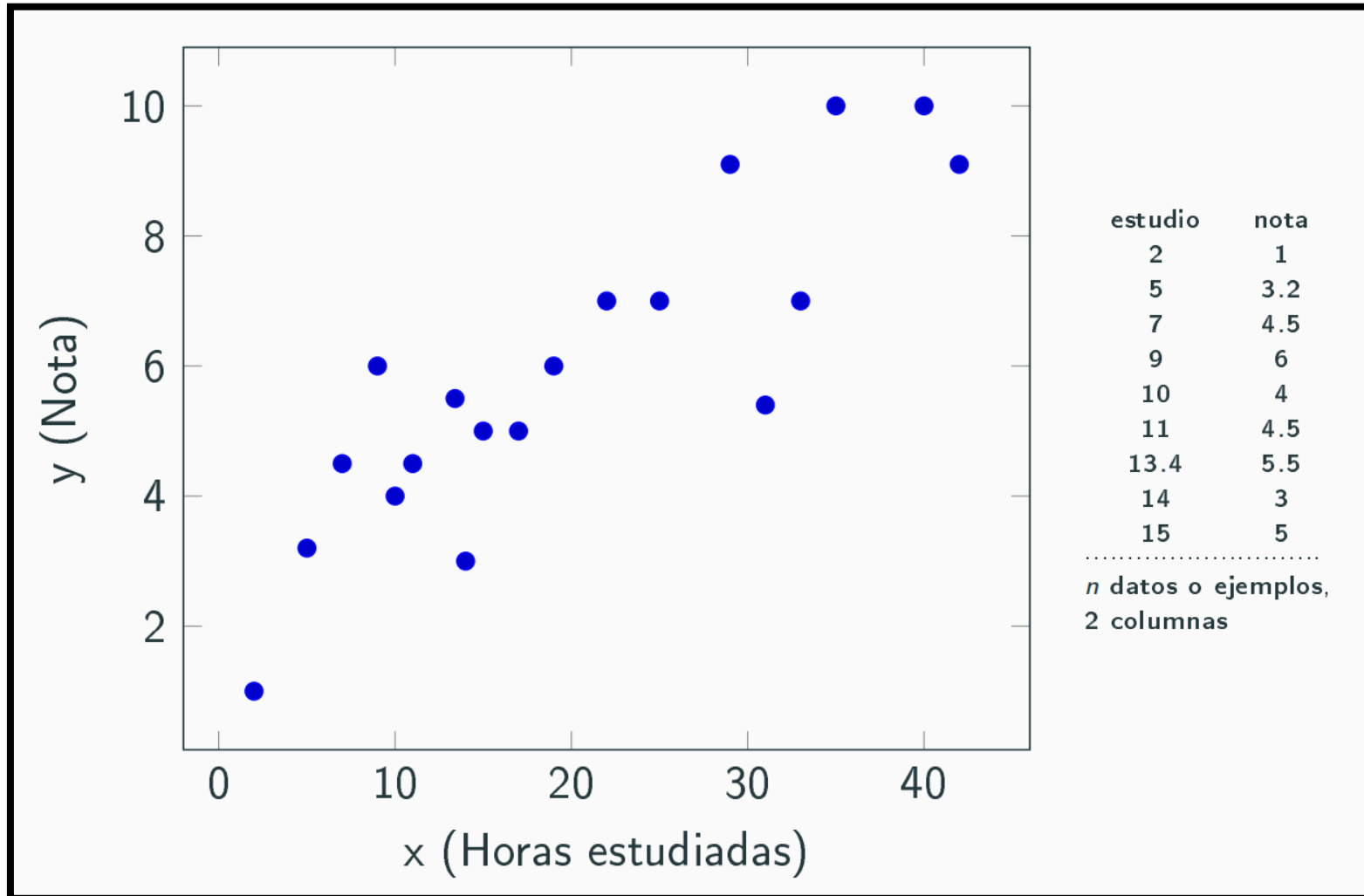
Clase 2 - 2019

Profs: Franco Ronchetti - Facundo Quiroga



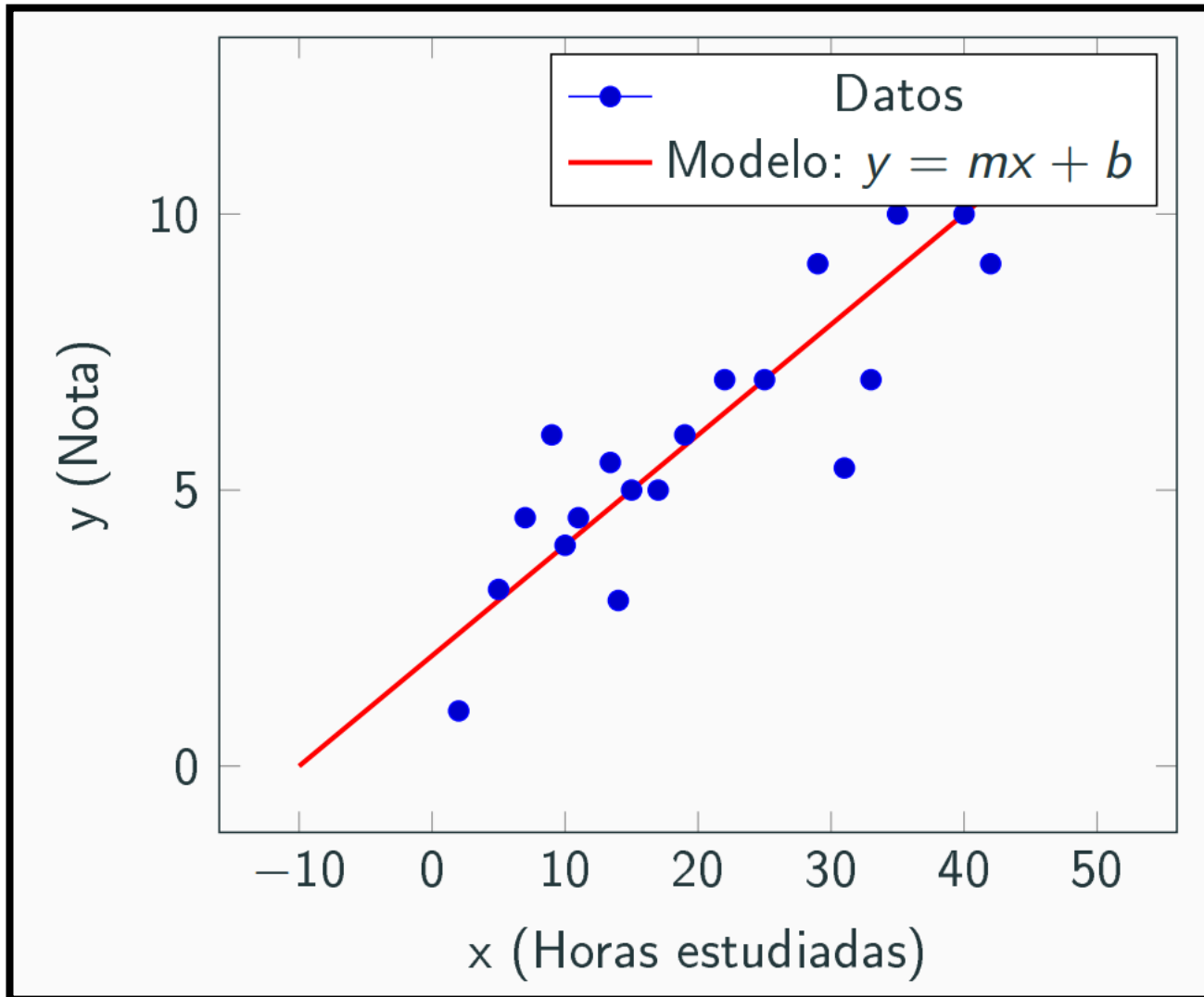
REGRESIÓN LINEAL

Regresión Lineal



Si un nuevo alumno
estudió $x = 20hs$,
¿cuál será su nota?

Regresión Lineal



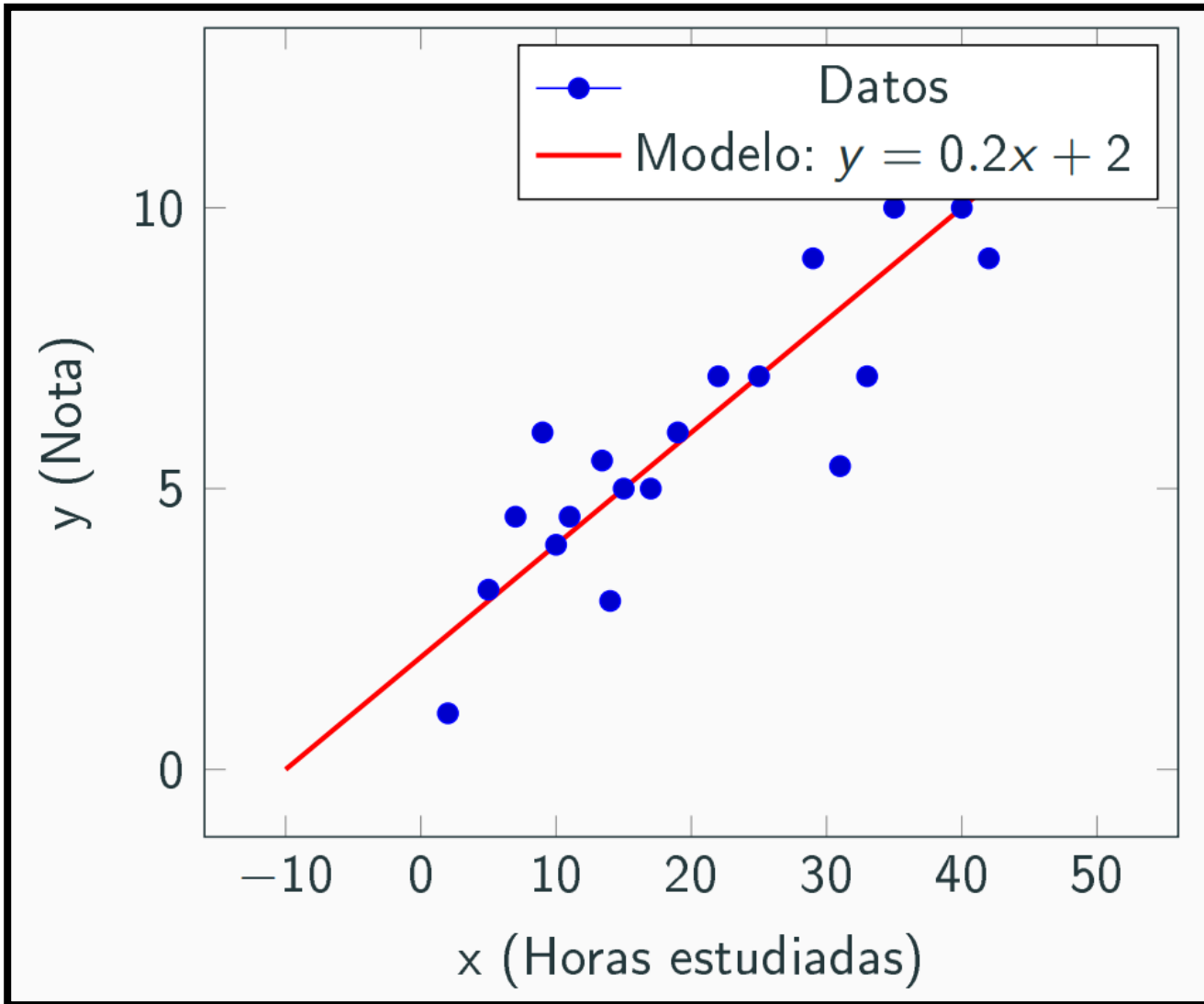
Asumimos que hay una relación lineal entre x e y :

$$y = mX + b$$

Sólo necesitaríamos calcular los parámetros m y b

Algunos autores llama **Hipótesis** a la función que se busca estimar, para cualquier técnica del Aprendizaje Automático.

Regresión Lineal. ¿Cómo predecir?



Suponiendo

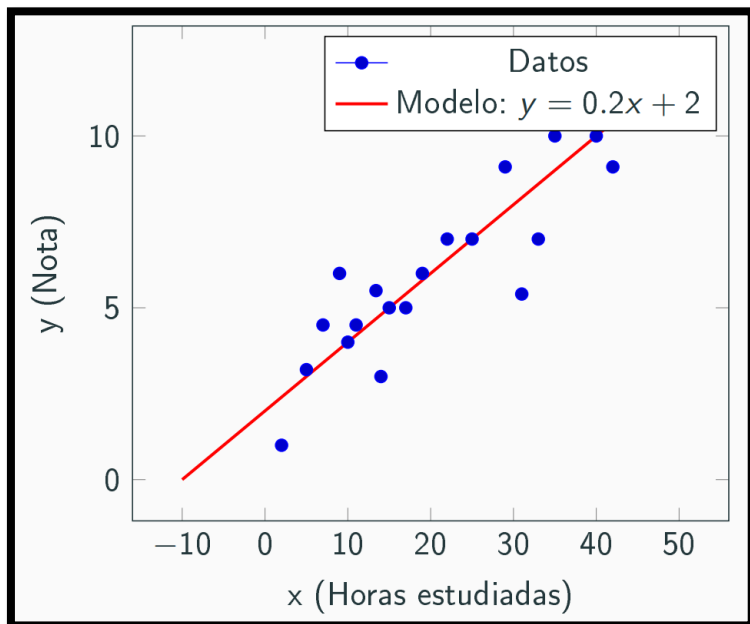
$$y = 0,2x + 2$$

¿Qué nota predice el modelo para $x=20$?

$$f(20) = m20 + b$$

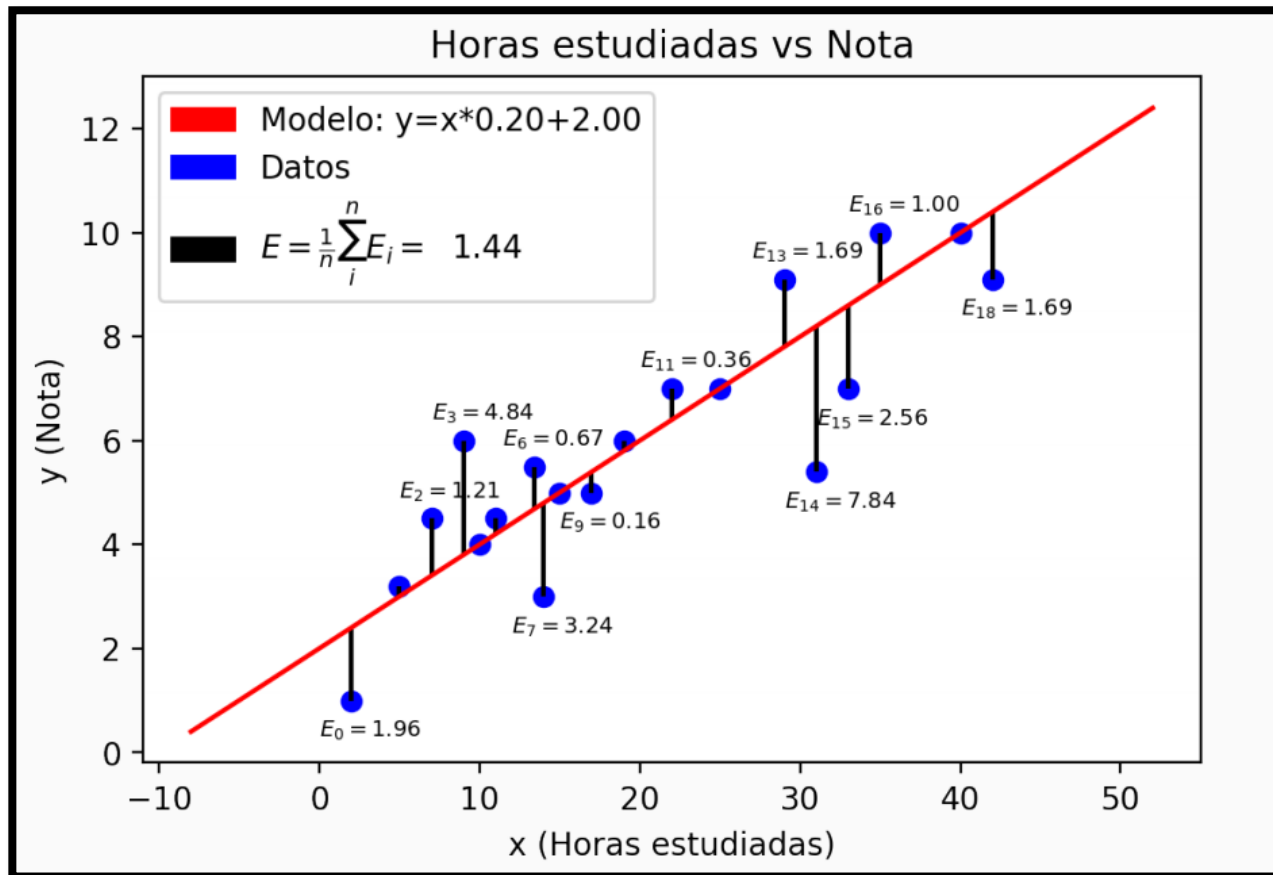
$$= 0,2 \times 20 + 2 = 6$$

¿Por qué un modelo lineal?



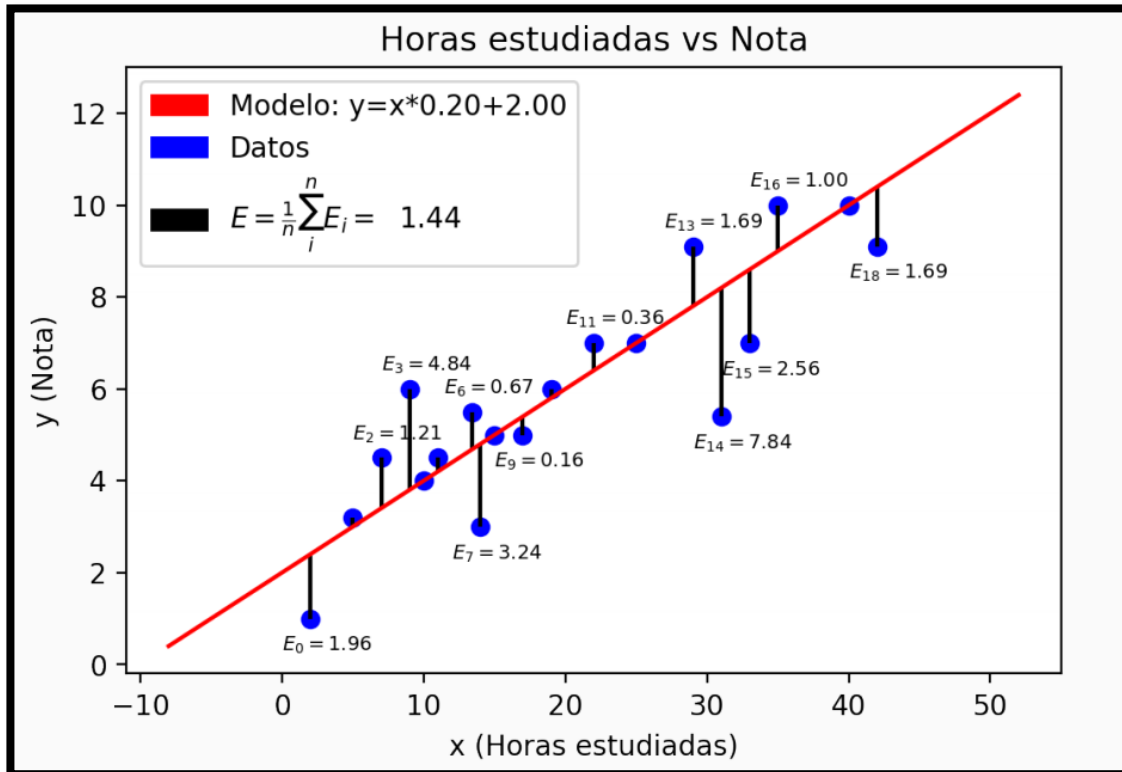
- Es el modelo más simple (polinomio de grado 1)
- Una recta nunca va a poder pasar por todos los puntos. Cada punto (x, y) se aproxima con cierto error.
- Necesitamos una medida de error para aprender los parámetros m y b con el menor error posible.
- A medida que aumenta la dimensionalidad, mejora la eficacia.

Error del modelo (m, b)



- Si los datos no pertenecen a la recta, entonces habrá errores que llamaremos E_i .
- Necesitamos una medida de error $E = E(m, b, x, y)$

Función de costo del modelo: Error cuadrático medio



$$E = \frac{1}{n} \sum_i^n (y'_i - y_i)^2$$

Donde:

(x_i, y_i) = el *i*ésimo elemento de la base de datos

y'_i = el resultado de mi modelo (m, b)

y_i = el valor esperado (real) para x.

Nuestro objetivo será minimizar este error

Regresión lineal. Aprendizaje

“Aprender” significa encontrar los valores óptimos para m y b en base a los datos que tenemos. También se suele usar el término “Entrenamiento”.

Datos	
estudio	nota
2	1
5	3.2
7	4.5
9	6
10	4

**Aprendizaje
(optimización)**

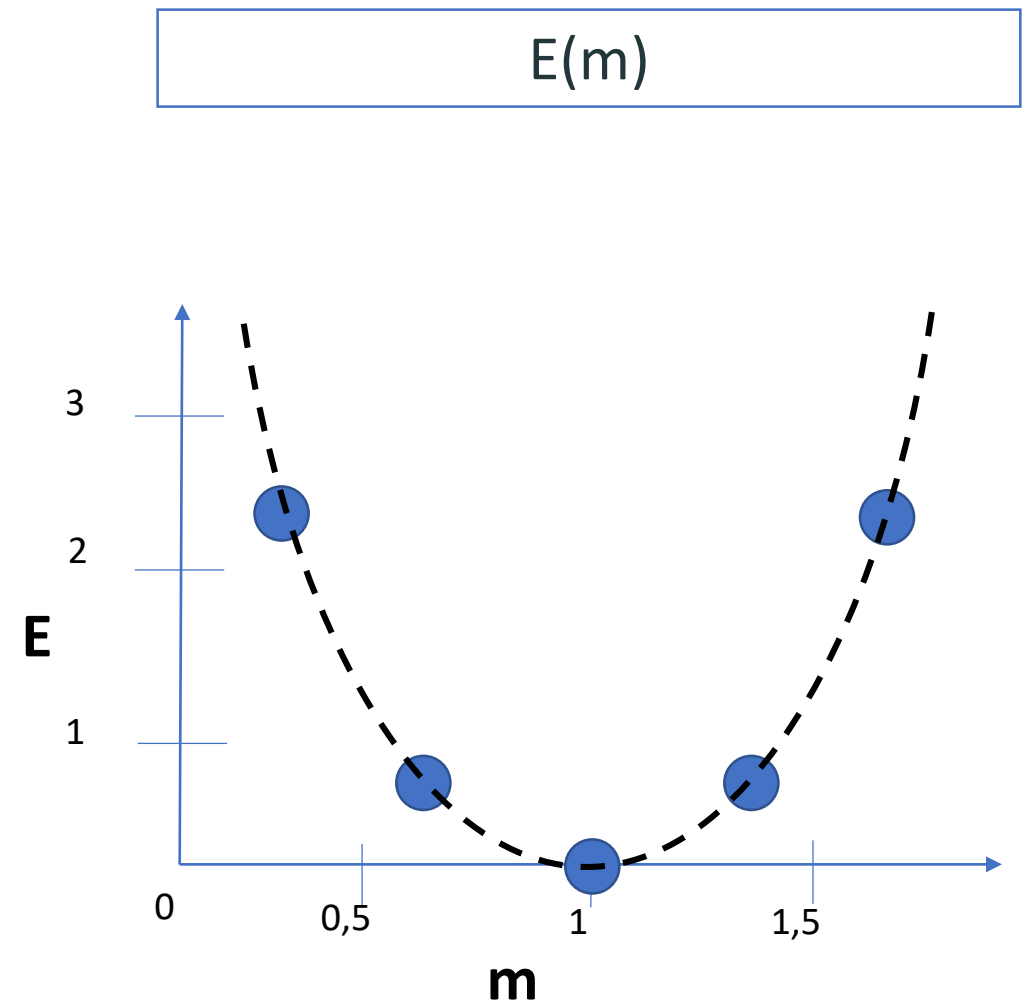
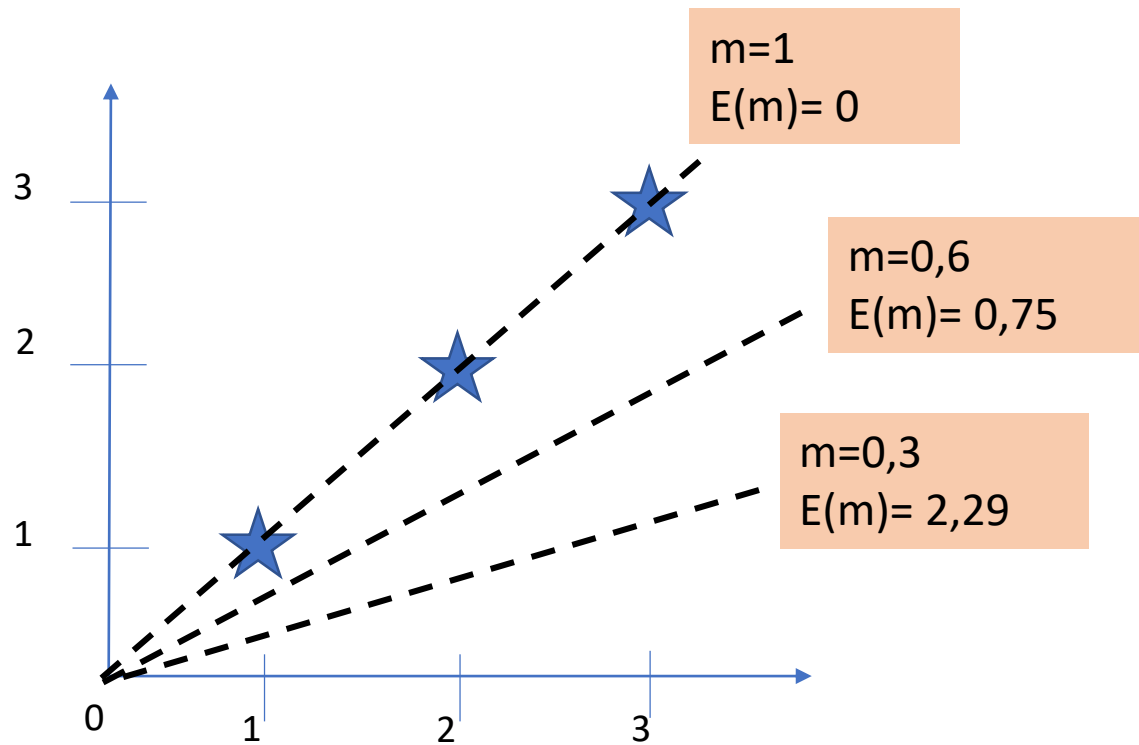
Modelo generado
(puede ser óptimo o no)

m y b

Aprendizaje Supervisado

Regresión lineal. Intuición de función de error

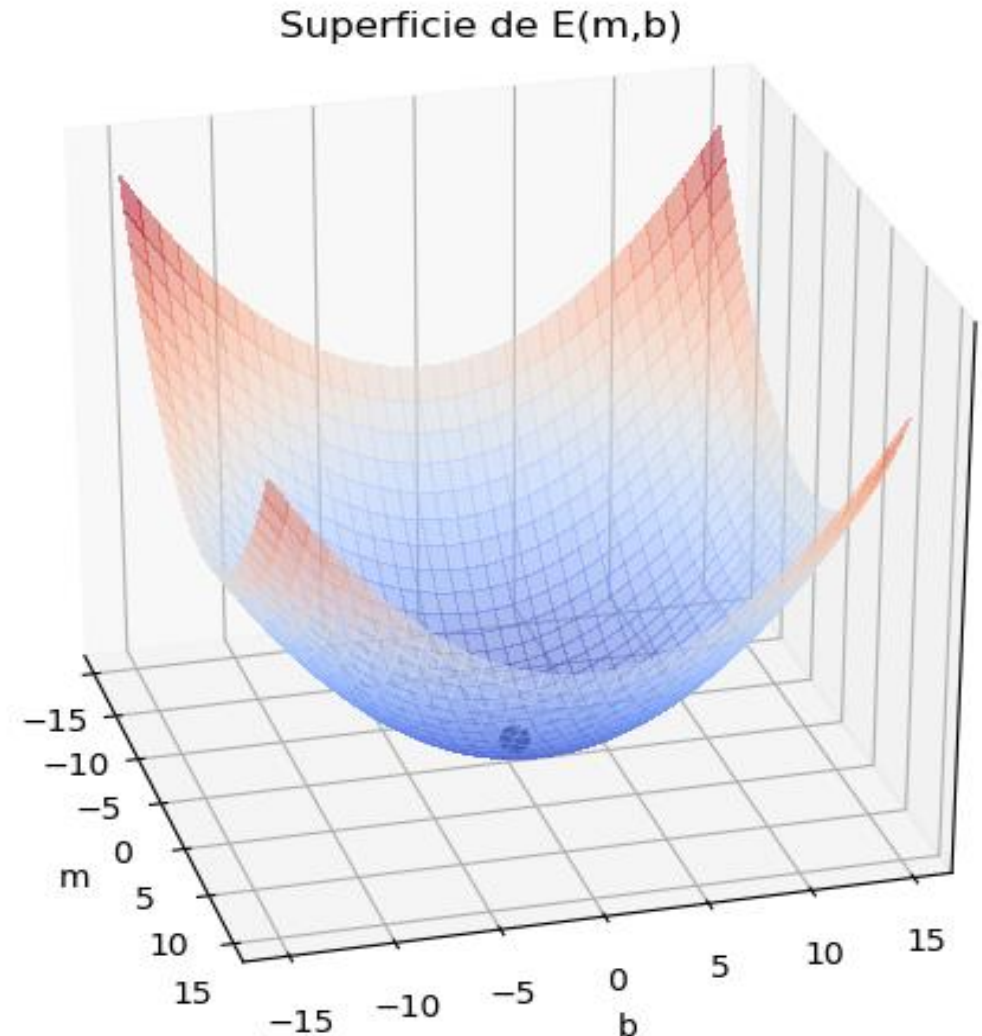
Supongamos esta distribución de punto, y supongamos $b=0$



Regresión lineal. Minimización del error

$E(m,b)$ siempre es convexa.
Posee un solo mínimo local, que es el mínimo global

Esto quiero decir que podemos realizar un método iterativo buscando el óptimo: **descenso del gradiente**.



Regresión lineal. Minimización del error

Objetivo:

encontrar m y b tal que $E(m,b)$ sea mínimo

Métodos clásicos (analíticos)

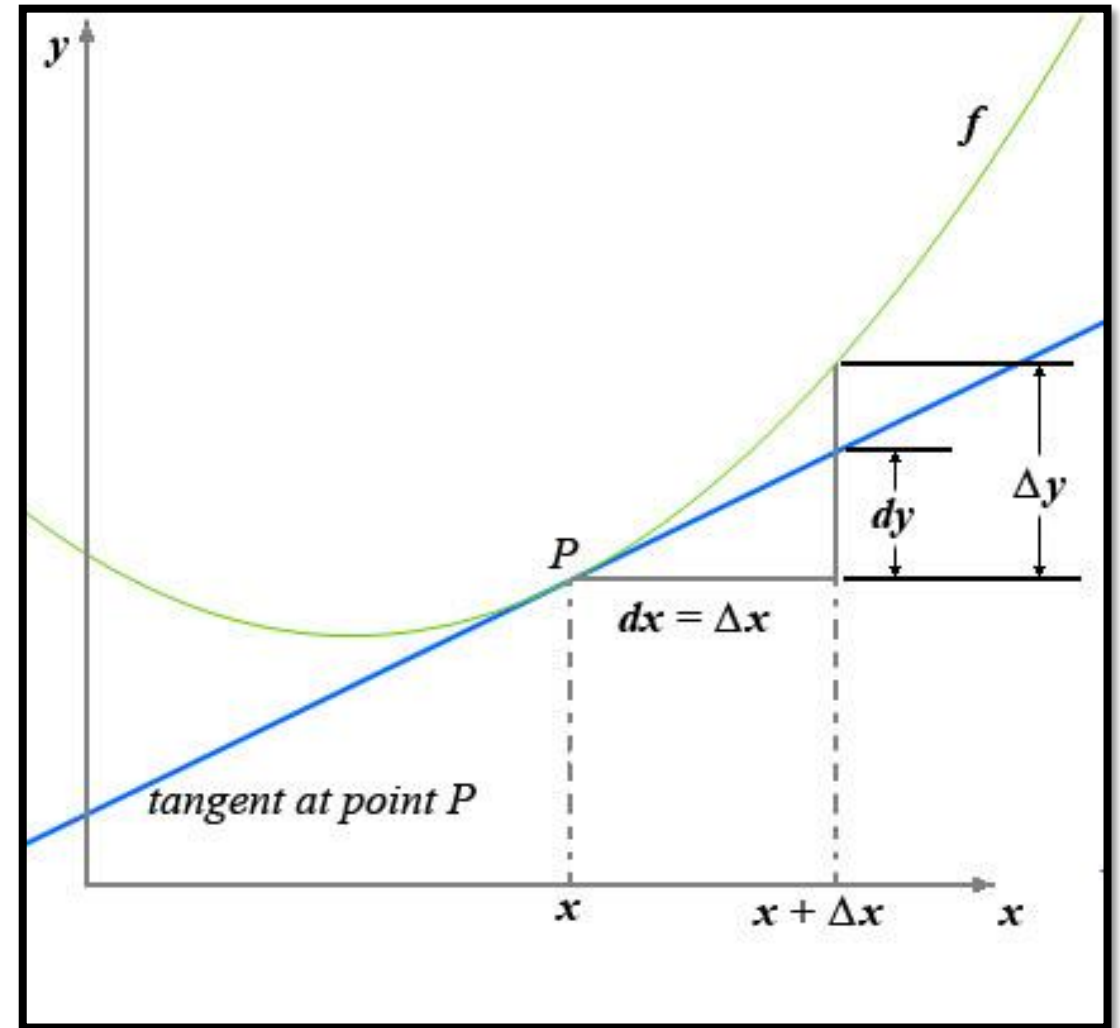
- Cálculo: $\frac{\partial E}{\partial b} = 0$ y $\frac{\partial E}{\partial m} = 0$, despejo m y b .
- Álgebra lineal: $Y = mX + b = \begin{pmatrix} 1 & x \end{pmatrix} \begin{pmatrix} b \\ m \end{pmatrix}$, proyecto.
- Probabilidades: $y = mx + b + e$ con $e \sim \mathcal{N}(0, \sigma)$, estimo m y b con MLE

Descenso del gradiente

- Iterativo
- Generalizable (se puede utilizar para otros modelos como Redes Neuronales, Máquinas de Vectores Soporte, etc.)
- Es más rápido cuando hay muchos datos

Derivada. Repaso.

La derivada de la función en un punto es equivalente a la pendiente de la recta tangente. Es decir, dy/dx .

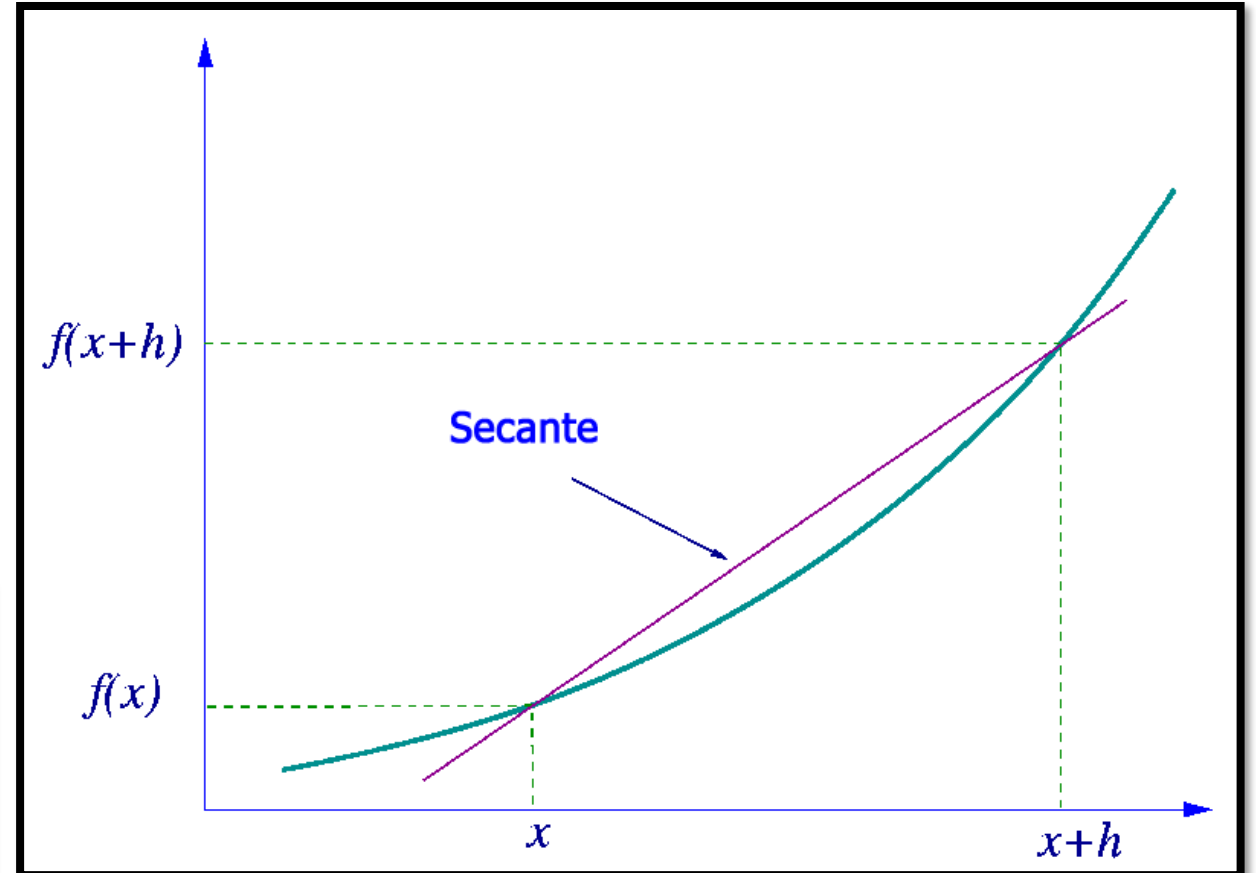


Derivada. Repaso.

Suponiendo un h pequeño, la derivada de f en x es el límite del valor del cociente diferencial, conforme las líneas secantes se aproximan a la línea tangente.

Es decir, el límite de la pendiente de la recta secante.

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} = \begin{cases} > 0 & \text{si } f \text{ crece} \\ < 0 & \text{si } f \text{ decrece} \\ = 0 & \text{pto crítico} \end{cases}$$

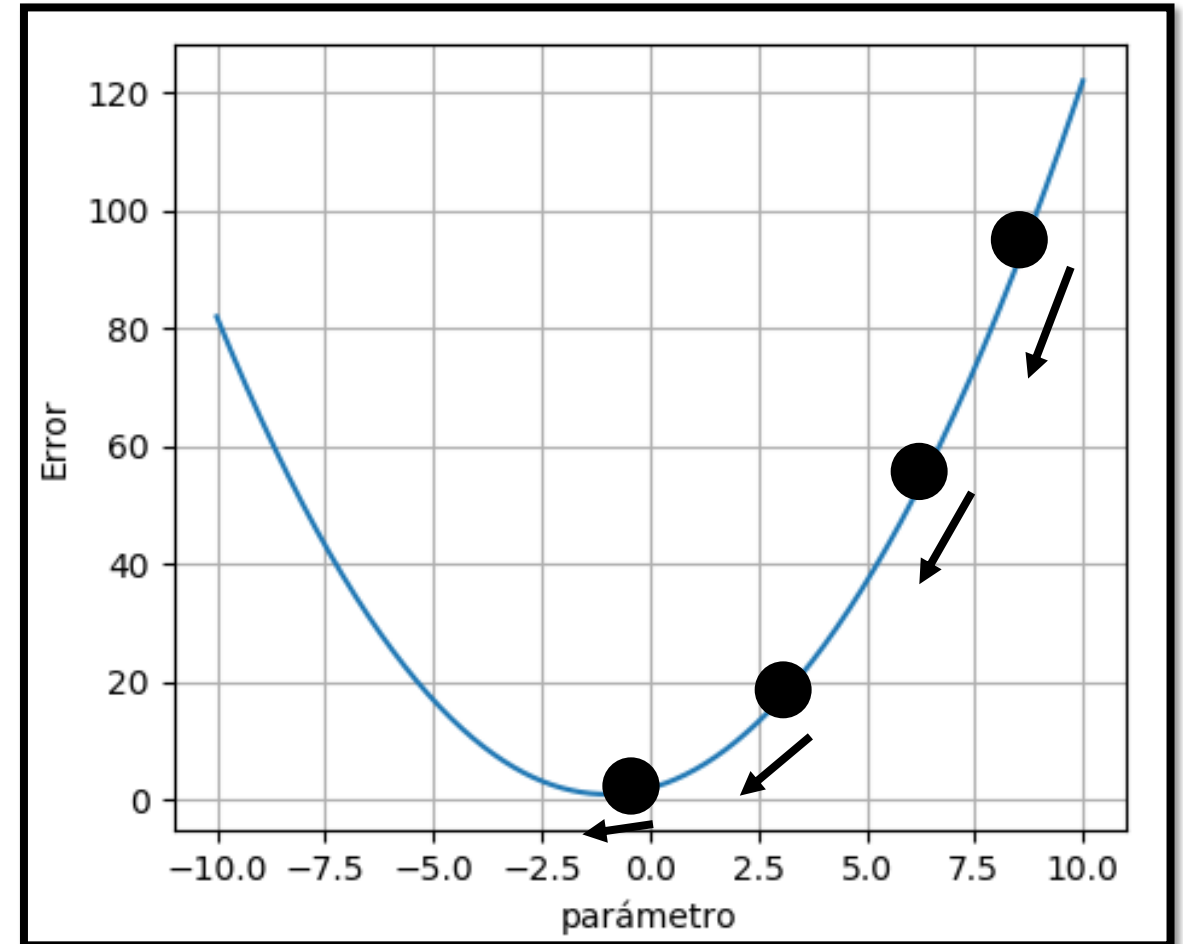


Descenso de gradiente en 1D

Hasta converger:

$$\theta = \theta - \alpha \frac{\delta}{\delta \theta} E(\theta)$$

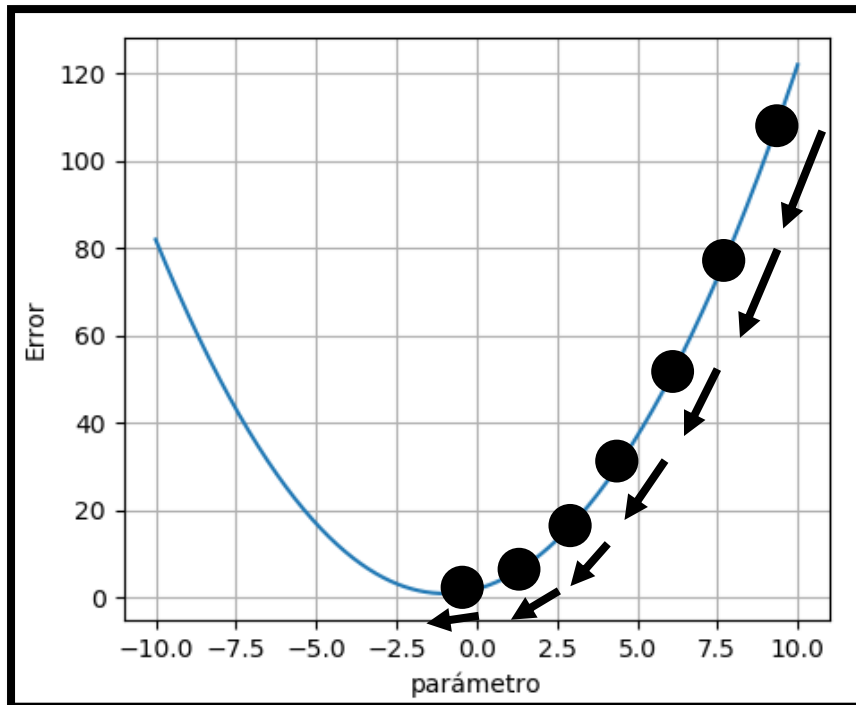
Iterativamente, calcular la derivada del **Error** y recalcular el parámetro θ con un factor de ajuste α .



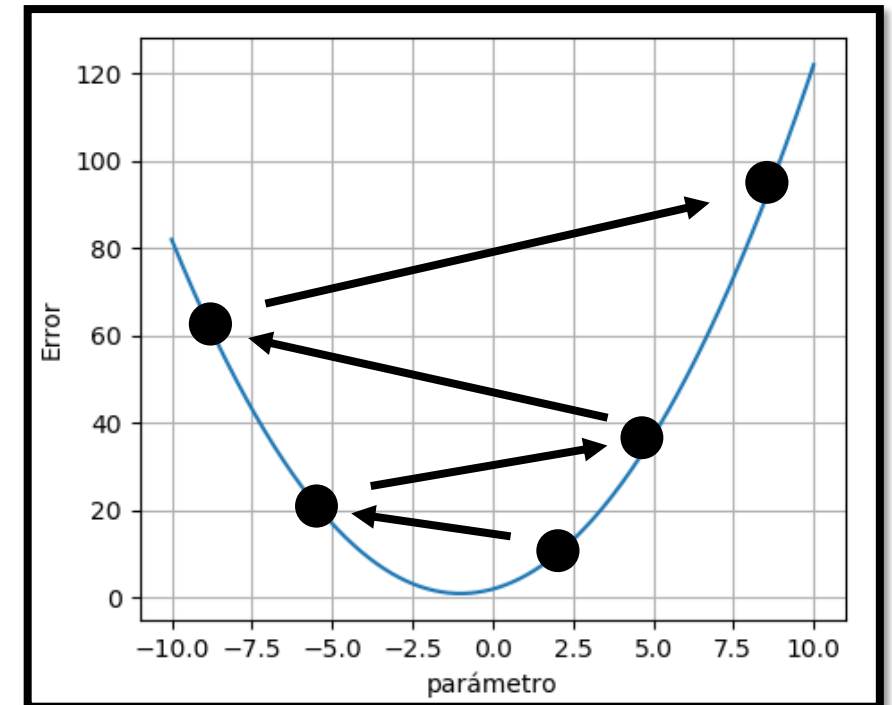
Descenso de gradiente en 1D. Alfa

Hasta converger: $w = w - \alpha \frac{\delta}{\delta w} E(w)$

Si α es muy pequeño, el descenso puede ser muy lento (y no esquivar mínimos locales en caso de otras funciones).



Si α es muy grande, el descenso puede “saltar” el mínimo y nunca converger.



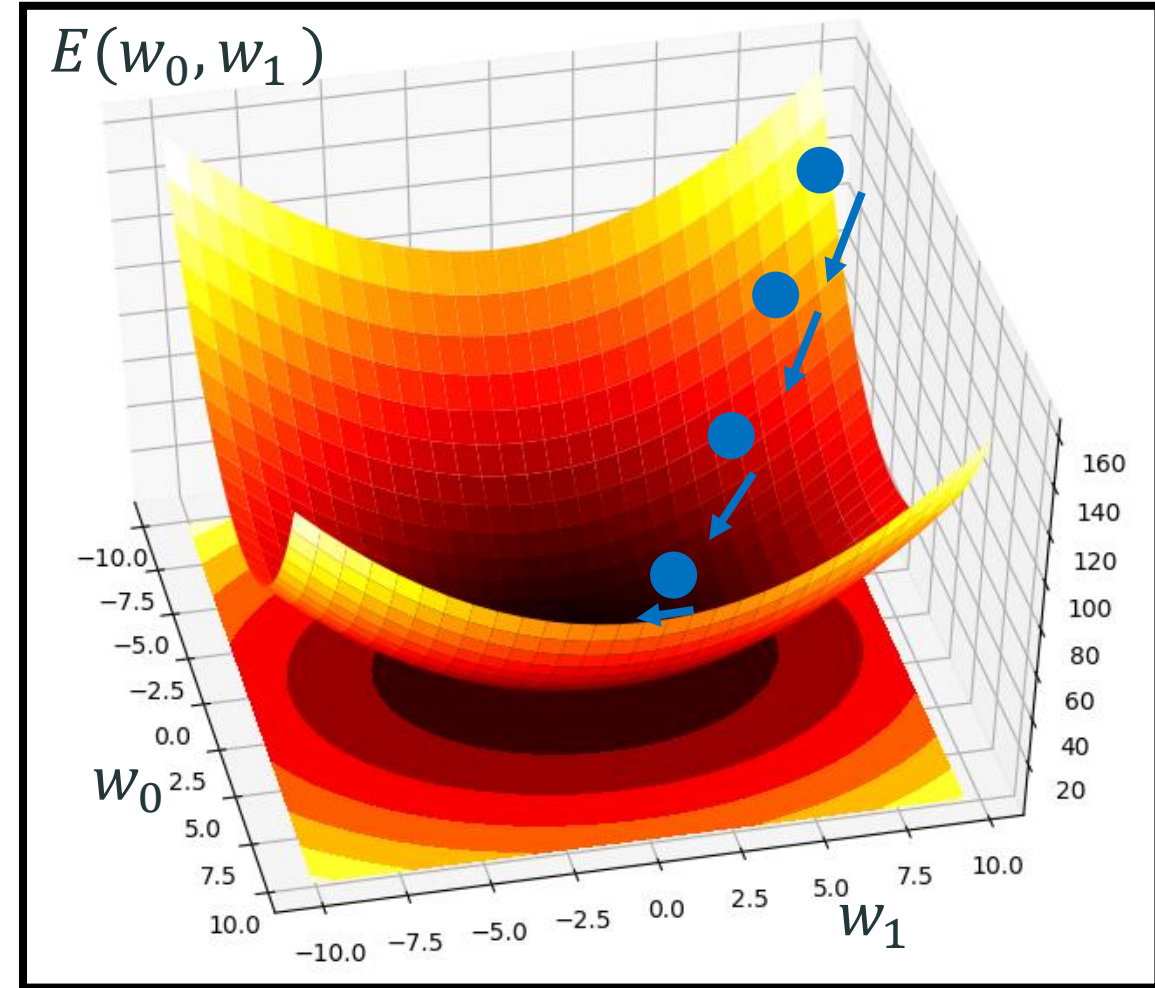
Descenso de gradiente en 2D

Hasta converger:

$$w_i = w_i - \alpha \frac{\delta}{\delta w_i} E(w_0, w_1)$$

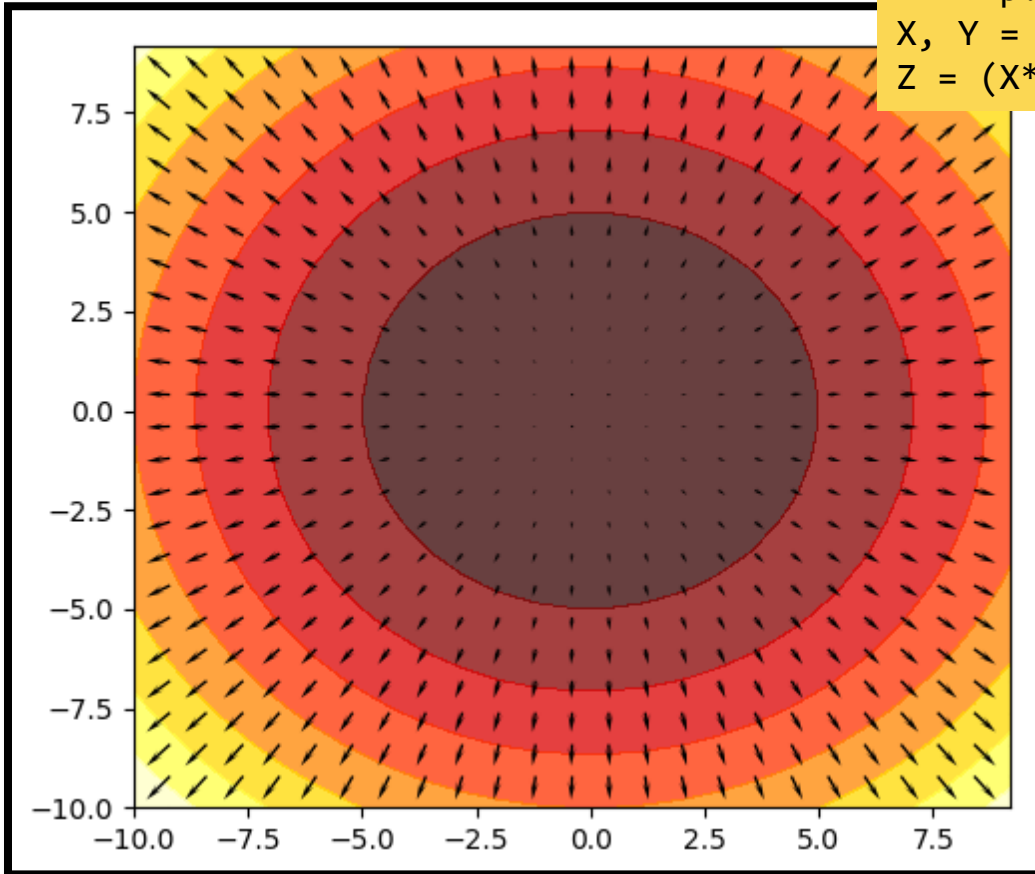
para $i = \{0,1\}$

Iterativamente, calcular la derivada del **Error** con respecto a ambos parámetros y recalcularlos al mismo tiempo.

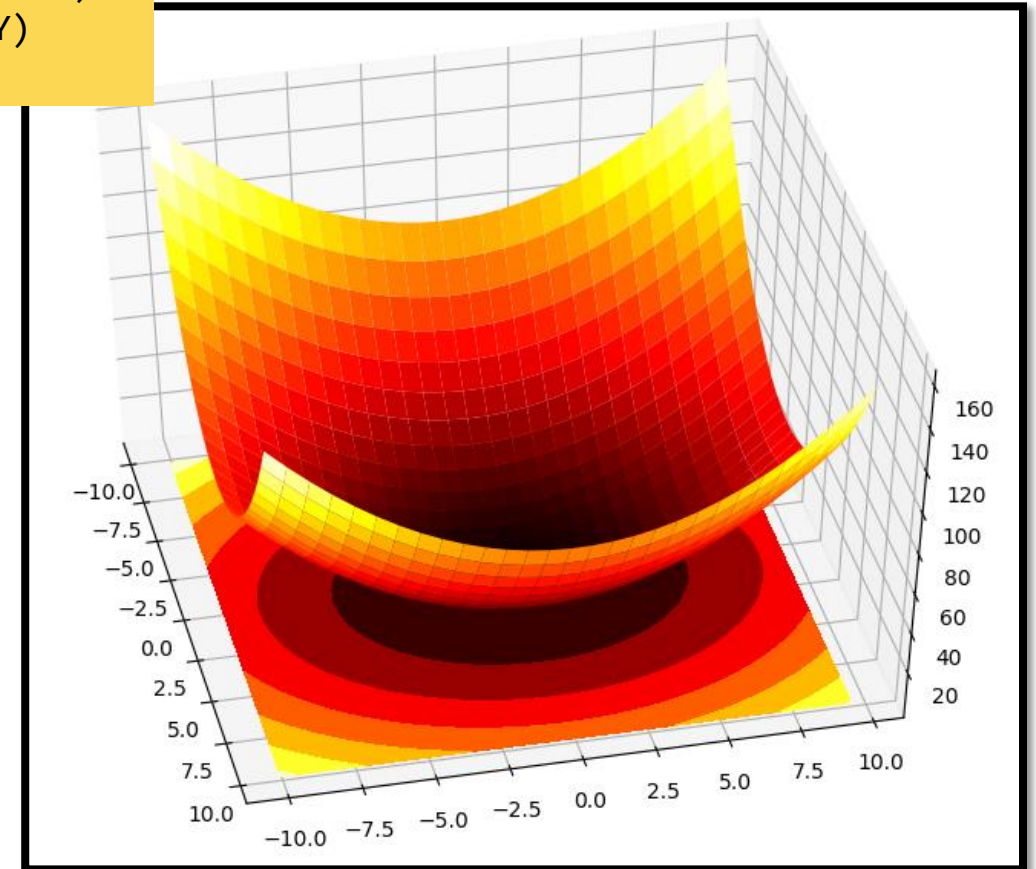


Descenso de gradiente en 2D. Figuras

```
X = np.arange(-10, 10, 0.8)
Y = np.arange(-10, 10, 0.8)
X, Y = np.meshgrid(X, Y)
Z = (X**2 + Y**2)
```



```
plt.contour(X, Y, Z)
dhdY, dhdX = np.gradient(Z)
plt.quiver(X, Y, dhdX, dhdY)
```



```
ax = Axes3D(fig)
ax.plot_surface(X, Y, Z)
ax.contourf(X, Y, Z)
```

Descenso de gradiente para Regresión Lineal

Entrenamiento:

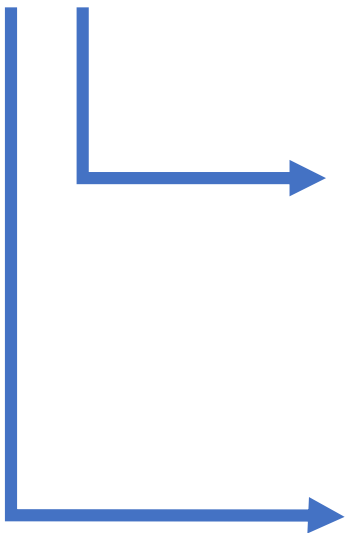
$$w_i = w_i - \alpha \frac{\delta}{\delta w_i} E(w_0, w_1)$$

para $i = \{0,1\}$

Nuestro modelo de Regresión Lineal:

$$y = mx + b$$

$$E = \frac{1}{n} \sum_i^n (y'_i - y_i)^2$$


$$\frac{\delta E}{\delta m} E(m, b) = \frac{2}{n} \sum_i^n (y'_i - y_i) x_i$$

$$\frac{\delta E}{\delta b} E(m, b) = \frac{2}{n} \sum_i^n (y'_i - y_i)$$

Descenso de gradiente para Regresión Lineal

Derivadas parciales del Error

$$\frac{\partial E}{\partial m} E(m, b) = \frac{2}{n} \sum_i^n (y'_i - y_i) x_i$$

$$\frac{\partial E}{\partial b} E(m, b) = \frac{2}{n} \sum_i^n (y'_i - y_i)$$



Entrenamiento:

$$m = m - \alpha \frac{2}{n} \sum_i^n (y'_i - y_i) x_i$$

$$b = b - \alpha \frac{2}{n} \sum_i^n (y'_i - y_i)$$

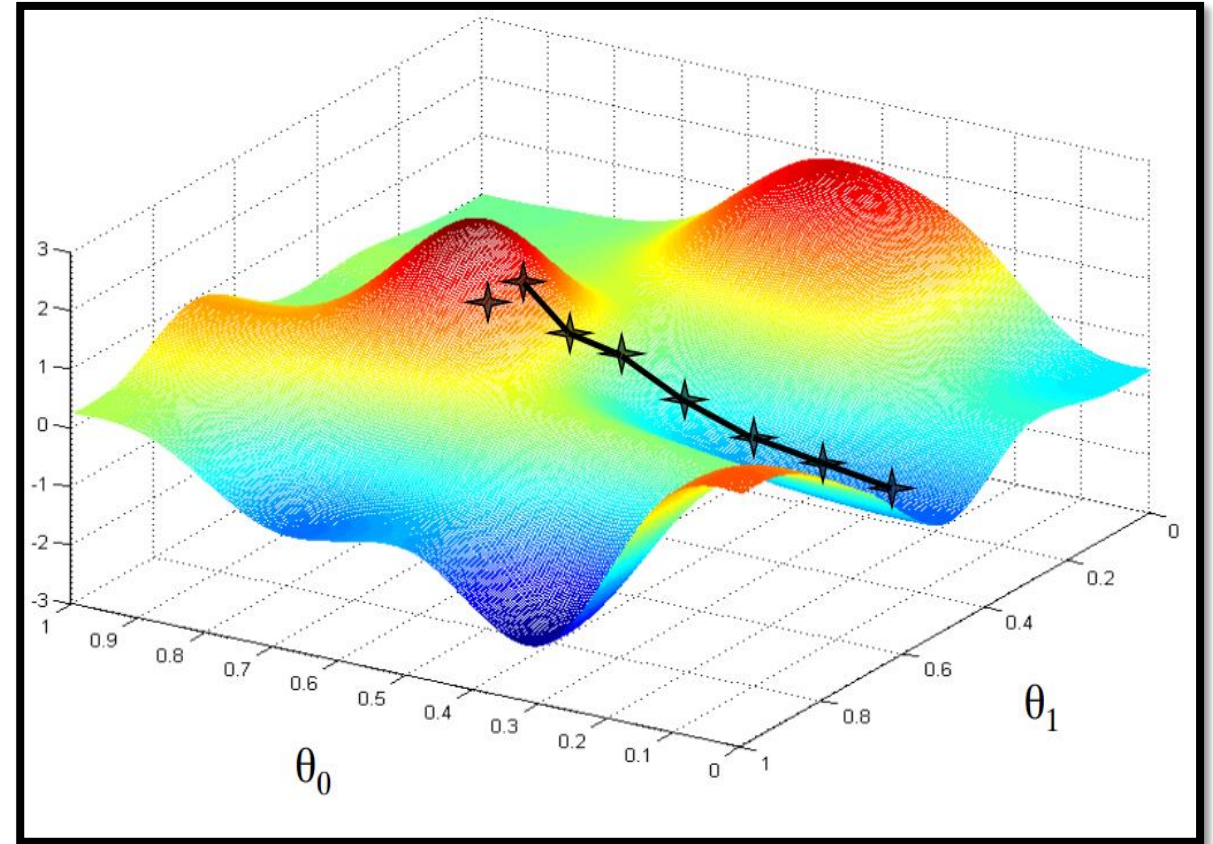
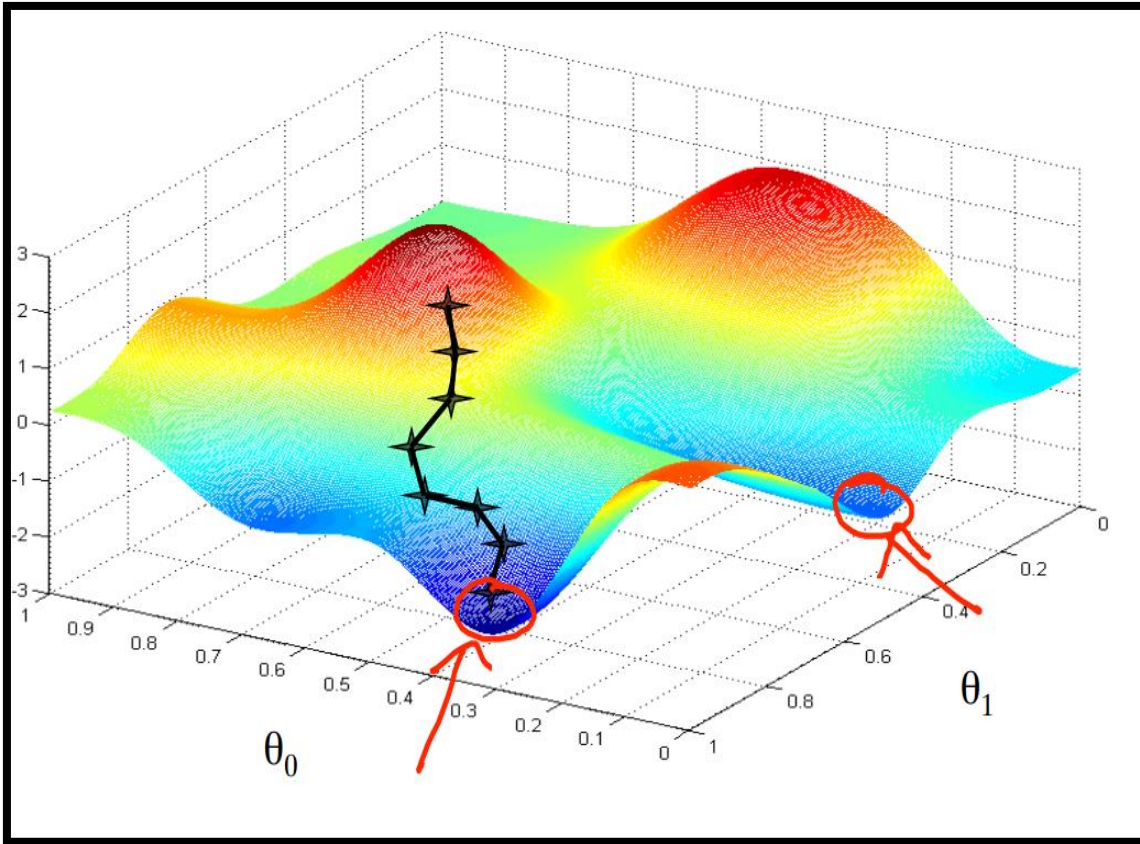
Entrenamiento por Descenso de gradiente

Algoritmo:

- Comenzar con **m** y **b** aleatorios (o sensatos).
- Iterar hasta converger:
 - Calcular: $\Delta \mathbf{E} = \left(\frac{\delta E}{\delta m}, \frac{\delta E}{\delta b} \right)$
 - Actualizar **m**: $m - \alpha \frac{\delta E}{\delta m}$
 - Actualizar **b**: $b - \alpha \frac{\delta E}{\delta b}$
 - Calcular el Error para los nuevos parámetros m y b

Valores iniciales de m y b

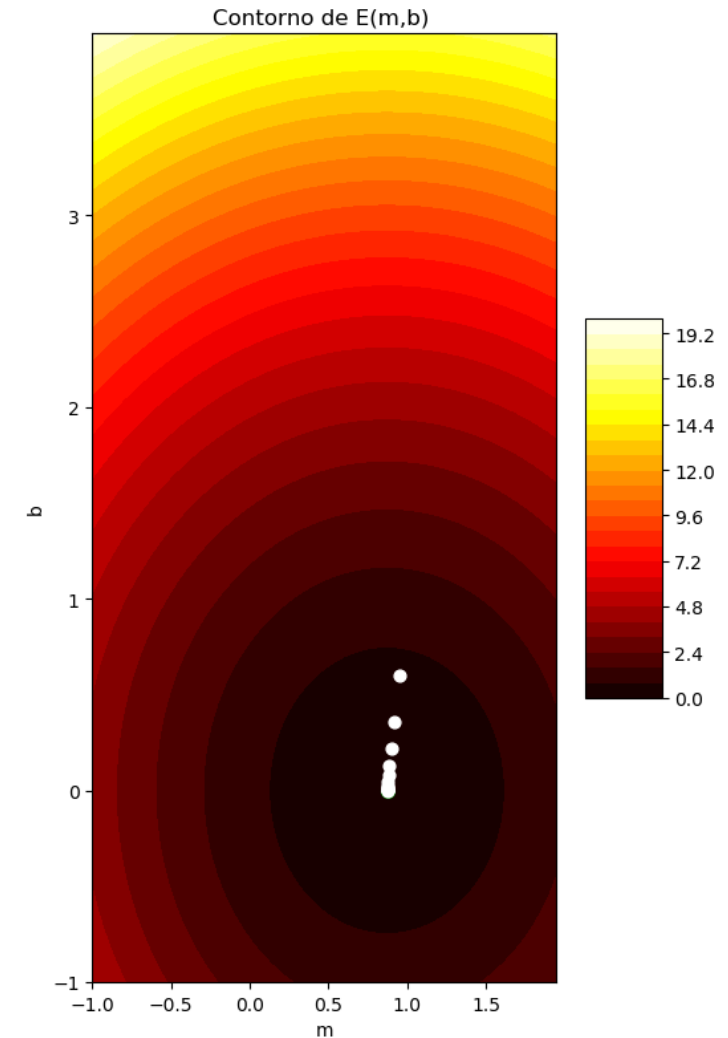
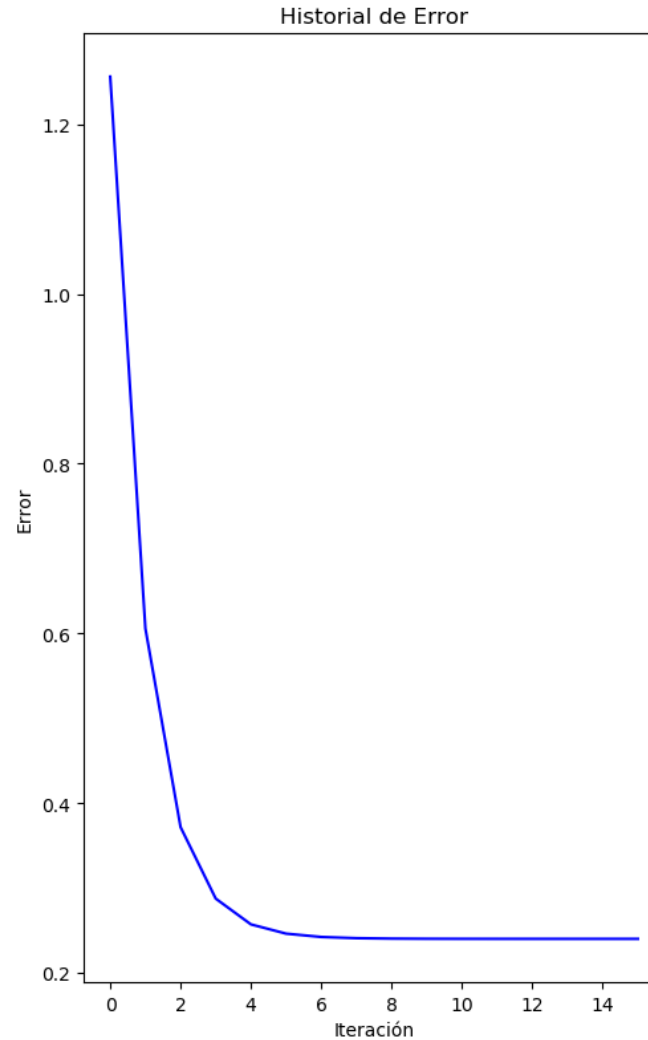
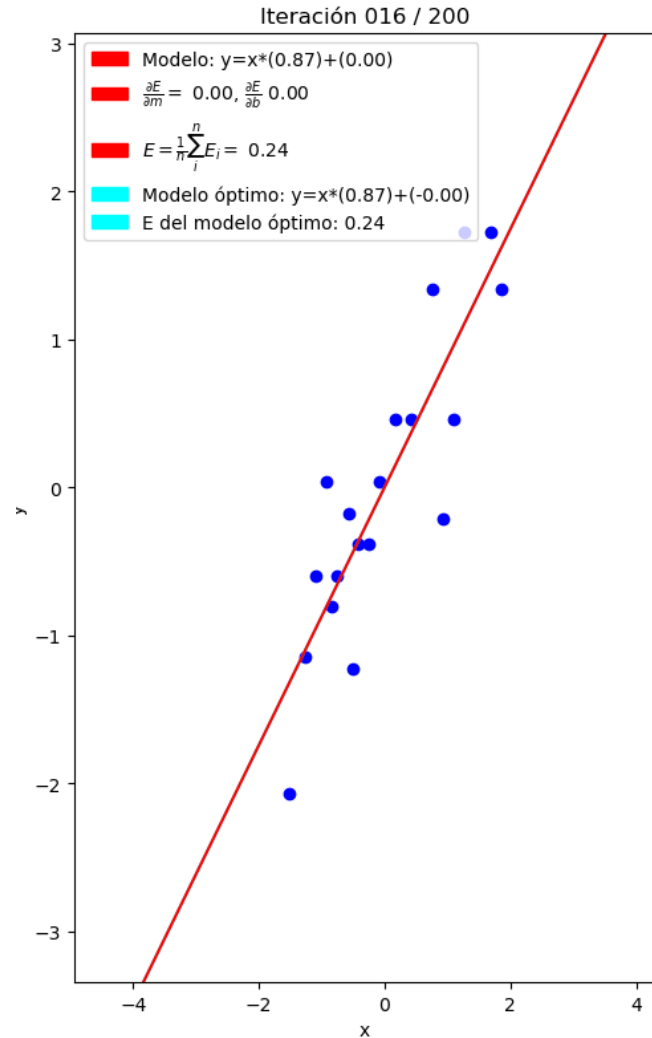
Cuando veamos Redes Neuronales, el descenso podría caer en mínimos locales.



Resumen. Regresión lineal

- RL asume que la relación entre x e y es lineal (con un poco de ruido)
- El modelo que optimizamos es la función: $\mathbf{y} = \mathbf{m}\mathbf{x} + \mathbf{b}$
- La forma estándar de calcular el error es el error cuadrático medio: $E = \frac{1}{n} \sum_i^n (\mathbf{y}'_i - \mathbf{y}_i)^2$
- Dado un dataset, se pueden encontrar m y b óptimos de varias maneras:
 - Las clásicas tienen soluciones analíticas (en forma cerrada).
 - Descenso de gradiente es iterativo pero sirve para varios tipos de modelo.
- Regresión lineal es un modelo de caja blanca; podemos interpretar m y b .

Regresión Lineal. Ejemplo.



Regresión Lineal. Ejemplo.

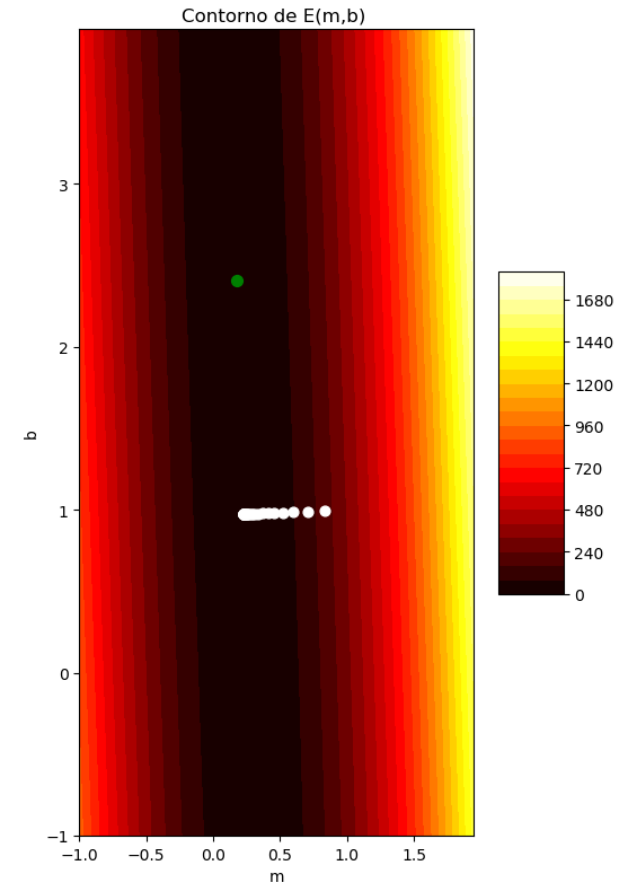
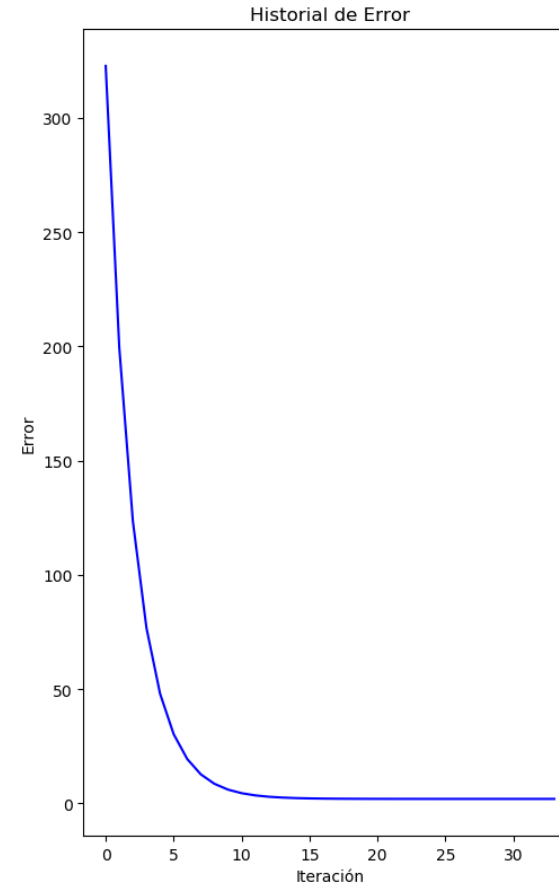
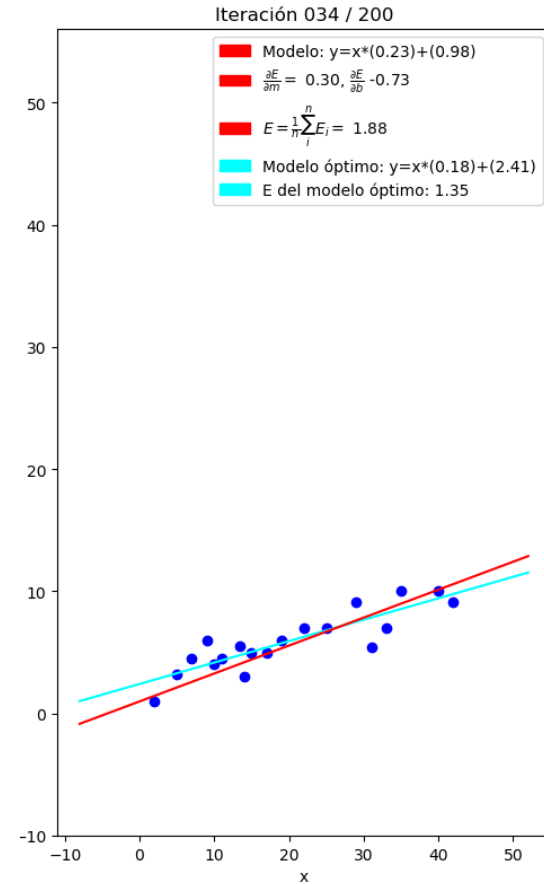
Algunos comentarios sobre el ejemplo:

- Alfa es un hiperparámetro. Si es muy chico, el algoritmo tarda mucho en converger. Si es muy grande, se salta el mínimo y diverge.
- Los valores iniciales de m y b afectan a la optimización. Hay diferentes métodos de inicialización. También se puede aprovechar la experticia del dominio. En el caso de las notas, ¿qué valores serían sensatos?
- Los valores están normalizados

Regresión Lineal. Ejemplo.

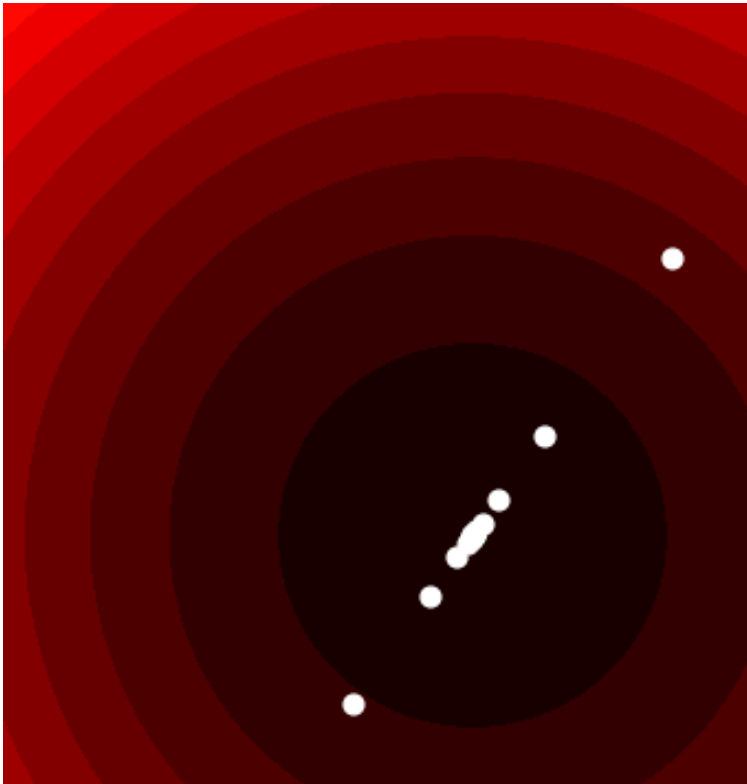
La importancia de la normalización.

Si las variables no están normalizadas, la curva de error tiene diferentes escalas para cada parámetro. El descenso del gradiente podría no encontrar el mínimo, o tardar mucho tiempo.

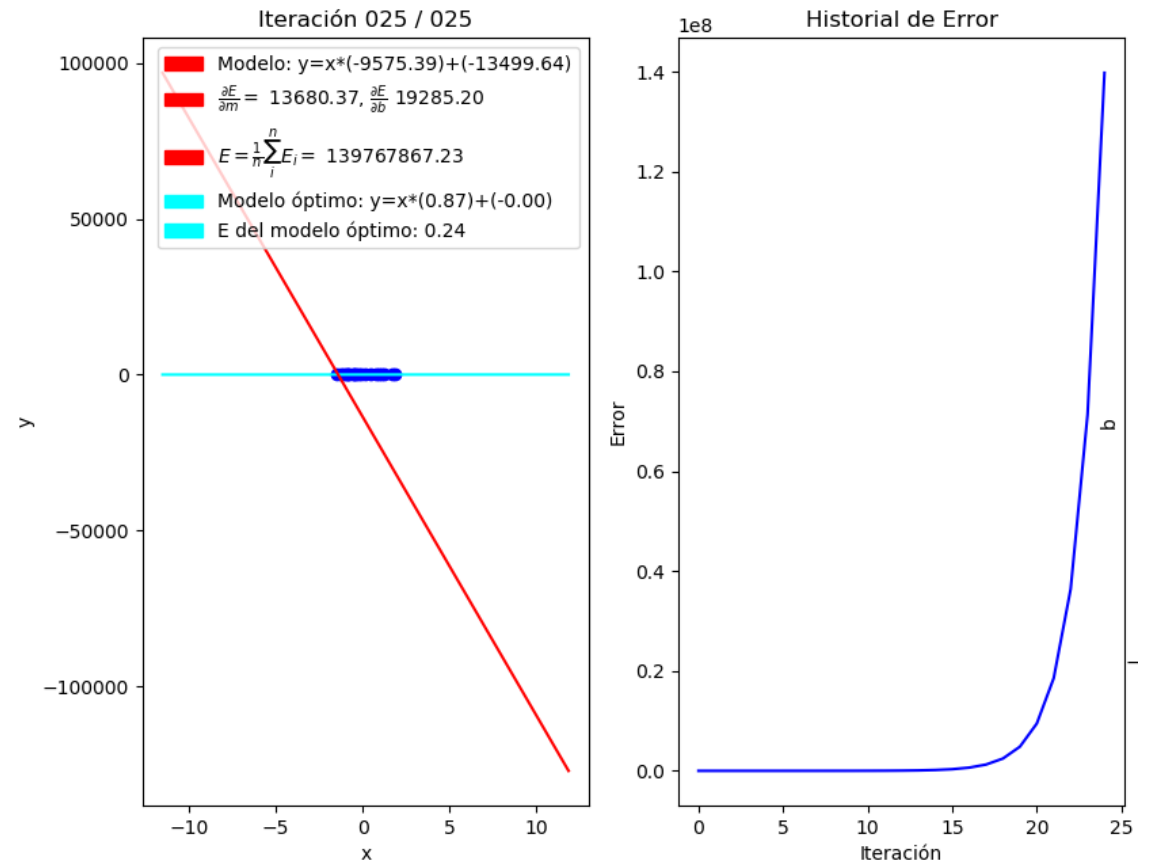


Regresión Lineal. Ejemplo.

Alfa grande, muy
cerca de divergir

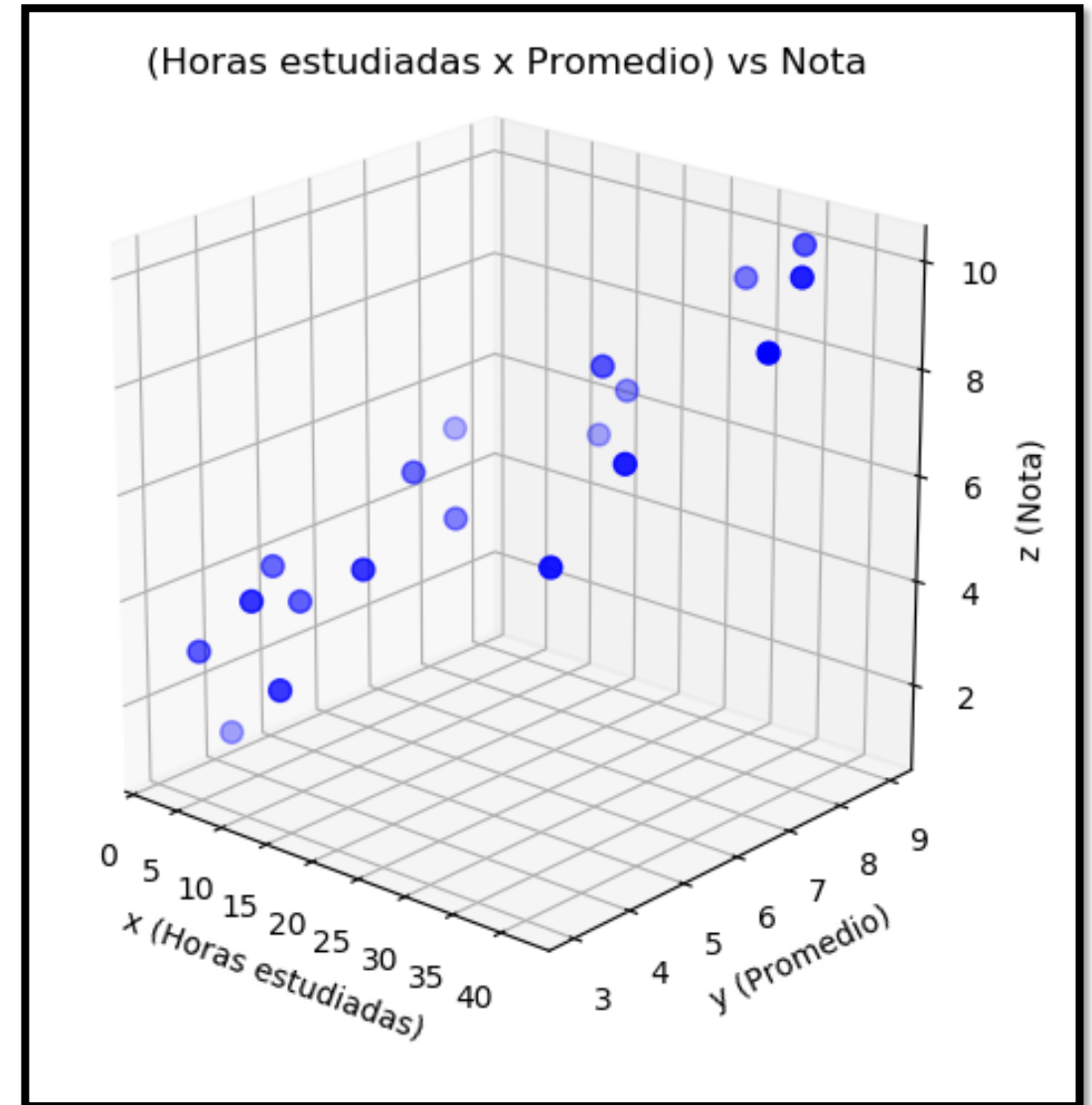


Alfa muy grande, el
algoritmo diverge



Regresión lineal con múltiples variables

- Hasta ahora, los datos tuvieron una sola dimensión ($x_i \in \mathbb{R}$).
- Qué sucede si tenemos más información de cada ejemplo? ($x_i \in \mathbb{R}^d$).



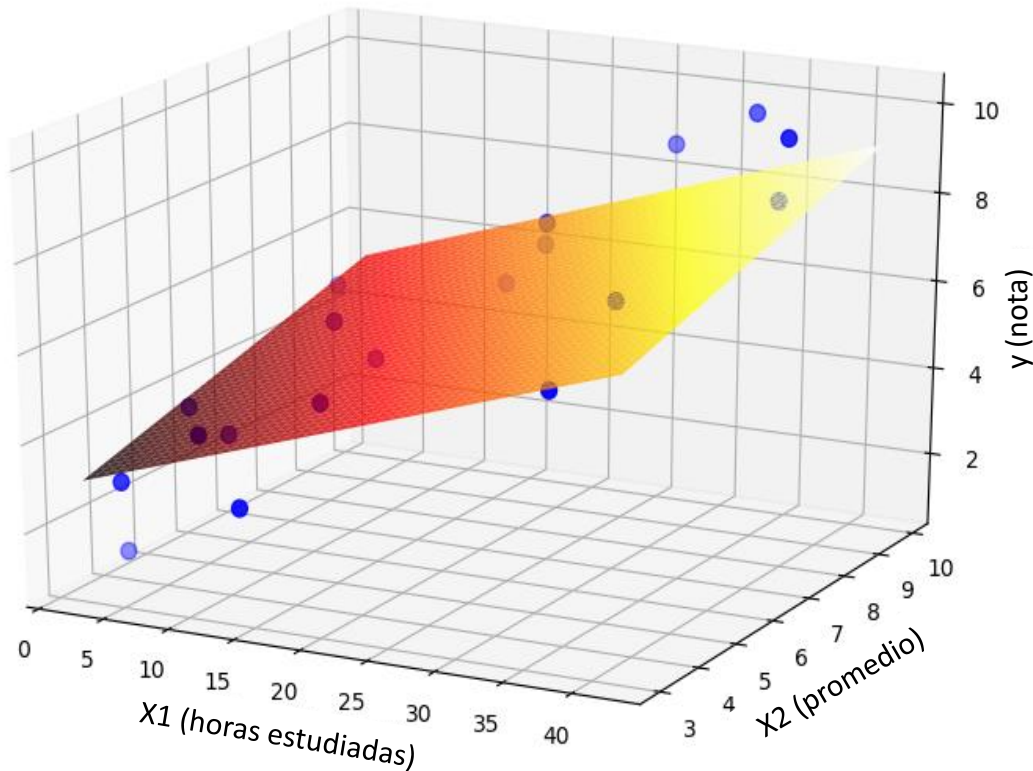
Regresión lineal con múltiples variables

Ahora conocemos también el promedio general del alumno.

Estudio (x)	Promedio (y)	Nota (z)
2	4	1
5	3	3,2
7	4	4,5
9	7	6
10	4	4
11	3	4,5
13,4	5	5,5
14	3	3

Regresión lineal con múltiples variables

(Horas estudiadas x Promedio) vs Nota



Modelo:

$$y = f(x_1, x_2) = \mathbf{x_1 m_1} + \mathbf{x_2 m_2} + \mathbf{b}$$

$$E = \frac{1}{n} \sum_i^n E_i = \frac{1}{n} \sum_i^n (y'_i - y_i)^2$$

$$E = \frac{1}{n} \sum_i^n ((x_1 m_1 + x_2 m_2 + b) - y_i)^2$$

Regresión lineal con múltiples variables

¿Qué cambia?

- Prácticamente nada.
- Ahora tenemos tres parámetros a optimizar (m_1, m_2, b).
- Derivadas?...
- Ya no es posible graficar el Error ☹
- Comienza a ser más difícil encontrar los parámetros manualmente.

Regresión lineal con múltiples variables

Derivadas parciales

$$\Delta \mathbf{E} = \left(\frac{\delta E}{\delta m_1}, \frac{\delta E}{\delta m_2}, \frac{\delta E}{\delta b} \right)$$

$$\circ \frac{\delta E}{\delta m_1} = \frac{2}{n} \sum_i^n (\mathbf{y}'_i - \mathbf{y}_i) \mathbf{x1}_i$$

$$\circ \frac{\delta E}{\delta m_2} = \frac{2}{n} \sum_i^n (\mathbf{y}'_i - \mathbf{y}_i) \mathbf{x2}_i$$

$$\circ \frac{\delta E}{\delta b} = \frac{2}{n} \sum_i^n (\mathbf{y}'_i - \mathbf{y}_i)$$

Regresión lineal con múltiples variables

Derivadas parciales. Notación general para muchas variables.

$$\Delta \mathbf{E} = \left(\frac{\delta E}{\delta x^0}, \dots, \frac{\delta E}{\delta x^m} \right), m = \text{cant features}$$

$$\frac{\delta E}{\delta x^j} = \frac{2}{n} \sum_i^n (y'_i - y_i) x_i^j, \text{ para } j \in \{0..m\}, x^0 = 1$$

Recordar: n = cantidad de ejemplos,
 y = valor real de la función,
 y' = valor computado por mi modelo

Nota: Con 2 *features* no podemos ver la curva del error, pero sí el modelo. Con más *features* solo podremos ver el Error total y los parámetros del modelo entrenado.

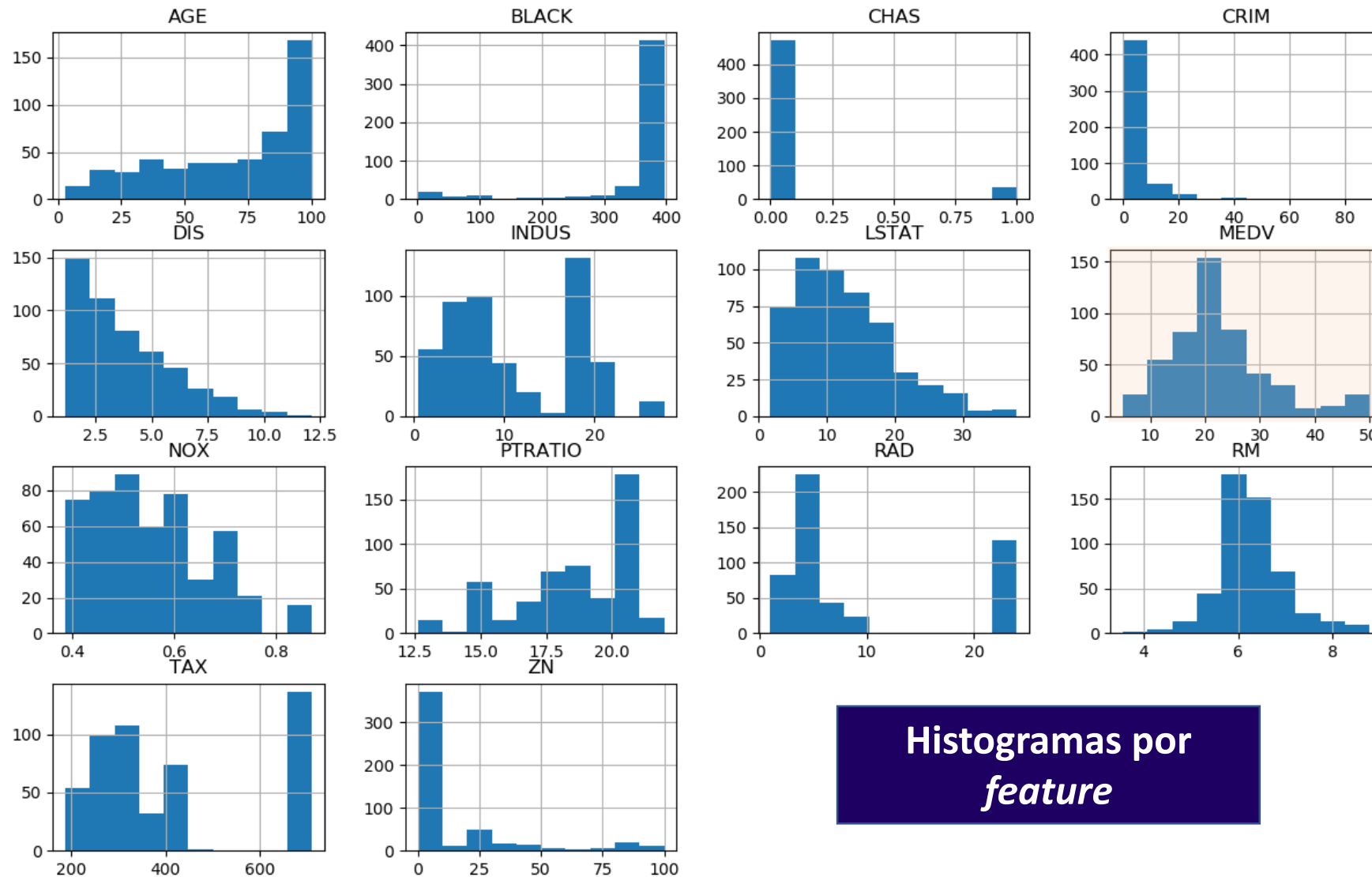
Ejemplo regresión lineal. Boston Housing Dataset.

Objetivo del dataset:

Predecir el precio de una casa en base a información existente en una base de datos.

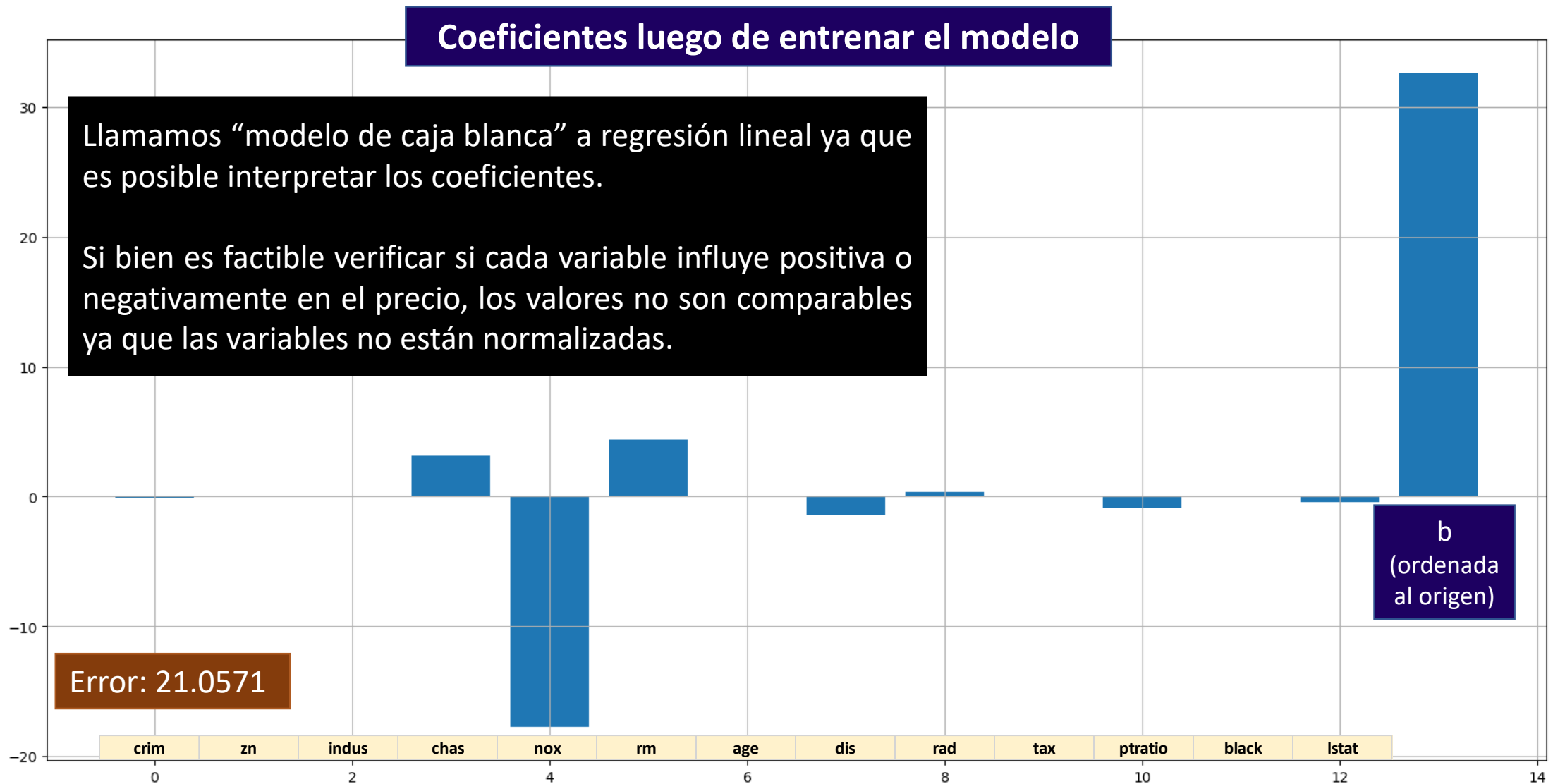
Variable	Detalle
crim	crímenes per cápita
zn	proporción de áreas residenciales
indus	proporción de negocios no retail
chas 1	si está cerca del rio, 0 sino
nox	concentración de nitrógeno
rm	habitaciones promedio
age	proporción de inmuebles anteriores a 1940
dis	distancia promedio a centros de empleo
rad	accesibilidad por autopistas
tax	impuestos
ptratio	relación alumno-docente en escuelas
black	coeficiente de personas negras (dataset del 1980!)
lstat	porcentaje de personas de bajo status
Medv	Valor medio de viviendas ocupadas por sus dueños

Boston Housing Dataset.



Histogramas por
feature

Modelo entrenado. Boston Housing Dataset.



Modelo entrenado. Boston Housing Dataset.

