

Numpy Python.
 float bool str list Dict

listas = [1, 1.2, false, 'str', [1, 2, 3], {key: value,}] Flexible

np.array = [1, 2, 3, 4] iterable
↑
pos

matrices A (F, c)

A (1, 4)

→ 4 columns.
A = [1, 2, 3, 4]
→ 1 file

Arreglos en n Dimensiones

1D = [1 2 3 4] (1, 4)
 ↑ Filas
 ↓ Columnas

2D = $\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 0 & 1 & 2 \end{bmatrix}$ (3, 4)

D

	0	1	2	3	4
0	1	2	3	4	5
1	6	7	8	9	10

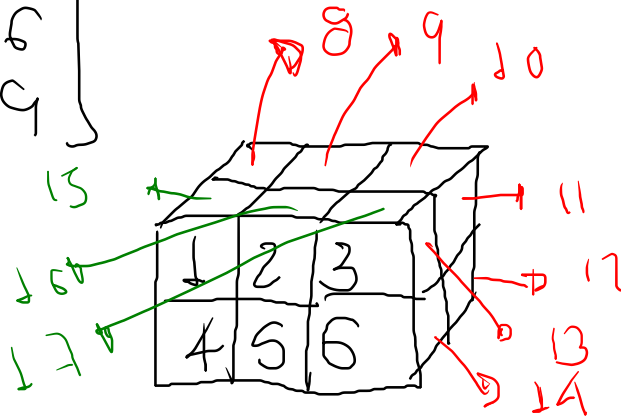
$$D(2,3) = 9.$$

Arreglo con 3 dimensiones

$$1D = [1 \ 2 \ 3 \ 4] \quad (1,4) \quad \checkmark$$

$$2D = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \quad (3,3) \quad \checkmark$$

$$3D =$$



$$(2,3,2) \quad \checkmark$$

$$4D$$

$$\text{listA} = [\text{[3, [3, [3, [3]}]]] \quad \underline{\underline{\text{listA}}}$$

$$\text{listA 3 D} = [\text{[[3], [3], [3]}]$$

$$l = [\text{[1, 2, 3], [1, 2, 3], [1, 2, 3]}] \Rightarrow \begin{bmatrix} \textcircled{1} & 2 & 3 \\ 1 & 2 & 3 \\ 1 & 2 & \textcircled{3} \end{bmatrix} \begin{matrix} (0,0) \\ (2,2) \end{matrix}$$

lista 3 Dimensionales

(2, 3, 2)

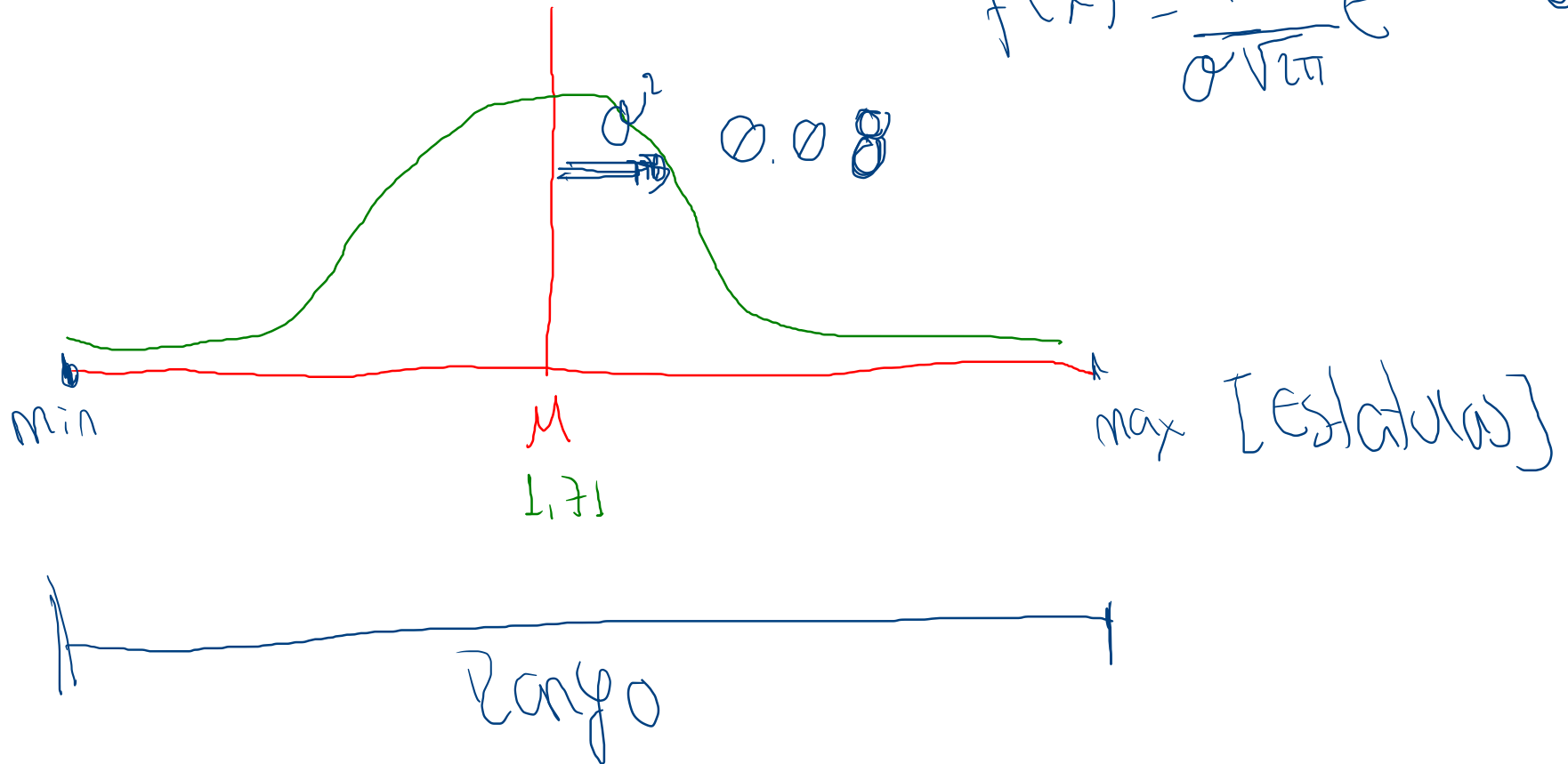
$$l = \left[\begin{array}{c} [[] , [] , []] \\ \downarrow \\ [] , [] \end{array} , [[] [] []] \right]$$

A B C D
A B C D

$$\begin{array}{c} [[]] \\ \downarrow \\ [] , [] \end{array} , \begin{array}{c} [[]] \\ \swarrow \searrow \\ [] [] \end{array}$$

Distribucion Normal

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$



• Proceso de Analítica, calidad de Educación en Colombia.

→ Datos, (ICFES),
↳ Educación Básica
Saber 11 ~~11~~ → Resultados.
↳ Educación Superior
Saber Pro.

Caso de Negocio
↳ Comparativa.
• Extracción
• Análisis
• Resultados.

netH = np.append(arr, value)

↳ [1]

= [0.1 0.2 0.25 0.5 1 2 2.5 5 10 20]

↓ ↓ ↓
56 50 41

└

Frecuencia Relativa.

Total datos 3933

56 \rightarrow 63/3933

50 \rightarrow 79/3933.

i \rightarrow
 $mem = [56 \ 50 \ 41 \ \dots \ 52 \ 4, 21]$

$arr = \begin{cases} 56 \rightarrow 63/3933 \\ 50 \rightarrow 79/3933 \end{cases}$

12 13 14 \dots 98

$contor[12, mem]$

$pr.aprend()$

Lista
Arreglo = []

Arreglo.append(1)

numpy

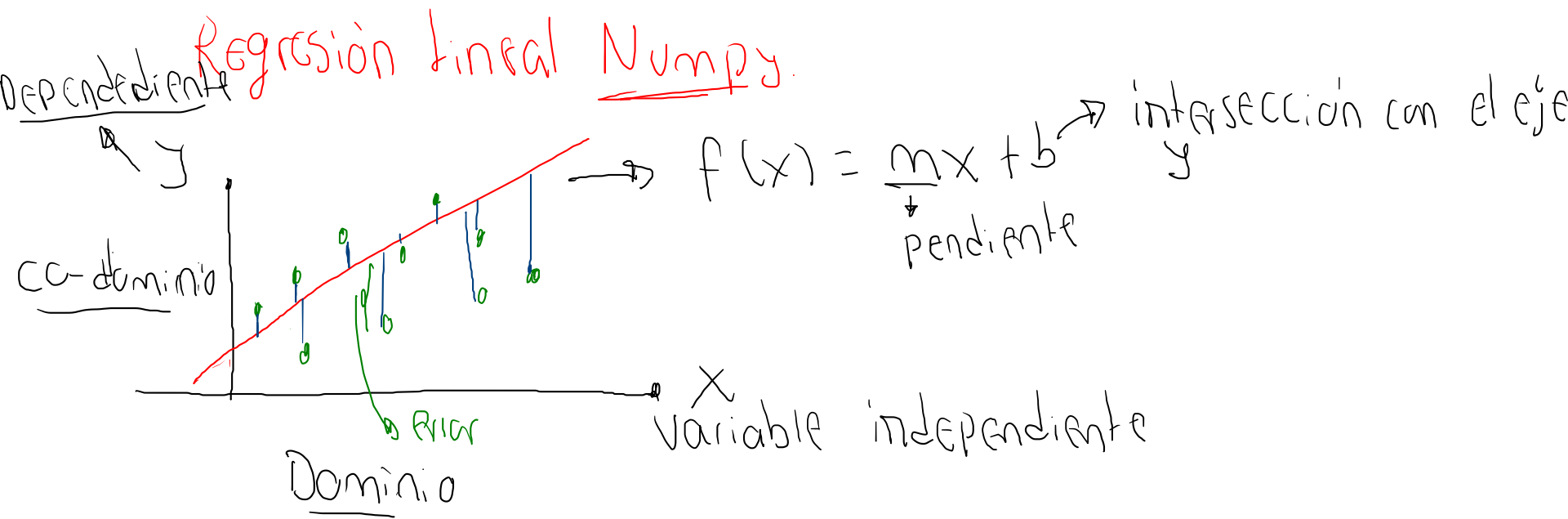
Arreglo = np.empty(0)

np.append(Arreglo, arr)

→ Asignarlo a un arreglo resul.

filas = 3
column = 3

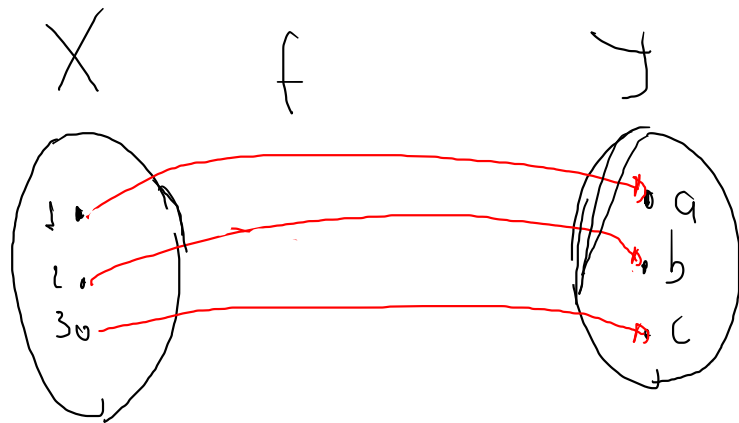
$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$



Error cuadrático medio $\rightarrow \bigcirc$

(minimizar el error cuadrático medio).

función matemática: Relación entre los elementos de 2 Conjuntos. El Dominio y el co-Dominio

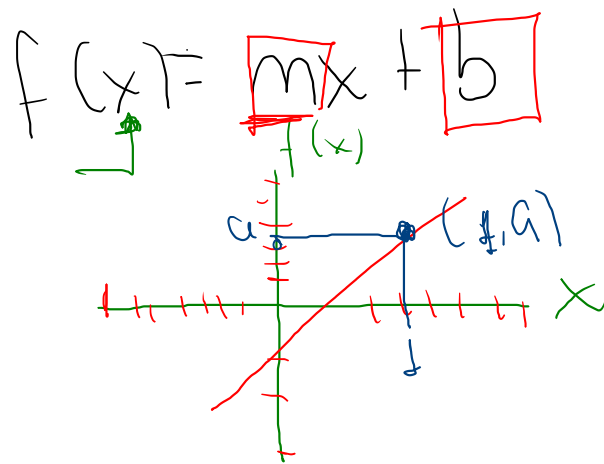


Dominio

Co-Dominio
Rango

$$f: X \rightarrow Y$$

$$f = \{(1, a), (2, b), (3, c)\}$$




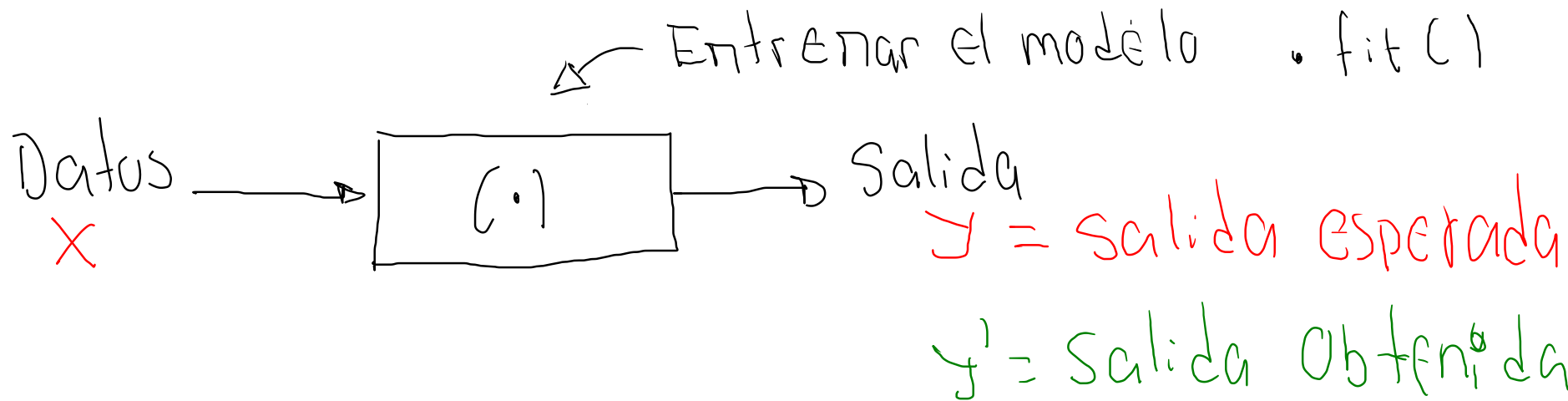
El modelo resultante

$$f(x) = 0.09183522x + 1.2858...$$

$$f(5) = 0.09183522(5) + 1.2858...$$

$$f(5) = 1.74505534$$


Predicción.



y — y'
↳ Reducir la distancia $mse \rightarrow 0$

$y = y_true$

$y = y_pred$

R^2 = coeficiente de determinación

$[0 - 1]$

↓
No sirve

↳ Es perfecto el modelo