Argentina programa 4.0

Universidad Nacional de San Martín

Módulo 2 - Ciencia de Datos

Semana 1. Elementos de matemática

Funciones

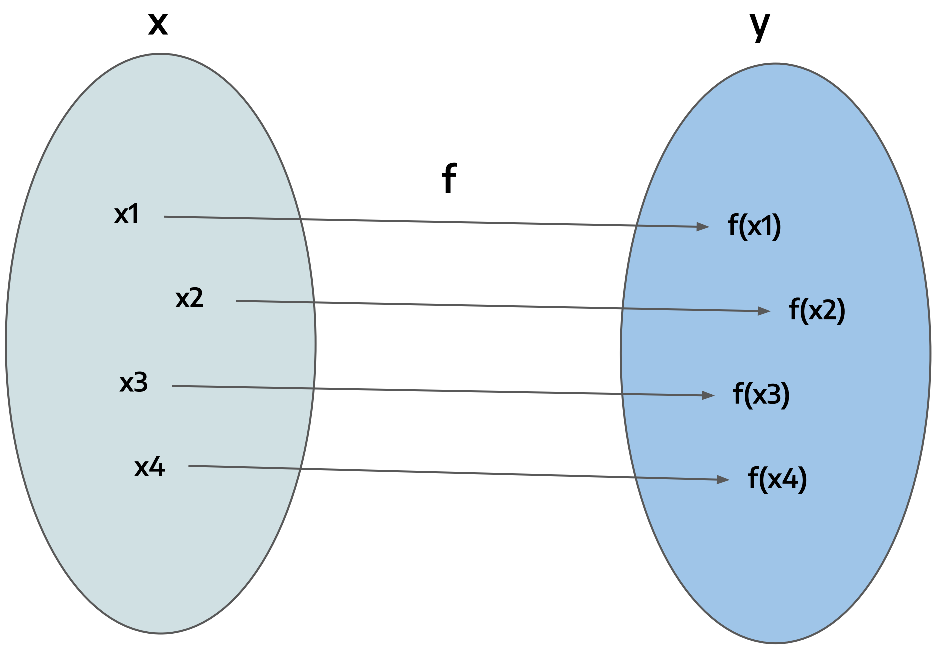
Elementos de matemática: funciones

Funciones

- Fundamentales en cálculo
- Son tipos especiales de *relaciones*
- Se pueden expresar como *reglas.*
    - Por cada input **x**, una función devuelve un output **y**
    - $y = x^2$; $y = a * x + b$; $y = 3.141519 * x^2$; y = random.sample(range(1,12), 11)
- X e Y son variables
    - Area de una circulo = π * radio^2
    - Volumen de una esfera = 4/3 * π * radio^3
    - Costo viaje en taxi (CABA) = $293 (bajada bandera) + ($14.65 por cada 100 metros)

Funciones

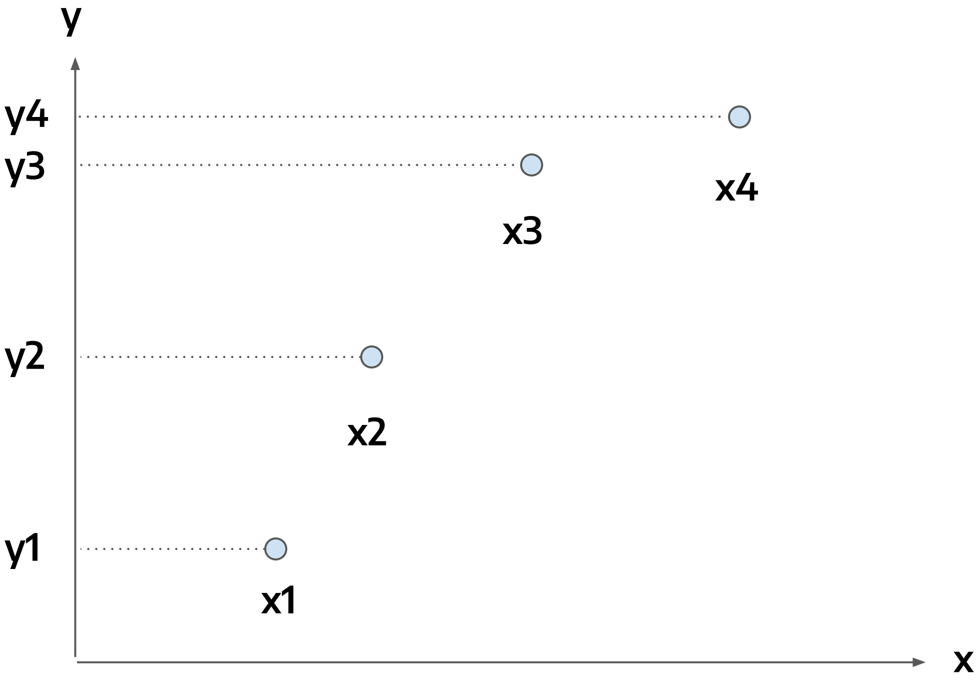
De qué otras maneras podemos representar o visualizarlas?



Conjunto X = valores input
Conjunto Y = valores output
Función f = relación

Funciones

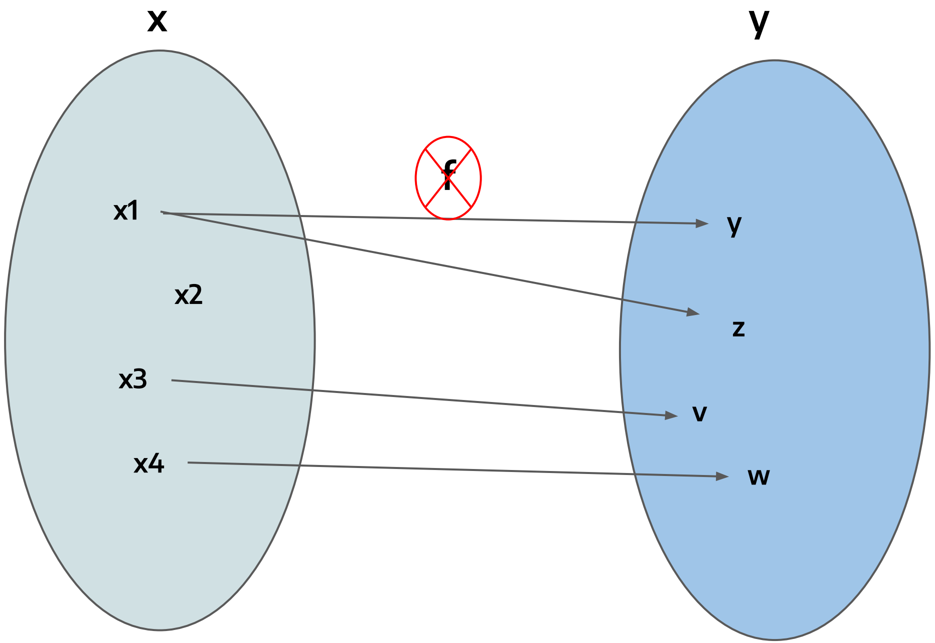
De qué otras maneras podemos representar o visualizarlas?



Conjunto X = valores input
Conjunto Y = valores output
Sistema de ejes cartesianos (x,y)

Funciones

Qué es una función?



Es una función:

- Para cada valor de x, hay un valor de y

No es una función:

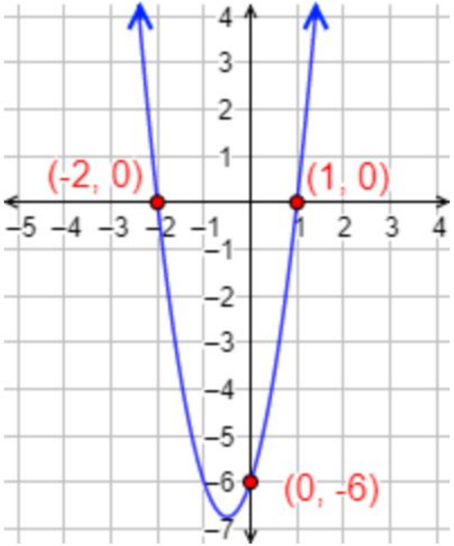
- Hay *más de un valor de x* por cada valor de y

Algunos tipos de funciones matemáticas

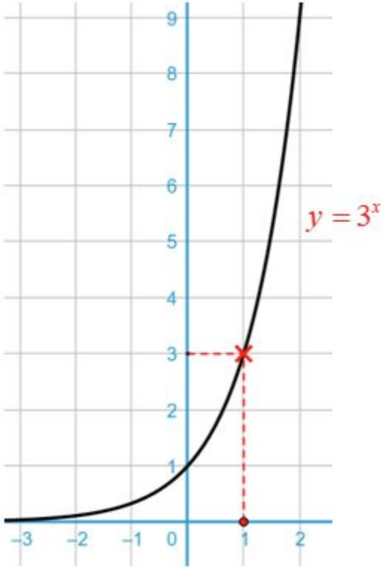
Lineal:	$f(x) = ax + b$
Cuadrática	$f(x) = ax^2 + b + c$
Polinomio de grado n:	$f(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0$
Racionales	$f(x) = (a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0)/(b_n x^n + b_{n-1} x^{n-1} + \ldots + b_0)$
En partes	$f(x) = \lvert x \rvert$ -x, si $x < 0$ y x, si $x \geq 0$ $f(x) = x^2$ si $x<2$ y 4 si $x > 2$
Exponenciales	$f(x) = a^x$
Logaritmicas	$f(x) = \log_a(x)$ $a > 0, a \neq 1$
Trigonométrica:	$f(x) = \sin(x)$ $f(x) = \cos(x)$

Visualización de funciones

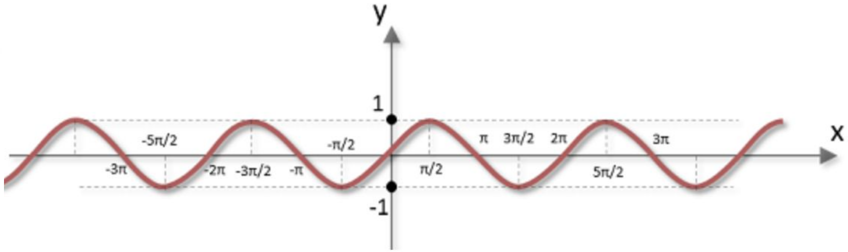
$y = 3x^2 + 3x - 6$



$y = 3^x$



$y = \sin(x)$



Funciones multivariable

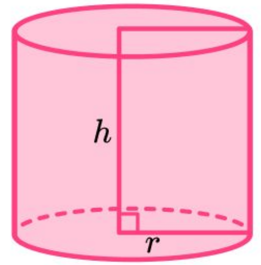
Funciones donde hay más de un input

$f(x,y) = x^2 + y^3$

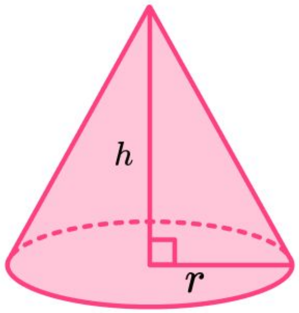
$z = x^2 + y^3$

$y = x_1^2 + x_2^3$

Otros ejemplos



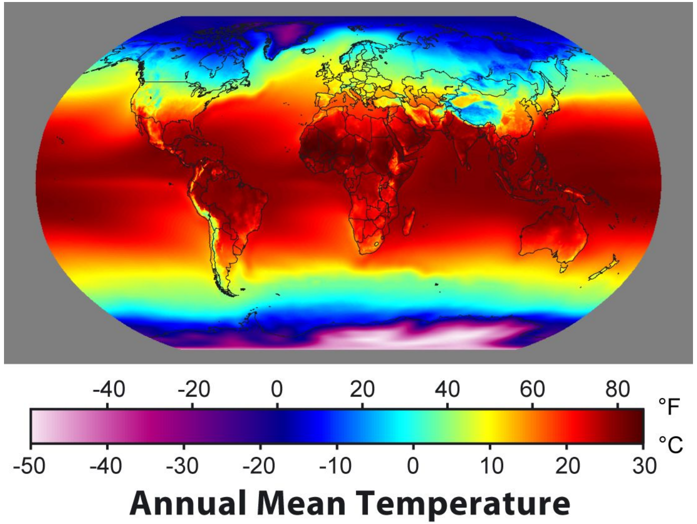
Volume of a cylinder $= \pi r^2 h$



Volume of a cone $= \frac{1}{3}\pi r^2 h$

Funciones multivariable

temperatura = f(longitud, latitud)



score de riesgo financiero = f(ingresos, egresos, ahorros, bienes de capital, ... etc)

$SRF = x_1 - x_2 + x_3 + x_4$

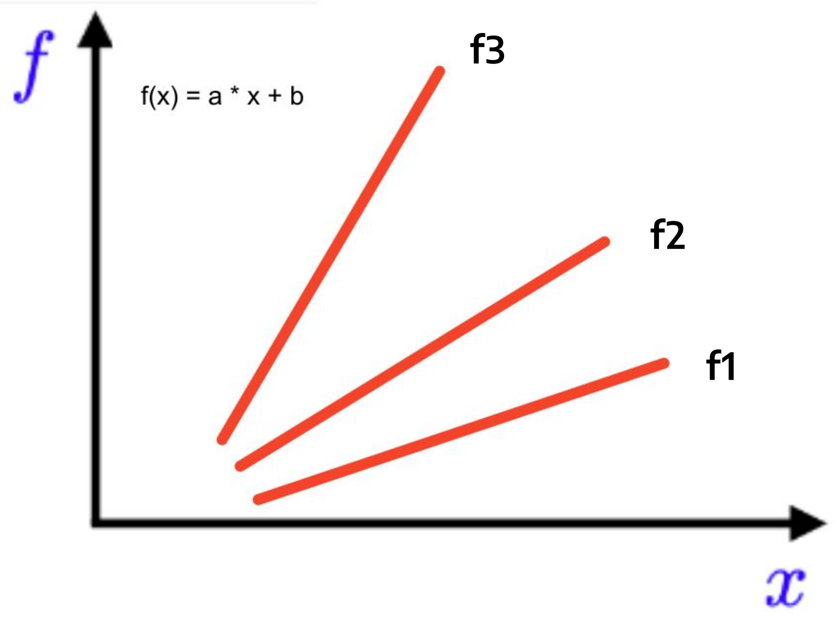
$SRF = (x_1 * 0.5) - (1/3 * x_2) + x_3^2 + (2 * x_4)$

Pesos (weights)

Funciones y velocidad de cambio

Una función es una regla que me permite ***calcular*** un valor de ***output*** (resultado) para cualquier valor válido de input (x).

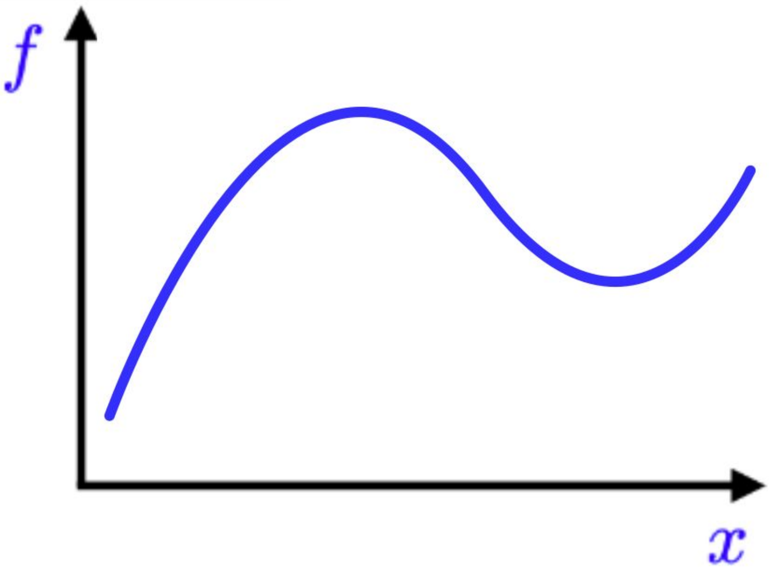
Pero no todas las funciones del mismo tipo son iguales. Qué podemos decir respecto de la manera en que cambia ***y*** con respecto a ***x*** ?

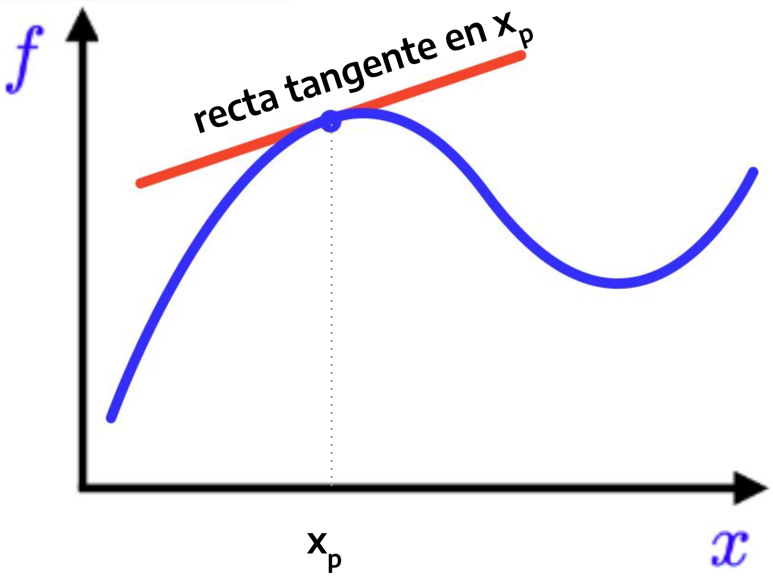


Funciones y velocidad de cambio

Y cómo es la velocidad o grado de cambio de una función más compleja?

Depende del punto donde me pare!





Funciones y velocidad de cambio

En matemática, la velocidad de cambio de una función en algún punto es su ***derivada***.

No vamos a ver más que esto sobre derivadas, hay muchas reglas para calcularlas, y más aprendizaje para quien tenga interés!

Pero si saber que existen, qué son y para qué las vamos a usar:

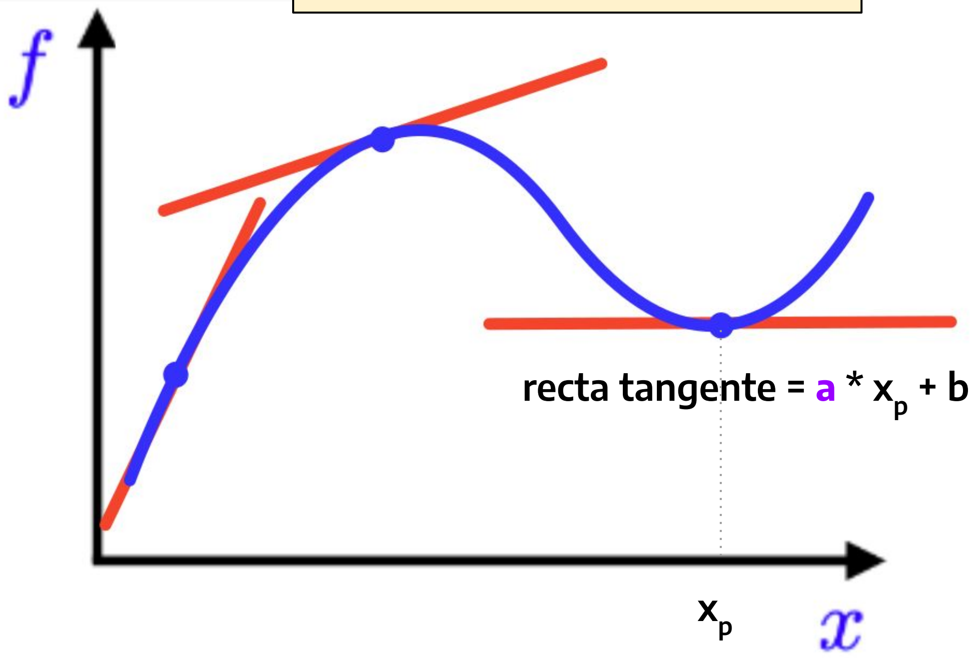
- Para identificar cuándo la función sube (o baja)
- Para identificar dónde están los máximos (o mínimos) de una función

Derivadas (velocidad de cambio)

Cómo medir velocidad de cambio?

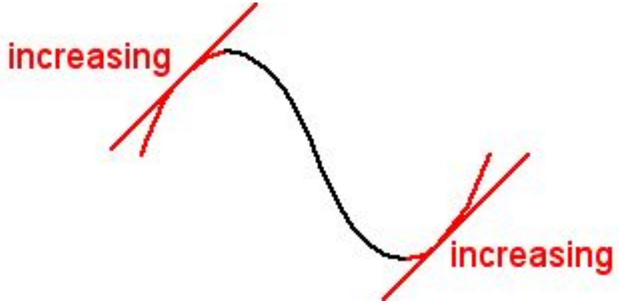
La ***tangente*** a una curva en un punto **P** es una recta que toca a la curva solo en dicho punto, llamado *punto de tangencia*.

Usando el valor de las pendientes de las distintas rectas tangentes en los distintos puntos x_p



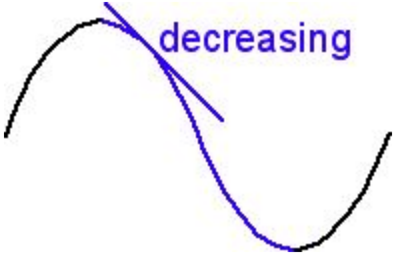
Rectas tangentes con pendiente positiva

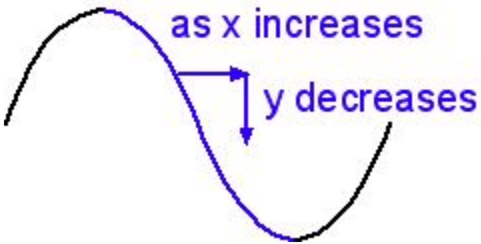
Indican que la función ***sube***



Rectas tangentes con pendiente negativa

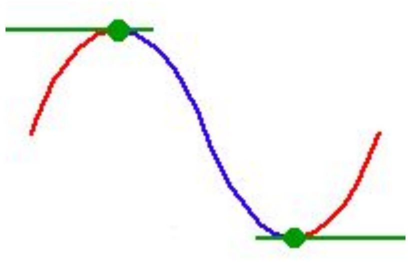
Indican que la función ***baja***





Rectas tangentes con pendiente nula

Cuando la tangente es ***horizontal***, el valor de la pendiente es **cero (0)**.

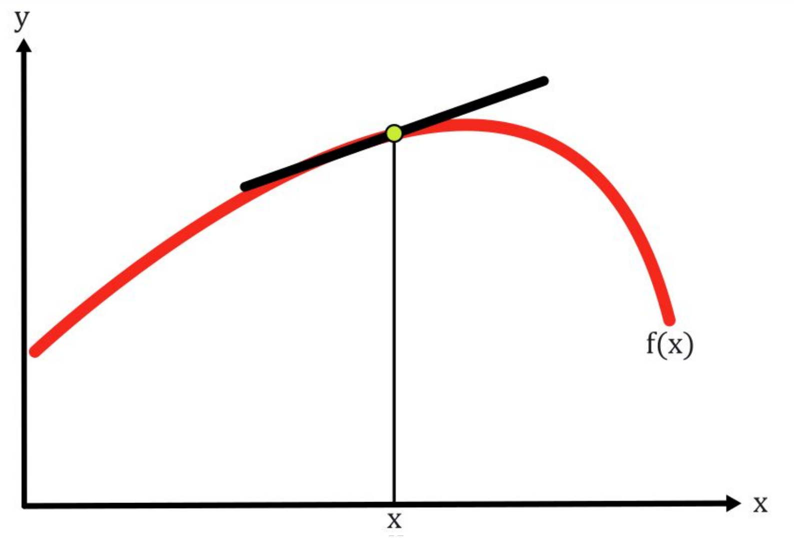


Indican puntos donde la función cambia de sentido (pasa de aumentar a disminuir o *viceversa*)

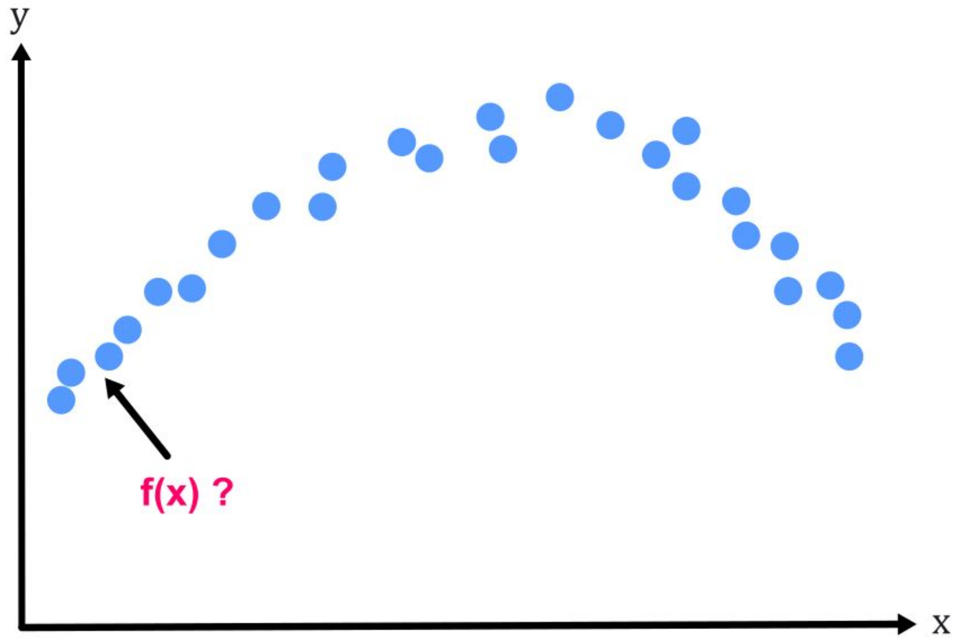
Son los mínimos (o máximos) de la función!!!

Cómo calcular pendientes?

- **Análiticamente (si sabemos calcular derivadas)**
  - Derivar funciones (no lo vamos a ver en este curso)
  - Necesitamos conocer la función!
    - $y = ax+b$; $y = x^2$; $y = 2x^3 + 4x^2 + x$
- **Numéricamente**
  - No necesitamos conocer la función
  - Podemos aproximar las pendientes usando los datos que tenemos



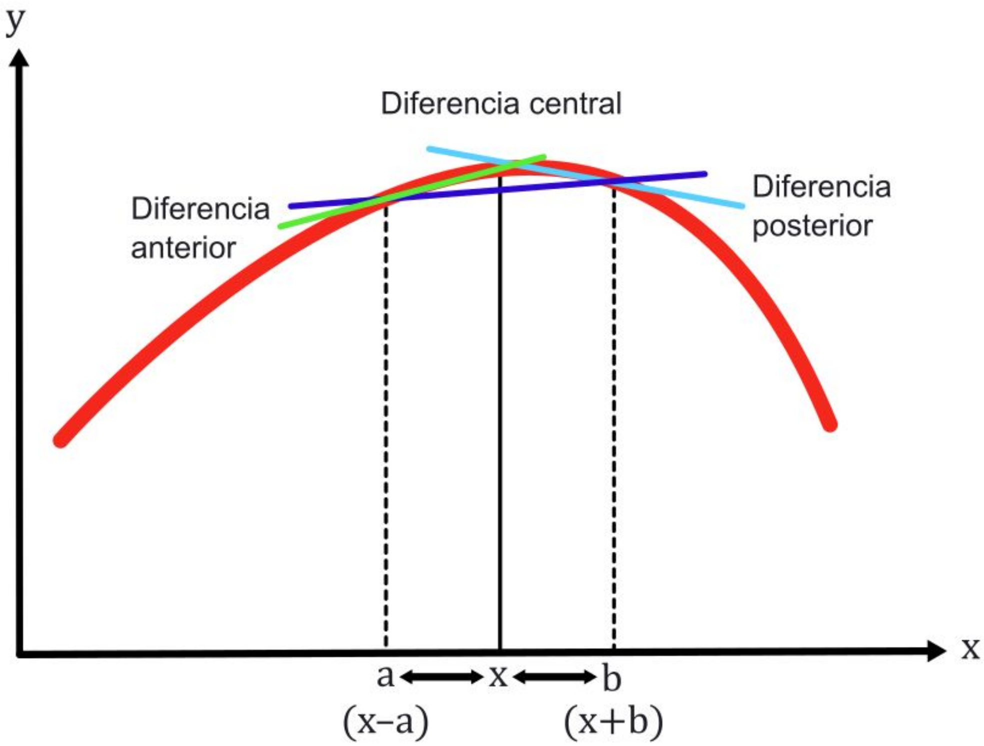
Cómo hacemos si tenemos datos pero no sabemos cuál es la ecuación de la función?



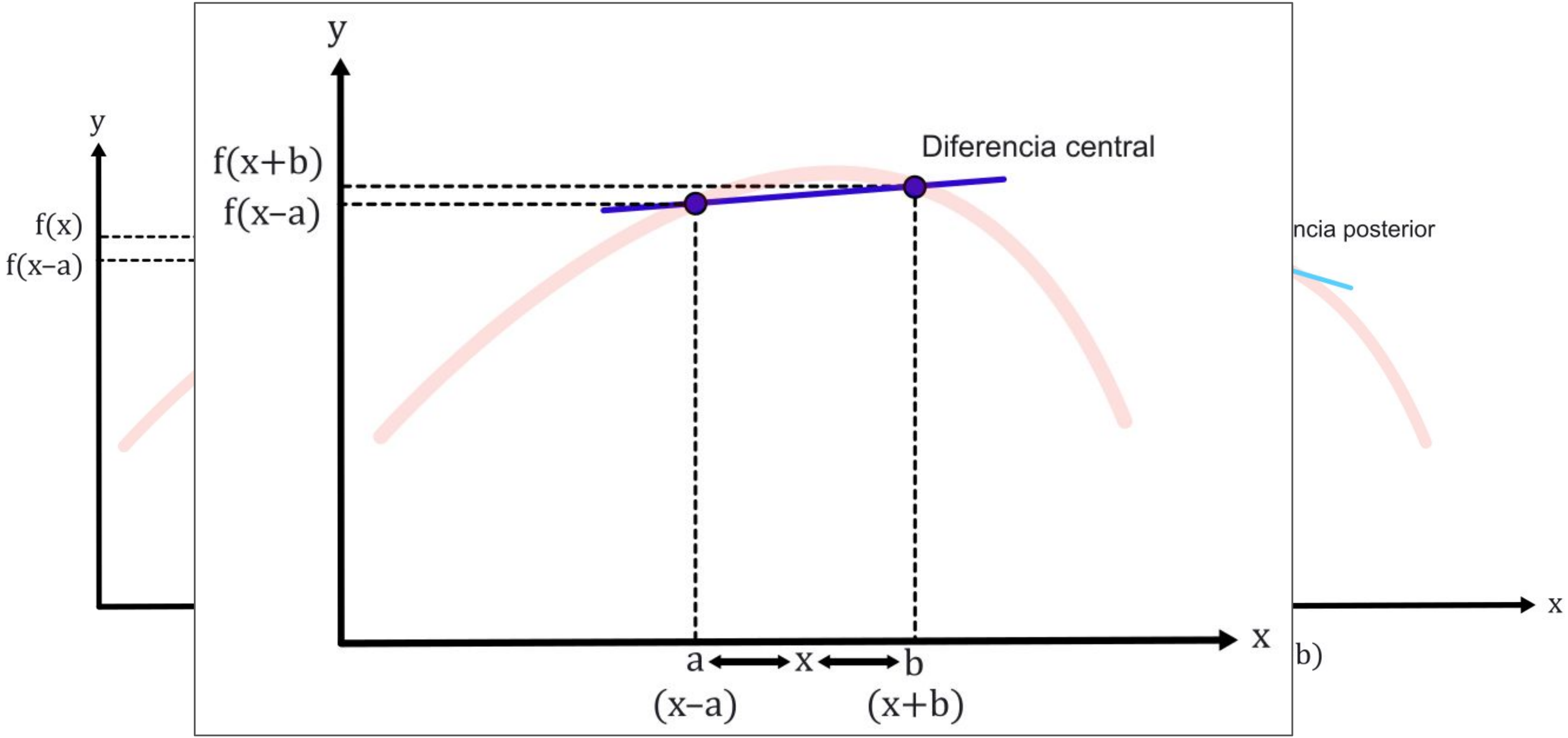
Aproximamos numéricamente

Dado cualquier punto x, usamos puntos vecinos (antes y después)

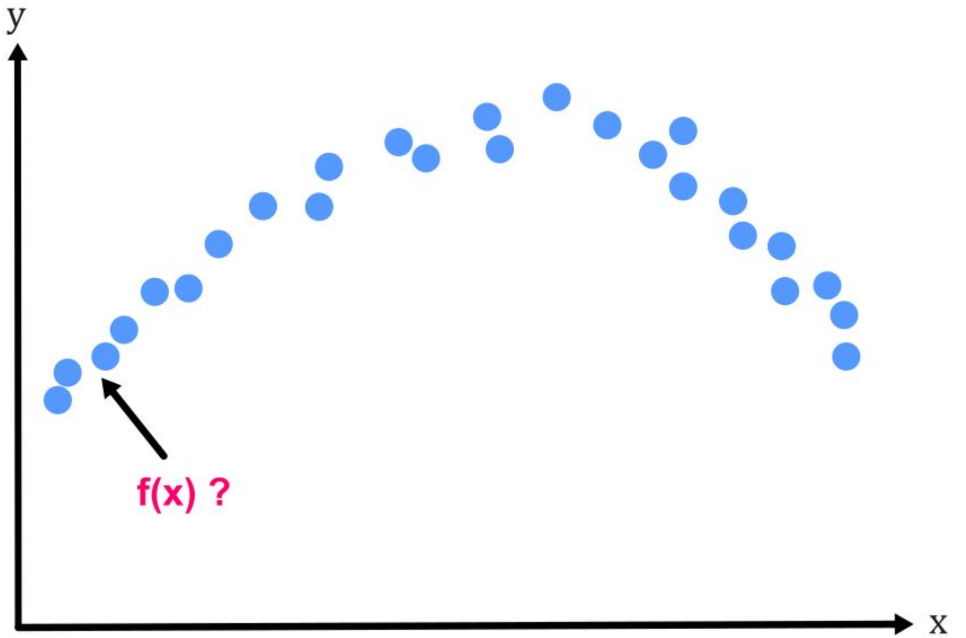
https://es.wikipedia.org/wiki/Diferencia_finita



Aproximación a la derivada



Cómo hacemos si tenemos datos pero no sabemos cuál es la ecuación de la función?



En Python

Podemos aproximarlas usando NumPy (numpy.gradient)

NumPy no conoce la función, hace la estimación solamente en base a los números que existen en un array.

Dado un array de números, numpy.gradient devuelve un array con valores de derivadas (pendientes) del array original. Los valores negativos indican que la función *decrece* en ese punto. Los valores positivos indican que la función *crece* en ese punto.

https://numpy.org/doc/stable/reference/generated/numpy.gradient.html

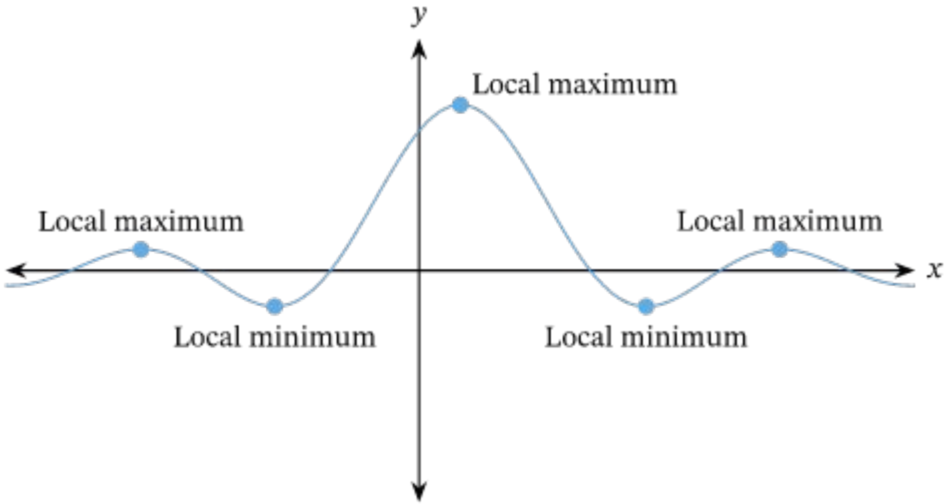
Sobre máximos y mínimos

Máximo: los valores de ***y*** que devuelve la función **bajan** si me muevo en ***x***

Mínimo: los valores de ***y*** que devuelve la función **suben** si me muevo en ***x***

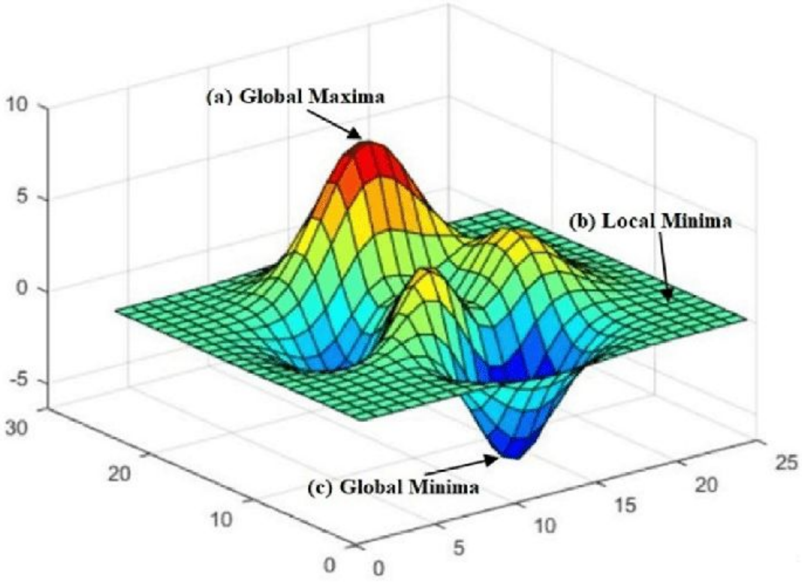
Valores máximos (o mínimos) son ***locales*** con respecto al entorno cercano

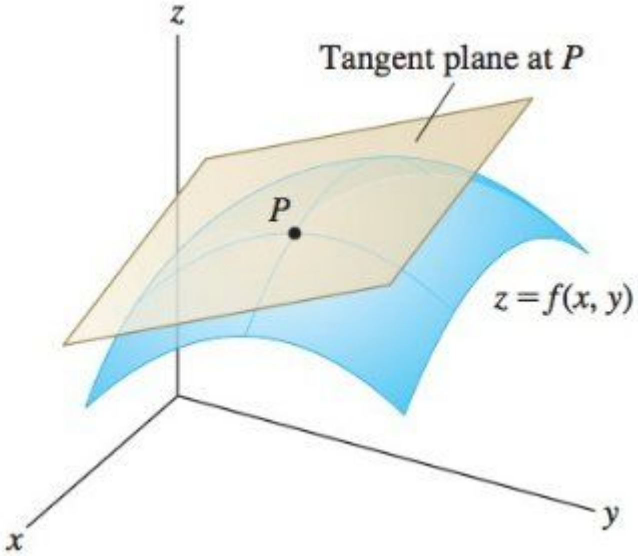
Valores máximos (o mínimos) son ***globales*** con respecto a todos los posibles valores de ***x*** (también se los llama valores extremos)



Y con más de una variable?

Con funciones multivariable podemos visualizar hasta 3 dimensiones. Después hay que usar la imaginación!





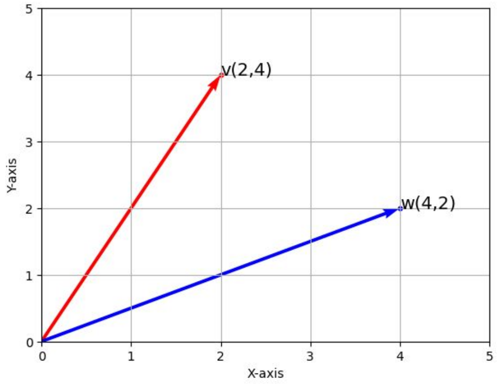
Vectores

Algo de jerga

En matemáticas (o en física), un ***vector*** es un término que se usa para referirse a cantidades que no pueden expresarse con un único número.

En ciencia de datos y en otros contextos, un ***vector*** es un término que se usa para referirse a una ***lista ordenada*** de elementos.

v	2	4
w	4	2

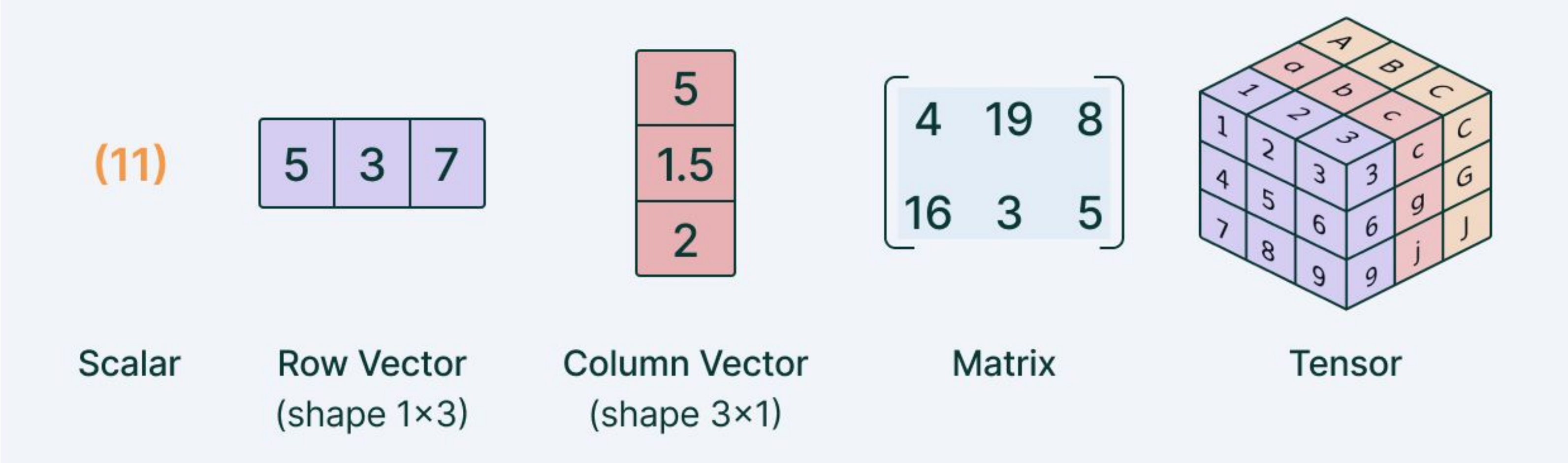


En Python ya vimos listas ordenadas de elementos:
Listas, Arrays (NumPy)

Escalares, vectores, matrices ...

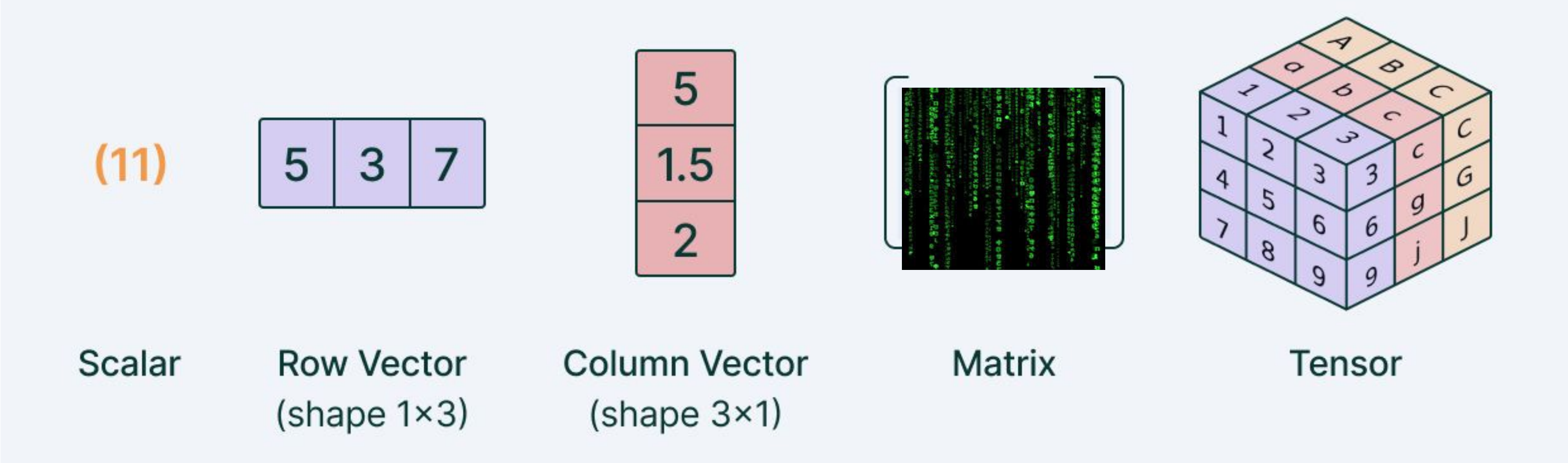
Un escalar es un valor unidimensional (ej un número).

Un vector es un conjunto ordenado de escalares (ej un array).



Una matrix es ...

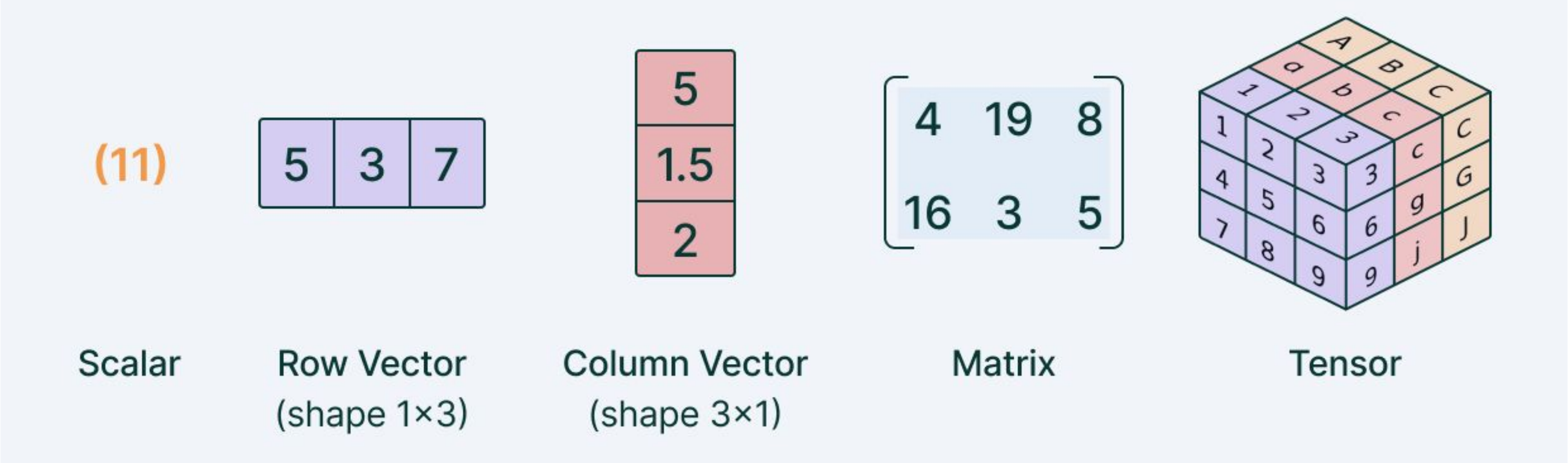
Lamentablemente nadie puede contarte qué es La Matrix. Tenés que verlo por vos mismo – Morpheus



Escalares, vectores, matrices ...

Un ***tensor*** es una generalización de los conceptos que ya vimos.

- Un escalar es un tensor de orden 0
- Un vector es un tensor de orden 1
- Una matriz es un tensor de orden 2
- ...



Operaciones con vectores y matrices

a =	1	4	7

\+

b =	2	5	8

a+b =	3	9	15

Suma (adición) de vectores

```
Vector resultado (a+b) = a1 + b1
                         a2 + b2
                         a3 + b3
```

Operaciones con vectores y matrices

a =	1	4	7

–

Resta (sustracción) de vectores

b =	2	5	8

a+b =	-1	-1	-1

```
Vector resultado (a-b) = a1 - b1
                         a2 - b2
                         a3 - b3
```

Operaciones con vectores y matrices

a =	1	4	7
		*	
b =	2	5	8
a+b =	2	20	56

Multiplicación de vectores

```
Vector resultado (a+b) = a1 * b1
                         a2 * b2
                         a3 * b3
```

Operaciones con vectores y matrices

a =	1	4	7
		/	
b =	2	5	8
a+b =	0.5	0.8	0.875

División de vectores

```
Vector resultado (a+b) = a1 / b1
                         a2 / b2
                         a3 / b3
```

Operaciones con vectores y matrices

a =	1	4	7

•

Producto escalar (dot product)

b =	2	5	8

=

a+b =	78

```
Vector resultado (a•b) = (a1 * b1)
                              +
                          (a2 * b2)
                              +
                          (a3 * b3)
```

El producto *escalar* da como resultado un valor unidimensional!

Operaciones con matrices

m

1	-1	2
0	-3	1

+

n

3	5	8
2	4	7

=

r

4	4	10
2	1	8

Suma (adición) de matrices

Matriz resultado (m+n) = $m_{11} + n_{11}$
$m_{12} + n_{12}$
$m_{13} + n_{13}$
$m_{21} + n_{21}$
$m_{22} + n_{22}$
...

Operaciones con matrices

m

1	-1	2
0	-3	1

•

v

2	1	0

=

r

1	-3

Dot product entre una matriz y un vector

Matriz resultado (m•n) = $(m_{11} * v_1) + (m_{12} * v_2) + (m_{13} * v_3)$ $\quad$ $(m_{21} * v_1) + (m_{22} * v_2) + (m_{23} * v_3)$

Operaciones con matrices

Muchas de estas operaciones ya las vimos cuando trabajamos en Python con arrays (NumPy):

- Suma, Resta, Multiplicación, División

Otras operaciones son nuevas, ej el producto escalar o ***dot product.***

- Estas son operaciones importantes en aprendizaje automático!