20 de enero 2026

Agente reactivo

f_A es desconocido

$P \longrightarrow f_A(P) \longrightarrow a$; $f_A : \mathcal{X} \rightarrow \mathcal{Y}$

espacio de entrada — espacio de salida

típicamente $\mathcal{X}$ es $\mathbb{R}^n$

$$\mathcal{X} = \mathbb{R}^n$$
$$x = (x_1, \ldots, x_n) \in \mathbb{R}^n$$

Si $\mathcal{Y} = \mathbb{R}$ yo le doy algo y me devuelve un número. REGRESIÓN.

Si $\mathcal{Y} = \{-1, 1\}$ entonces se le conoce como clasificación binaria

Si $\mathcal{Y} = \{C_1, C_2, \ldots, C_K\}$ se le conoce como clasificación en varias clases o multiclase.

Recordemos que f es desconocida. ¿Cómo la podemos encontrar? Vamos a decir que tengo un conjunto de datos

$\{ x^{(1)}, x^{(2)}, \ldots, x^{(n)} \}$ una muestra de $\mathcal{X}$.
Esto es una muestra de un muestreo desorganizado.

(Datos)
$$D = \{ (x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(n)}, y^{(n)}) \}$$

donde $y^{(i)} = f_A(x^{(i)}) + \underset{\text{error}}{\hat{e}}$ Variable aleatoria de dist. desconocida

¡Mis datos pueden tener errores! El problema entonces es encontrar una función $h^* : \mathcal{X} \rightarrow \mathcal{Y}$ (h^* es la mejor función posible (óptima) tal que $h^* \approx f_A$)

20 Enero 2026

¿Como sabemos que es h^*? $h^* \in H$ (hipotesis posibles)

$H = \{ h_\alpha / h_\alpha : \mathbb{R} \to \mathbb{R}$ donde $h_\alpha(x) = \alpha x, \forall \alpha \in \mathbb{R} \}$

$x = [\, x_1$ (INGRESO MENSUAL), x_2 (DEUDA), x_3 (PAGA A TIEMPO), x_4 (OTROS CRÉDITOS), x_5 (AVAL), x_6 (PROPIEDADES), x_7 (EDAD), x_8 (ESTADO CIVIL), x_9 (SI TIENEN HIJOS), x_{10} , x_{11} (GENERO), x_{12} (ETC) $]$

Yo quiero una funcion que diga si le damos crédito:

$x^{(i)} \in \mathbb{R}^{(12)}$

$$\begin{bmatrix} x_1^{(1)} & \dots & x_R^{(1)} \\ x_1^{(2)} & \dots & x_R^{(2)} \\ \vdots & & \\ x_1^{(n)} & \dots & x_R^{(n)} \end{bmatrix} \begin{bmatrix} \text{Si} \\ \text{No} \\ \\ \text{Si/No} \end{bmatrix} \text{ o } \begin{bmatrix} \$ \\ \$\$ \\ \\ \$\$\$ \end{bmatrix}$$

Clasificación binaria — Regresión (¿Cuanto?)

- Recaudamos datos
- Entrenamos modelo
- Logramos que la computadora resuelva un problema sin programarlo explícitamente

Mi funcion $h : X \, x \longmapsto y$

$\hookrightarrow h : (x^{(i)}, \Theta) = \hat{y}^{(i)}$ [x^i y parametros]

$\hookrightarrow h_\Theta (x^{(i)}) = \hat{y}^{(i)}$

Si Θ es un valor de parámetros fijo

$h_\Theta : X \to y$ Encontrar la mejor funcion es encontrar el mejor vector de parámetros. Estos vectores se pueden derivar.

$h(x) = \sum_{j=1}^{T} w_j x^j + w_0$ ← Esto es un polinomio

Asumiendo que $x^i \in \mathbb{R}$

$h_w \; \mathbb{R} \to \mathbb{R}$ Una regresion, un polinomio grado T

$W = (w_0, w_{0+1}, \dots, w_T) \in \mathbb{R}^{T+1}$, $\hat{y} = w_1 + w$

Análisis de algoritmos

21 de enero '26

Eficiencia = f (n) encontrar solución en menor tiempo posible o con menor uso de memoria

Subjetivo:
Simplicidad
Claridad
Idoneidad para datos esperados

Objetivo:
Run time efficiency
Eficiencia en utilización de memoria

Algoritmos de ordenamiento

1 2 3 4 5 6
5 1 3 2 4 6
6 5 4 3 2 1

- Programar el algoritmo
- Medir el tiempo de ejecución de una implementación particular en una computadora específica y para un conjunto seleccionado de datos. - Popular y útil - RISC y CISC

22 enero 2026

Hipótesis

1. $f: \mathcal{X} \to \mathcal{Y}$ exista y es desconocida
2. tengo $\{x^{(1)}, \dots, x^{(n)}\} \subseteq \mathcal{X}$ una muestra de mis datos con distribución desconocida
3. $D = \{(x^{(1)}, y^{(1)}), \dots, (x^{(n)}, y^{(n)})\}$ donde $y^{(i)} = f(x^{(i)}) + e^{(i)}$ donde $e^{(i)}$ son variables aleatorias

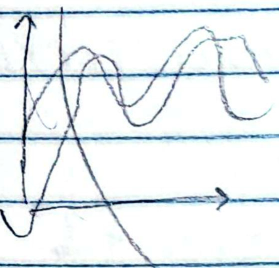
Conjunto de aprendizaje

4. Tenemos una función "parametrizada" $h: \mathcal{X} \times \Theta \to \mathcal{Y}$, típicamente $\Theta = \mathbb{R}'$
5. para un valor específico de θ, $h_\theta: \mathcal{X} \to \mathcal{Y}$
6. $h_\theta \in \mathcal{H}$ conjunto de hipótesis

· h dice que método estoy usando

El aprendizaje supervisado consiste en encontrar $h^* \in \mathcal{H}$ tal que $h^* \approx f$

$\hat{y} \to$ estimada



$L: \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ función de pérdida

$L(Y, \hat{Y}) = L(f(x), h^*(x))$

$L(Y, \hat{Y}) = \frac{1}{2}(Y - \hat{Y})^2$ $\quad y \in \mathbb{R}$

Esta es la del error cuadrado.

$= |Y - \hat{Y}|$, $\frac{|Y - \hat{Y}|}{Y}$ cuando $y > 0$, error absoluto porcentual.

$$L(y, \hat{y}) = \begin{cases} 0 & \text{si } y = \hat{y} \\ 1 & \text{si } y \neq \hat{y} \end{cases} = 1\{y \neq \hat{y}\} \quad y \in \{-1, 1\}$$

22 enero 2026

Hipótesis

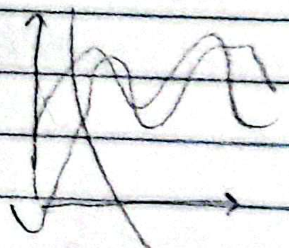
1. $f: X \to Y$ existe y es desconocida
2. tengo $\{x^{(1)}, \dots, x^{(n)}\} \subseteq X$ una muestra de mis datos con distribución desconocida
3. $D = \{(x^{(1)}, y^{(1)}), \dots, (x^{(N)}, y^{(N)})\}$ donde $y^{(i)} = f(x^{(i)}) + e^{(i)}$ donde $e^{(i)}$ son variables aleatorias

Conjunto de aprendizaje

4. Tenemos una función "parametrizada" $h: X \times \Theta \to Y$, típicamente $\Theta = \mathbb{R}^{l}$
5. para un valor específico de Θ, $h_\Theta: X \to Y$
6. $h_\Theta \in H$ conjunto de hipótesis

h dice qué método estoy usando

El aprendizaje supervisado consiste en encontrar $h^* \in H$ tal que $h^* \approx f$



$\hat{y} \to$ estimada

$L: Y \times Y \to \mathbb{R}$ función de pérdida

$L(Y, \hat{Y}) = L(f(X), h^*(X))$

$L(Y, \hat{Y}) = \frac{1}{2}(Y - \hat{Y})^2$ $\quad Y \in \mathbb{R}$

Esta es la del error cuadrado.

$= |Y - \hat{Y}|$, $\frac{|Y - \hat{Y}|}{Y}$ cuando $Y > 0$, error absoluto porcentual.

$$L(y, \hat{y}) = \begin{cases} 0 & \text{si } y = \hat{y} \\ 1 & \text{si } y \neq \hat{y} \end{cases} = 1\{y \neq \hat{y}\}$$

$y \in \{-1, 1\}$

22 enero 2026

$E_{out}(h^*) : \underset{x \in X}{E} [L(f(x), h^*(x))]$

$f \approx h^*$ sii $E_{out}(h^*) \approx 0$

Error fuera de muestra, porque lo calculamos con todos los datos.

Error en muestra solo se puede calcular con datos de aprendizaje

$E_{in}(h^*) = \frac{1}{M} \sum_{i=1}^{M} L(y^{(i)}, h(x^{(i)}))$

(asumimos que todos los datos con igual de probabilidad)

$f \approx h^*$ si → $E_{in}(h^*) \approx 0$
→ $E_{in}(h^*) \approx E_{out}(h^*)$

> Regresion lineal y logistica. Redes neuronales
> Método de árboles de decisión
- MSE, Mid absolute error (MAE), MAPE
> DUDA:

Θ = Vector de parámetros

$h_\Theta(x) = w_1 x_1 + w_2 x_2 + w_3 x_1 x_2 + b$

$\Theta = (w_1, w_2, w_3, b) \in \mathbb{R}^4$

Escaneado con CamScanner

22 enero 2026

DESIGUALDAD DE HOEFFDING

$$\Pr\left(\left| E_{out}(h^*) - E_{in}(h^*) \right| > \overset{\text{error}}{\varepsilon}\right) \le 2\, e^{-2\varepsilon^2 m}$$

donde M es el tamaño de la muestra

la probabilidad que no se parezcan.

Si yo tengo poco datos no se puede decir nada acerca del aprendizaje

¿Como sabemos cuantos datos necesitamos?

dvc (dimensión vechnik - chernovensky)

$d_{vc}(H) \approx$ # de parametros independientes

El aprendizaje es posible si $10 * d_{vc}(H) \ll M$

$\longleftrightarrow \quad E_{in}(h^*) \approx E_{out}(h^*)$

23 enero 2026

- Necesitamos tener 10 veces más datos que parámetros independientes

atributos

m casos x n atributos

Arreglo de casos

$$
\begin{bmatrix} x^{(1)T} \\ x^{(2)T} \\ \vdots \\ x^{(m)T} \end{bmatrix}
\begin{array}{c} \rightarrow \\ \rightarrow \\ \\ \rightarrow \end{array}
\begin{array}{c}
\begin{array}{cccc} a_1 & a_2 & \cdots & a_n \end{array} \\
\begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \cdots & x_n^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \cdots & x_n^{(2)} \\ \vdots & \vdots & \cdots & \vdots \\ x_1^{(m)} & x_2^{(m)} & \cdots & x_n^{(m)} \end{bmatrix}
\end{array}
\quad
\begin{array}{c} y \\ \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{bmatrix} \end{array}
$$

$X_{(m,n)}$ matriz $\qquad y_{(m,1)}$ y es un vector

$x^{(i)} \in \mathbb{R}^n \; ; \; y^{(i)} \in \mathbb{R}$

conjunto de hipótesis $h_\theta(x)$

$h_\theta(x) = w_1 x_1 + w_2 x_2 + \ldots + w_n x_n + b$

Función lineal?

$h_\theta(x) = w_1 x_1 + w_2 x_2 \ldots + w_n x_n + b = \sum_{j=1}^{n} w_j x_j + b$

$= w^T x + b = x^T w + b$

Si $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$

$\quad w = (w_1, \ldots, w_n) \in \mathbb{R}^n$

$\theta = (w_1, \ldots, w_n, b) \in \mathbb{R}^{n+1}$

Aprendizaje implica que $E_{in}(h^*) \approx E_{out}(h^*)$

> Tener suficientes datos

> Que $E_{in}(h^*) \approx 0$

> Quisiera que $h^* \in \arg\min_{h \in H} E_{in}(h)$

23 enero 2026

$h^* \equiv \arg\min_{h \in H} E_{in}(h) \longleftrightarrow \theta^* = w^*, b^* = \arg\min_{\substack{w \in R^n \\ b \in R}} E_{in}(h_{w,b})$

> b es sesgo aquí

> Encontrar mejores argumentos

~~mejor w, mejor b~~

$$w^*, b^* = \arg\min_{\substack{w \in R^n \\ b \in R}} \frac{1}{M} \sum_{i=1}^{M} \frac{1}{2} \left(y^{(i)} - h_{w,b}(x^{(i)})\right)^2$$

MSE

error cuadrado promedio o mean square error

24:57

$x^{(i)} = y^{(i)}$

$$w^*, b^* = \arg\min_{\substack{w \in R^n \\ b \in R}} \frac{1}{M} \sum_{i=1}^{M} \frac{1}{2} \left(y^{(i)} - x^{(i)T} w - b\right)^2$$

$$= \arg\min_{\substack{w \in R^n \\ b \in R}} \frac{1}{2M} \left[(y^{(1)} - x^{(1)T} w - b) \ldots (y^{(m)} - x^{(m)T} w - b) \right]$$

todo $\times \begin{bmatrix} y^{(1)} - x^{(1)T} w - b \\ y^{(2)} - x^{(2)T} w - b \\ \vdots \\ y^{(n)} - x^{(n)T} w - b \end{bmatrix}$

$$\underbrace{\begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(n)} \end{bmatrix}}_{y} - \underbrace{\begin{bmatrix} x^{(1)T} \\ x^{(2)T} \\ \vdots \\ x^{(n)T} \end{bmatrix}}_{X} w - \underbrace{\begin{bmatrix} \vdots \\ \vdots \end{bmatrix} b}_{b}$$

23 enero 2026

entonces

$$= \arg \min_{\substack{w \in \mathbb{R}^n \\ b \in \mathbb{R}}} \frac{1}{2M} \left[y - X^T w - \vec{1} b \right] \left[y - Xw - \vec{1} b \right]$$

33:19

$$= \arg \min_{\theta \in \mathbb{R}^{n+1}} \frac{1}{2M} \left[y - [X, \vec{1}]\theta \right]^T \left[y - [X, \vec{1}]\theta \right]$$

- mínimo respecto a θ donde θ es un vector

→ DEMO $35x^5 + 4x + 3$

Derivas e igualas a 0

$$3 \cdot 35x^2 + 4 = 0$$

$$x = \pm \sqrt{\frac{-4}{3 \cdot 35}}$$

$f(x_1, \ldots, x_n)$

$f(x)$ $x \in \mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}$

$$\vec{\nabla}_x f(x) = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1} \\ \frac{\partial f(x)}{\partial x_2} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{bmatrix}$$

Gradiente

derivadas parciales

- El mínimo se encuentra cuando el gradiente sea 0.
- Calculamos gradiente e igualamos a 0.

23 enero 2026
entonces

$$= \arg \min_{\substack{w \in \mathbb{R}^n \\ b \in \mathbb{R}}} \frac{1}{2M} \left[y - X^T w - \vec{1} b \right] \left[y - X_w - \vec{1} b \right]$$

33:19

$$= \arg \min_{\theta \in \mathbb{R}^{n+1}} \frac{1}{2M} \left[y - [X, \vec{1}] \theta \right]^T \left[y - [X, \vec{1}] \theta \right]$$

- mínimo respecto a θ donde θ es un vector

> DEMO $35x^5 + 4x + 3$

Derivas e igualas a 0

$$3 \cdot 35 x^2 + 4 = 0$$

$$x = \pm \sqrt{\frac{-4}{3 \cdot 35}}$$

$f(X_1, \ldots, X_n)$

$f(X) \quad x \in \mathbb{R}^n, \; f: \mathbb{R}^n \to \mathbb{R}$

$$\vec{\nabla}_x f(x) = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1} \\ \frac{\partial f(x)}{\partial x_2} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{bmatrix}$$

Gradiente

derivadas parciales

- El mínimo se encuentra cuando el gradiente sea 0.
- Calculamos gradiente e igualamos a 0.

23 enero 2026

$$= \arg\min_{\theta \in R^{n+1}} \frac{1}{2M} \left[y - [X_e]\theta\right]^T \left[y - [X_e]\theta\right]$$

$[X, \vec{1}]$

$$\left[y - X_e\theta\right]^T \left[y - X_e\theta\right]$$

$$\frac{\partial V(x)^2}{\partial x} = 2V(x) \frac{\partial V(x)}{\partial (x)} \; ; \; \frac{\partial (y - kx)^2}{\partial x}$$

$$= -2(y - kx)k$$

entonces

$$\underset{(n+1,m)}{X_e^T} \frac{-2}{2M} \underset{m,1}{\left[y - X_e\theta\right]}$$

$$X_e^T\left[y - X_e\theta\right] = \vec{0}$$

$$X_e^T y - X_e^T X_e\theta = \vec{0}$$

$$X_e^T X_e\theta = X_e^T y$$

$$\theta = \underbrace{\left[X_e^T X_e\right]^{-1} X_e^T}_{\text{pseudo inversa de penrose }(X_e)} y$$

Metodo de mínimos cuadrados

o pinv (X_e) $\longrightarrow$ numpy en python