Funciones Matemáticas para Data Science e Inteligencia Artificial

Tema 1: Aprendamos lo elemental

Que es una funcion?

Concepto de funcion:

- Es una **regla** donde a cada elemento de un conjunto A se le **asigna** un elemento de un conjunto B.

  Una función (f) es una relación entre un conjunto dado X (llamado dominio) y otro conjunto de elementos Y (llamado codominio) de forma que a cada elemento x del dominio le corresponde un único elemento $f(x)$ del codominio (los que forman el recorrido, también llamado rango o ámbito).

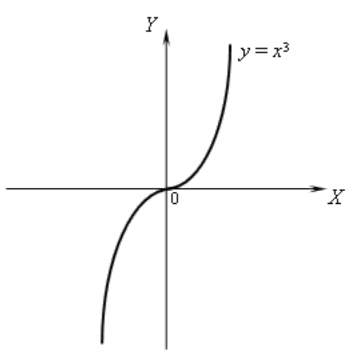
Formas de Representar una Funcion:

- Verbalmente

  "A cada letra del abecedario se le asigna un número entero diferente".
- Numéricamente

x	y
2	4.5
1	9
0	3
-1	1.5
-2	0

- Visualmente

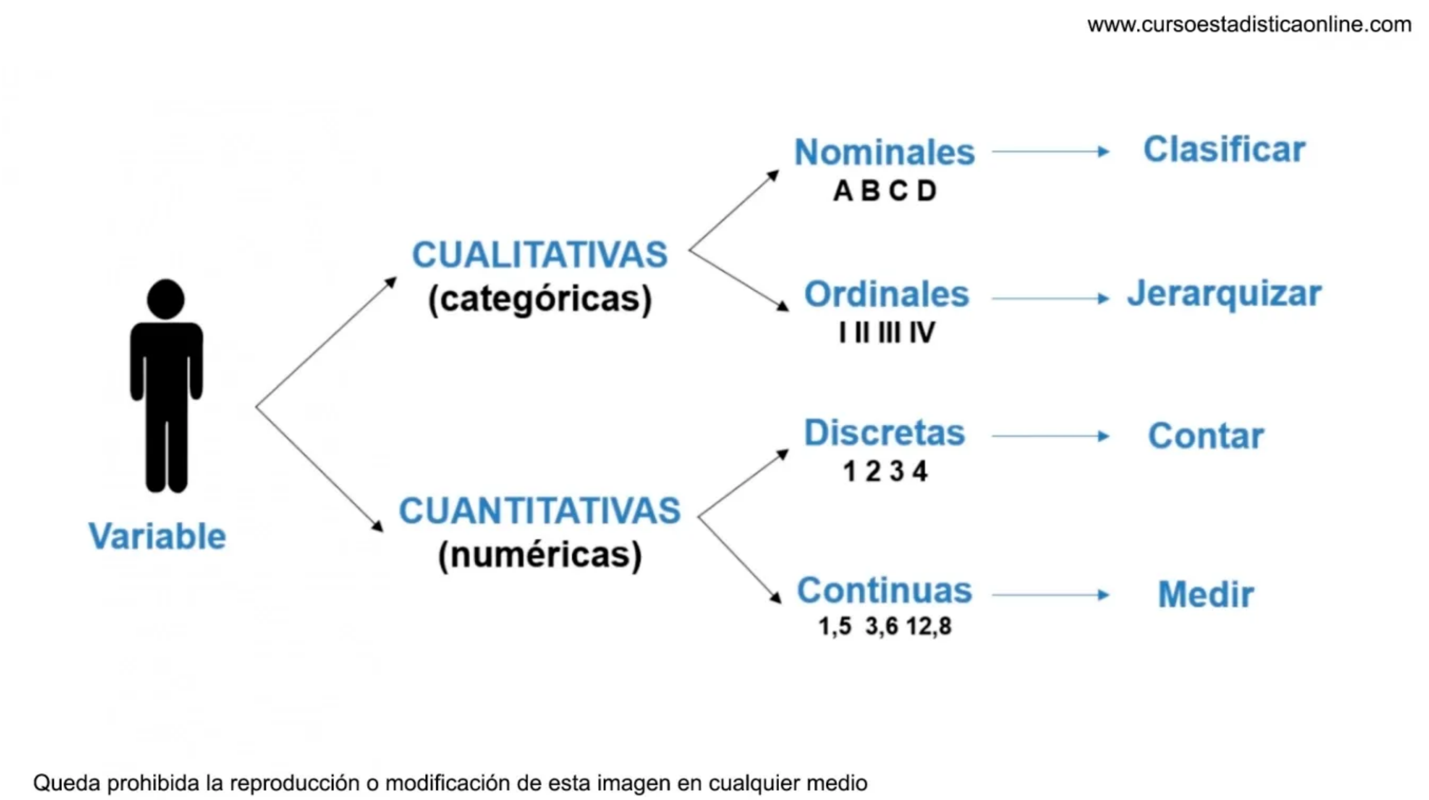


- Algebraicamente

  $y = f(x) = x^2$

Tipos de variables

Las variables se clasifican en dos grandes grupos, Variables **Cualitativas** y **Cuantitativas**



Variables Cualitativas (Categoricas)

- Nominales:

  Son las que les asignamos una cualidad. Poseen un valor intrinseco, pero no necesariamente un orden. Nos permite **clasificar objetos**.

  ```
  Ejemplo (rojo, verde, azul)
  ```

- Ordinales:

  Representan un orden. Nos permite **jerarquizar** cosas por sus valores.

  ```
  Ejemplo (alto, medio, bajo)
  ```

- Binarias:

  Solo toman dos valores, usualmente usadas para representar estados. Nos permite escoger entre un estado.

  ```
  Ejemplo (Existe o no existe, está frío o caliente, uno o cero)
  ```

Variables Cuantitativas (Numericas)

- Discretas:

  Representan un gurpo de datos finitos para tomar ciertos valores. Nos permite **contar** *"pasos"* entre variables separadas

  ```
  Ejemplo (una persona puede tener 0, 1, 2 o 3 amigos pero nunca 3.5.)
  ```

- Continuas:

  Sus valores pueden verse como infinitos al tomar cualquier vlaor dentro de los numeros reales en un rango establecido, podemos usar infinitos decimales para poder expresar sus valores. Nos permite **medir**

  ```
  Ejemplo (Medir la estatura de una persona)
  ```

Dominio y rango de una función

¿Qué valores pueden tener las funciones?

Dominio de una funcion:

- Los valores que toma x y que estan definidos en la funcion $f(x)$.

Rango de una función:

- Todos los resultados que nos puede dar una función.

Ejemplo

Tenemos una maquina para hacer cafe

- El **dominio** son los granos de cafe
- La **funcion** es nuestra cafetera
- El **rango** son todas las clases de cafe que podemos preparar

Cómo leer las matemáticas

[Tabla de símbolos matemáticos]

Tabla simbolica

Símbolos

Símbolo	Significado
$\mathbb{N}$	conjunto de los números naturales
$\mathbb{Z}$	conjunto de los números enteros
$\mathbb{Q}$	conjunto de los números racionales
$\mathbb{R}$	conjunto de los números reales
$\mathbb{C}$	conjunto de los números complejos
$\mathbb{R}^+$	conjunto de los reales positivos
$\{a,b,\ldots\}$	conjunto de elementos a,b,...
$\varnothing$	conjunto vacio
$\cap, \bigcap$	intersección de conjuntos
$\cup, \bigcup$	unión de conjuntos
$\subset$	incluido en el conjunto
$\not\subset$	no incluido en el conjunto
$\in$	pertenece a un conjunto
$\notin$	no pertenece a un conjunto
$A \setminus B,\ A-B$	conjunto diferencia
$\wp(A)$	conjunto de partes
$n(A)$	cardinal del conjunto
$A',\ \overline{A}$	conjunto complementario de A
$A\times B$	producto cartesiano
$\{x \mid x\in P\}$	todos los x que satisfacen P
$\{x:\ldots\}$	todos los x tales que ... es cierto
(a,b)	intervalo abierto
$[a,b]$	intervalo cerrado
$[a,b),(a,b]$	intervalo semiabierto
$(a,\infty),[a,\infty)$	semirrecta derecha
$(-\infty,a),(-\infty,a]$	semirrecta izquierda
$(-\infty,\infty)$	recta real

Símbolo	Significado
$=$	igual
$<$	menor que...
$\leq$	menor o igual que...
$>$	mayor que...
$\geq$	mayor o igual que...
$\neq$	distinto
$\propto$	proporcional a
$\approx$	aproximadamente igual
$\equiv$	idénticamente igual
$\pm, \mp$	más menos / menos más
$\sum$	sumatorio
$\prod$	producto
$\forall$	para todo, cuantif. universal
$\exists$	existe, cuantif. existencial
$\Rightarrow$	implica (si…entonces…)
$\Leftrightarrow$	equivale (si y solo si)
$/$	tal que
$\therefore$	por lo tanto, por consiguiente
$\because$	porque, puesto que
$\neg$	negación
$\wedge$	conjunción ("y", "además")
$\vee$	disyunción ("o")
∞	infinito
$:$	razón
$::$	proporción
$a=\dot{b}$	a es múltiplo de b
$\div$	progresión aritmética
$\div\div$	progresión geométrica

Símbolo	Significado
$n!$	factorial
$\lvert x\rvert$	valor absoluto
$\sqrt{\ }$	raíz cuadrada
$\%$	tanto por ciento
‰	tanto por mil
π	número pi, $\pi=3{,}1415\ldots$
e	número e, $e=2{,}7182\ldots$
ϕ	número fi (áureo), $\phi=1{,}6180\ldots$
$\parallel$	paralelo
$\perp$	perpendicular
$\measuredangle$	ángulo
$\binom{m}{n}$	número combinatorio
C_m^n	combinaciones
P_m	permutaciones
V_m^n	variaciones
$\Pr(A)$	probabilidad
$\Pr(A\mid B)$	probabilidad condicional
$\log$	logaritmo decimal (base 10)
$\log_a$	logaritmo de base a
$\ln$	logaritmo neperiano (base e)
$\sin\alpha$	seno de α
$\cos\alpha$	coseno de α
$\tan\alpha$	tangente de α
$\cot\alpha$	cotangente α
$\sec\alpha$	secante α
$\csc\alpha$	cosecante α
(a_n)	sucesión con término n-ésimo
Δ	incremento
σ	desviación típica
$\mathrm{Var}(X)$	varianza

Símbolo	Significado
$f', y', \frac{dy}{dx}$	derivada
$x\to c$	x tiende a c
$\lim\limits_{x\to c}$	límite cuando x tiende a c
$\int$	signo de integral
$A_{m\times n}$	matriz A de dimensión mxn
A_m	matriz cuadrada de orden m
a_{ij}	elementos a_{ij} de una matriz
$\mathrm{rang}\,A$	rango de una matriz
A^T	matriz transpuesta
A^{-1}	matriz inversa
$\lvert A\rvert, \det A$	determinante de una matriz
$f:X\to Y$	función, aplicación
$[x]$	parte entera
$\circ$	composición de funciones
f^{-1}	función inversa
$Dom\ f$	dominio de f
i	unidad imaginaria, $i^2=-1$
$\mathrm{Re}\,z$	parte real de un número complejo
$\mathrm{Im}\,z$	parte imaginaria de un complejo
$\lvert z\rvert$	módulo de un número complejo
$\overline{z}$	conjugado de un complejo
$\mathrm{Arg}\,z$	argumento de un complejo
Ox, Oy, Oz	ejes de coordenadas
$\vec{v}$	vector
$\lvert\vec{v}\rvert$	módulo de un vector
$\lVert P\rVert$	norma
$\vec{i},\vec{j},\vec{k}$	base ortonormal en un espacio
$\vec{a}\cdot\vec{b}$	producto escalar de vectores
$\vec{a}\times\vec{b}, \vec{a}\wedge\vec{b}$	producto vectorial de vectores

Math Quick Reference Card – SÍMBOLOS 1.1 – (cc) www.3con14.com

Cómo leer las matemáticas: Conjuntos

- Los **Diagramas de Venn** son otra forma de representar un conjunto de manera gráfica. Los diagramas nos permiten comprender y analizar fácilmente los conjuntos.
- Conjunto vació (ϕ)

Nota: El conjunto que contenga el elemento 0 o el ϕ es no vacío.

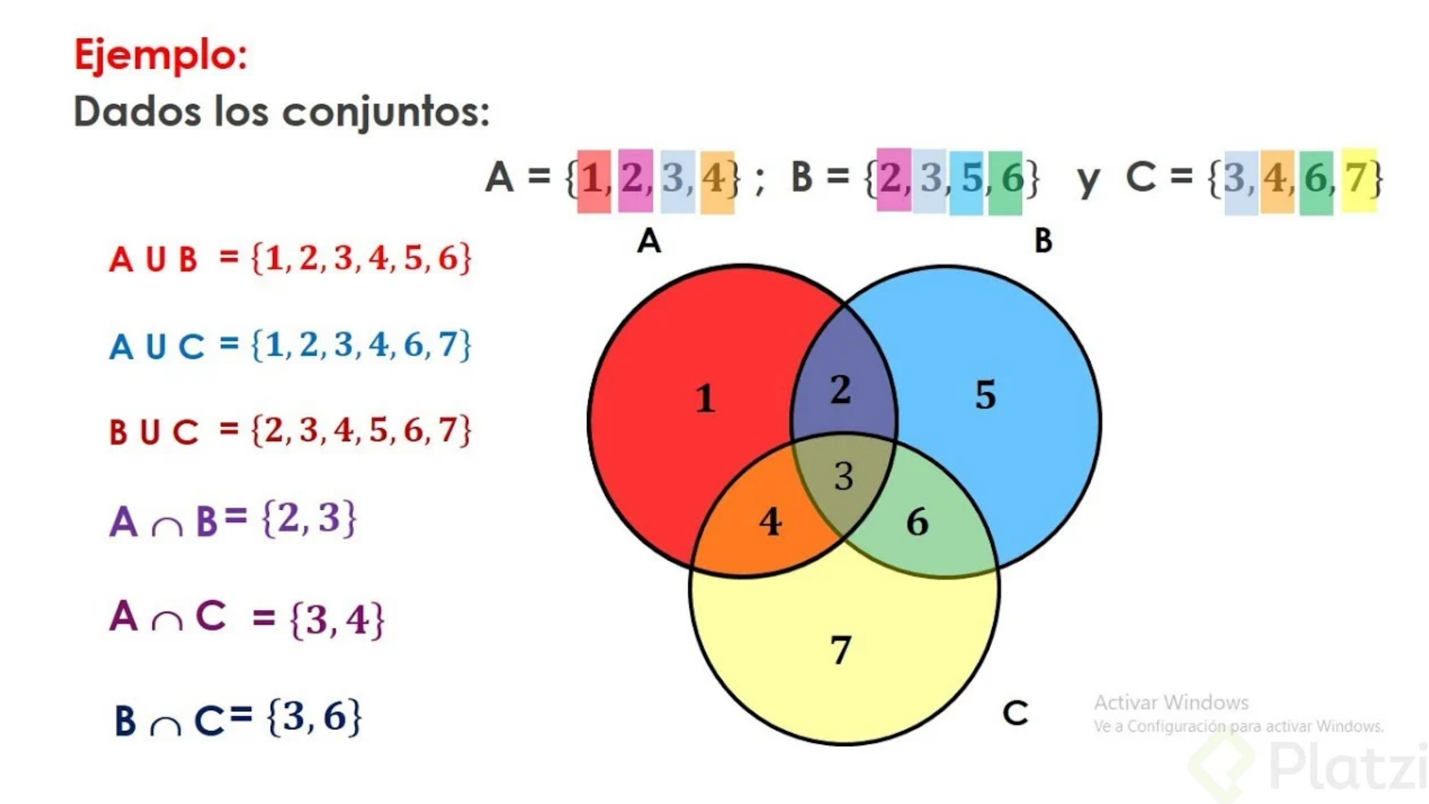
- La **unión** de dos conjuntos A y B es el conjunto formado por todos los elementos de ambos conjuntos sin repetir ninguno y se representa así: $A \cup B = \{x|x \in Aóx \in B\}$
- La **intersección** de dos conjuntos A y B son el conjunto de elementos que forman parte tanto de A como de B, es decir: $A \cap B = \{x|x \in Ayx \in B\}$

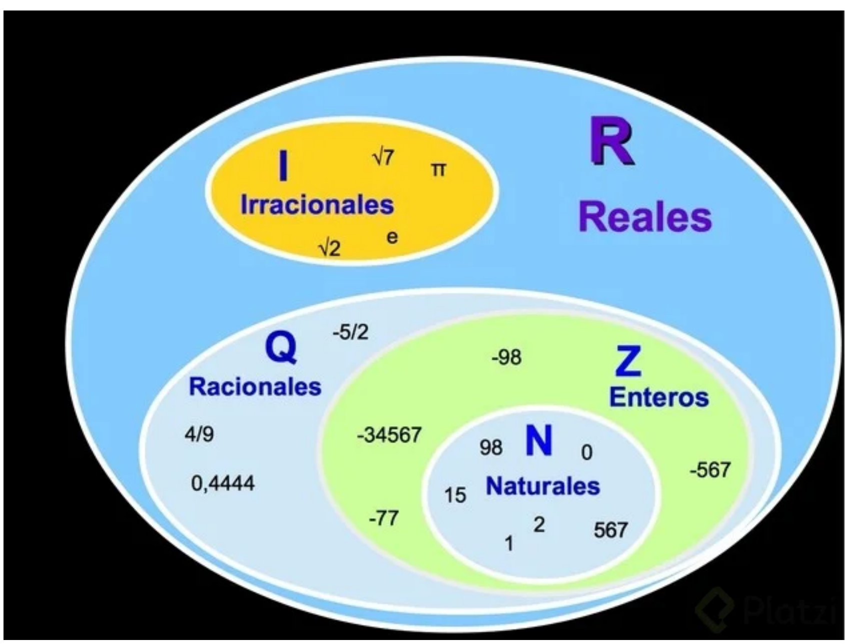
- La diferencia de dos conjuntos A y B, es un conjunto formado por todos los elementos de A que no están en B y se escribe así: $A/B = A - B = \{x \in A | x \notin B\}$
- Conjunto universo (Ω)

Es el conjunto que incluye a todos los elementos y un conjunto A **subconjunto** de Ω.

- El **complemento** de A son todos aquellos elementos del conjunto universo que no pertenecen a A, es decir: $A' = \{x \in \Omega | x \notin A\}$
- Según su número de elementos, los conjuntos se pueden clasificar como **finitos o infinitos.**
- La cardinalidad de un conjunto A es la cantidad de elementos diferentes que tiene y se representa con $|A|$

Un conjunto es finito si se conoce su cardinalidad (podemos contar sus elementos) Un conjunto es infinito si se desconoce su cardinalidad (no podemos contar sus elementos).





Datos Interesantes sobre los conjuntos

La teoria de conjuntos fue traida al mundo por George Cantor en el siglo XIX, fue una teoria revolucionaria para la matematica.

Todo elemento de la naturaleza podia ser contenido en un conjunto (una agrupación de elementos); pero mucho cuidado aquí: Eso incluye que haya un conjunto que incluya a los conjuntos, sí... ¿No te suena raro? ¿Paradojico?

Esto fue molesto para muchos matematicos, pues, debian exitir conjuntos infinitos! ¿Como podria haber algo inifinito?

Ahí inicia la guerra de las matematicas: Dos grandes bandos Intuicionistas y formalistas

Los intuicionistas satanizaban al pobre Cantor, Kronecker dijo:

Cantor es un cientifico charlatan y corruptor de la juventud

convencidos de que la matematica era una creacion pura de la mente humano y que los infinitos no eran reales. El mismo Poincare dijo:

***Las futuras generaciones consideraran a la teoria de los conjuntos una enfermedad de la que se habran recuperado ***

Los formalistas si creian que las matematicas podian ser colocadas de una manera segura a travez de la teoria de los conjuntos, Hilbert, la leyenda viva dijo:

Nadie nos expulsará del paraiso que Cantor creó para nosotros

En 1901 empiezan los problemas, ustedes ya saben cual es, formalicemos: Existe un conjunto R, este conjunto es el conjunto de todos los conjuntos que no se contienen a si mismos, si R no se contiene a si mismo, entonces el conjunto contiene a R, si R no se contiene a si mismo entonces debera contenerse a si mismo, pero si R si se contiene a si mismo, entonces por definicion deberá contenerse a si mismo, entonces R se contiene si y solo si no se contiene a si mismo A eso mis amig@s le llamamos la paradoja de autorreferencia

La paradoja del barbero

Tema 2: Todo sobre funciones

Funciones algebraicas

```
import matplotlib.pyplot as plt
import numpy as np
```

Función lineal

Tiene la forma de

$$f(x) = mx + b$$

Donde m y $b \in R$.

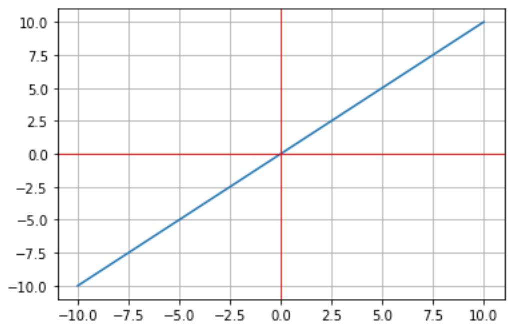
m puede ser calculada por:

$$m = \frac{y_2 - y_1}{x_2 - x_1}$$

- m se le conoce como **pendiente de la recta**.
- Y b es **el punto de corte** con el eje y.
- Su **dominio** es $Dom_f = (-\infty, \infty)$.
- Su **imagen** es $Im_f = (-\infty, \infty)$

```
# Ejemplo Funcion Lineal
def graficar(x,y):
    plt.plot(x,y)
    plt.grid()
    plt.axhline(y=0, color='r', linewidth=0.8)
    plt.axvline(x=0, color='r', linewidth=0.8)
    plt.show()


N = 100 # N es la resolucion de una funcion

m = 1 # Pendiente de la recta

b = 0 # Punto de corte con el eje Y
```

```
# Funcion lineal de primer grado
def f(x):
    return m*x+b

x = np.linspace(-10,10, num=N)

y = f(x)

graficar(x,y)
```



Funciones polinómicas

Tiene la forma de

$$P(x) = a_n x^n + a_{n-1} x^{n-1} + ... + a_2 x^2 + a_1 x + a_1$$

a una función que tiene esta forma se le llama **polinomio de grado** n. A los elementos a los llamaremos **coeficientes** donde $a \in R$.

Por ejemplo:

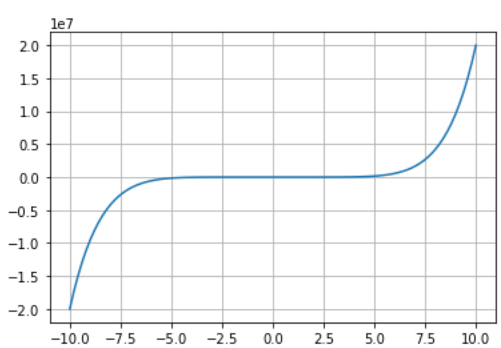
$$P(x) = 2x^7 - x^4 + 3x^2 + 4$$

$$P(x) = 2x^7 + 0x^6 + 0x^5 - x^4 + 0x^3 + 3x^2 + 0x + 4$$

que es un polinomio de grado 7.

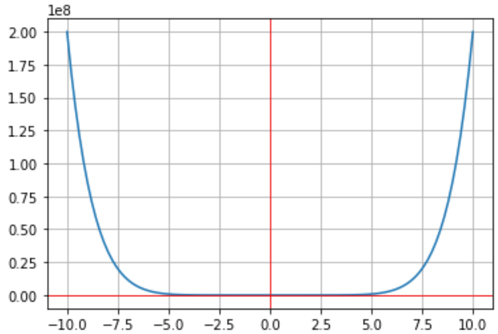
Ejemplos en codigo:

```
# Polinomio Grado IMPAR
def f(x):
    return (2*x**7)-(x**4)+(3*x**2)+4

y = f(x)

plt.plot(x,y)
plt.grid()
```



```
# Polinomio Grado PAR
def f(x):
    return (2*x**8)-(x**4)+(3*x**2)+4

y = f(x)
```

```
graficar(x,y)
```



Funciones potencia

Hay unas funciones que son un caso particular de las funciones polinómicas que son las funciones potencia, las cuales tienen la forma:

$$f(x) = x^a, a \in R$$

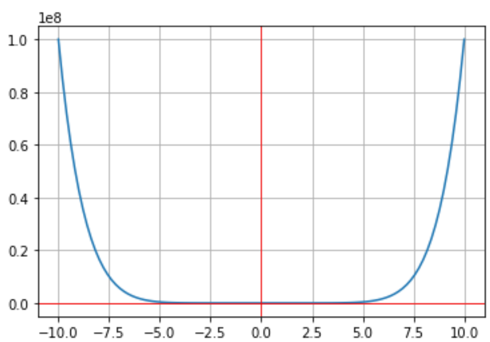
Por ejemplo:

$$f(x) = x^2$$

- El dominio de $f(x) = x^2$ es $Dom_f = (-\infty, \infty)$.
- Su imagen es $Im_f = [0, \infty)$

Ejemplos en codigo:

```
# Ejemplo Funcion Potencia

def f(x):
    return x**8

y = f(x)

graficar(x,y)
```



Funciones trascendentes

Son funciones que no pueden ser expresadas con polinomios.

Se consideran como funciones trascendentes a las exponenciales, logarítmicas, trigonométricas y trigonométricas inversas.

[Mas info]

[Notas Funciones Exponenciales y Logaritmicas]

Funciones trigonométricas

Algunos ejemplos son las funciones $cos(x)$, $sen(x)$ y $tan(x)$

Mas sobre funciones trigonométricas

Las funciones trigonométricas puede verse de dos maneras:

- Una manera es verlas como funciones de números reales;
- la otra es verlas como funciones de ángulos.

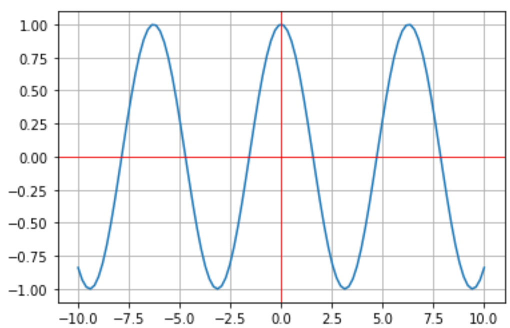
Aplicaciones

Las aplicaciones de la trigonometría son numerosas, incluyendo el **procesamiento de señales**, la **codificación digital** de **música** y **video**, para **encontrar las distancias a las estrellas**, producir **tomografías** para imágenes médicas y muchas otras.

Estas aplicaciones son muy diversas y necesitamos estudiar los dos métodos trigonométricos ya que ambos métodos son necesarios para las diferentes aplicaciones.

Una de las principales aplicaciones de la trigonometría es el **movimiento periódico**. Si usted se ha subido a una rueda de la fortuna conoce el movimiento periódico, es decir, un movimiento que se repite una y otra vez. El movimiento periódico es común en la naturaleza. Considere el diario amanecer y puesta de Sol, la variación diaria de los niveles de las mareas o las vibraciones de una hoja en el viento, y muchas otras situaciones.

```
# Ejemplo Funcion Trigonometrica
def f(x):
    return np.cos(x)

y = f(x)

graficar(x,y)
```



Función exponencial

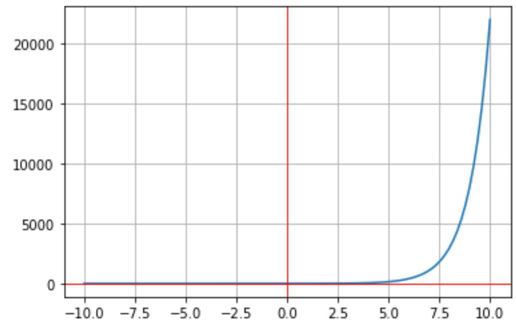
Tienen la forma de

$$f(x) = a^x$$

donde la base a es una constante positiva. Un gran ejemplo de una función exponencial es usando la base como el número de euler:

$$f(x) = e^x$$

```
def f(x):
    return np.exp(x)

y = f(x)

graficar(x,y)
```



Función logaritmo

El logaritmo está definido por la **relación**:

$$log_b(x) = n \Longleftrightarrow x = b^n$$

donde:

- b es la **base**.
- n es el **exponente** al que está elevado la base.
- x es el **resultado** de elevar la base b al exponente n

Informacion sobre logaritmos

El logaritmo es:

"La potencia a la que debo elevar la base, para obtener el número dentro del paréntesis"

$$log3(27) = 3$$

Cuando no hay número de abajo se asume que es el 10, asi entonces:

$$log(100) = 2$$

Ya que $10 * 10 = 100$

Y finalmente, si aparece ln es porque consideramos a la base el numero de Euler 2.71... ,

$$ln(7.38) = 2$$

porque $e * e = 7.38$.

Ejemplo:

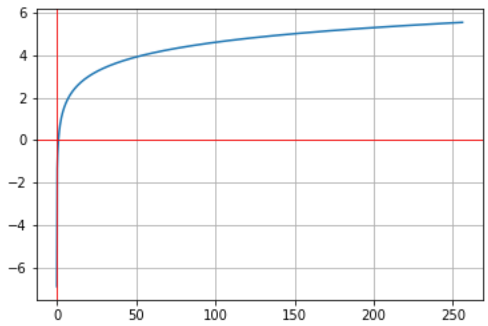
Teniendo b=2 y n=8, entonces:

$$2^8 = 256$$

Por lo que $x = 256$. Calculando el logaritmo base 2 de x es:

$$log_2(256) = 8$$

```
# Ejemplo Funciones Logaritmicas
def f(x):
    return np.log(x)

x = np.linspace(0.001, 256, num=1000)

y = f(x)
graficar(x,y)
```

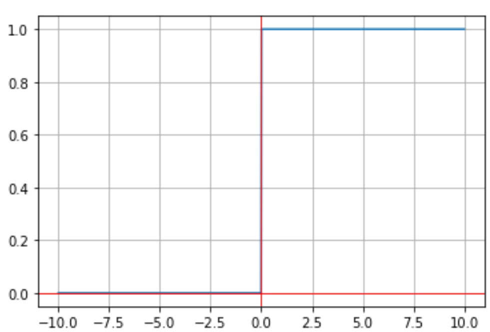


Funciones seccionadas

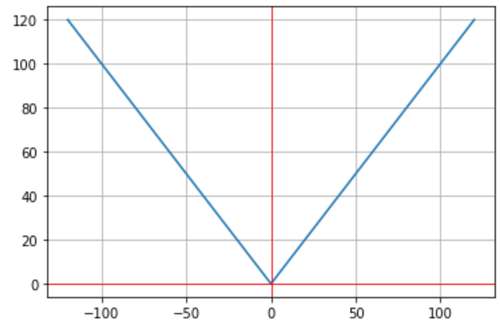
Son funciones que tienen diferentes valores definidos por un intervalo. Por ejemplo la función escalón de Heaviside:

$$\begin{cases} 0, & \text{para, } x < 0 \\ 1, & \text{para. } x \geq 0 \end{cases}$$

```
# Ejemplo Escalon de Heaviside
def H(x):
    Y = np.zeros(len(x))

    for idx,x in enumerate(x):
        if x >= 0:
            Y[idx] = 1
    return Y

N = 1000
x = np.linspace(-10,10,N)

y = H(x)

graficar(x,y)
```



```
# Ejemplo Valor Absoluto
def A(x):
    Y = np.zeros(len(x))

    for idx,x in enumerate(x):
        if x >= 0:
            Y[idx] = x
        else:
            Y[idx] = x*(-1)

    return Y

N = 1000
x = np.linspace(-120,120,N)

y = A(x)

graficar(x,y)
```



Funciones Compuestas

[Info extra]

Definición

Se define la **función compuesta** de dos funciones $f(x)$ y $g(x)$ cualesquiera, y designada por, a la función que transforma x en $g[f(x)]$:

$$x \xrightarrow{f} f(x) \xrightarrow{g} g[f(x)] = (gof)(x)$$

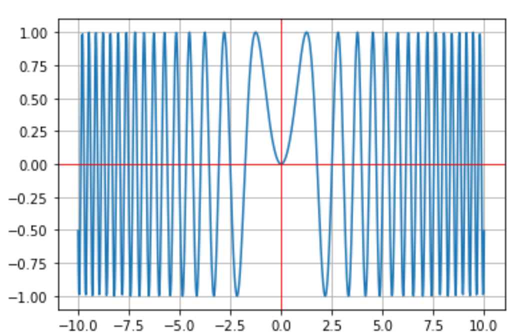
Donde:

: Se lee f **compuesta** con g. Es la propia función compuesta que permite transformar directamente x en $g[f(x)]$

Ejemplo Clase

$$x \xrightarrow{g} g(x) \xrightarrow{f} f[g(x)] = (fog)(x)$$

```
# f(g(x))
N = 1000
x = np.linspace(-10,10,num=N)

def g(x):
    return x**2

def f(x):
    return np.sin(x)

y = g(x)

f_o_g = (f(g(x)))
# f_o_g
graficar(x,f_o_g)
```



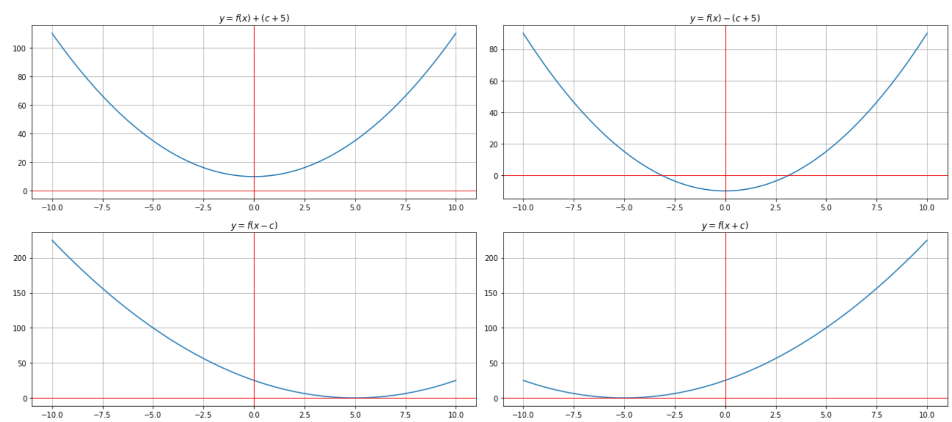
¿Cómo manipular funciones?

Desplazamientos verticales y horizontales

Siendo c una **constante** mayor que cero, entonces la gráfica:

- $y = f(x) + c$ se desplaza c unidades hacia arriba.
- $y = f(x) - c$ se desplaza c unidades hacia abajo.
- $y = f(x - c)$ se desplaza c unidades hacia la derecha.
- $y = f(x + c)$ se desplaza c unidades hacia la izquierda.

```
# Desplazamientos verticales y horizontales
N = 100

def f(x):
    return x**2

c = 5


x = np.linspace(-10,10, num=N)

y2 = f(x) + (c+5)
y3 = f(x) - (c+5)
y4 = f(x - c)
y5 = f(x + c)

fig, ax = plt.subplots(2,2, figsize=(18,8))

def graficar2(pos,y,title):
    ax[pos].plot(x,y)
    ax[pos].set_title(title)
    ax[pos].grid()
    ax[pos].axhline(y=0, color='r', linewidth=0.8)
    ax[pos].axvline(x=0, color='r', linewidth=0.8)

graficar2((0,0),y2,'$y = f(x) + (c+5)$')
graficar2((0,1),y3,'$y = f(x) - (c+5)$')
graficar2((1,0),y4,'$y = f(x - c)$')
graficar2((1,1),y5,'$y = f(x + c)$')
plt.tight_layout()
```



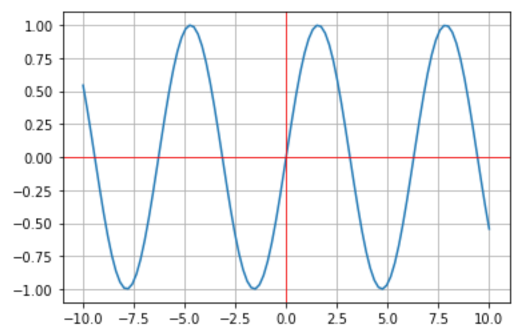
Alargamientos y compresiones

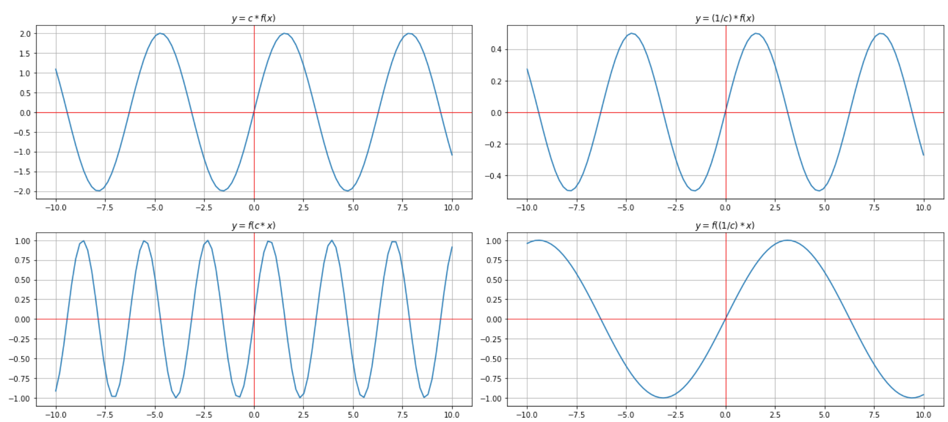
Siendo c una constante mayor que cero, entonces la gráfica:

- $y = c \cdot f(x)$ alarga la gráfica verticalmente en un factor de c.
- $y = \frac{1}{c} \cdot f(x)$ comprime la gráfica verticalmente en un factor de c.
- $y = f(c \cdot x)$ comprime la gráfica horizontelmente en un factor de c.
- $y = f(\frac{1}{c} \cdot x)$ alarga la gráfica horizontelmente en un factor de c.

```
# Ejemplos Alargamientos y Compresion
N = 1000

def f(x):
    return np.sin(x)

c = 2
```

```
y = f(x)
y1 = c*f(x)
y2 = (1/c) * f(x)
y3 = f(c * x)
y4 = f((1/c) * x)

graficar(x,y) # Funcion Original


fig, ax = plt.subplots(2,2, figsize=(18,8))

# graficar2(())
graficar2((0,0),y1,'$y = c*f(x)$')
graficar2((0,1),y2,'$y = (1/c) * f(x)$')
graficar2((1,0),y3,'$y = f(c * x)$')
graficar2((1,1),y4,'$y = f((1/c) * x)$')
plt.tight_layout()
```

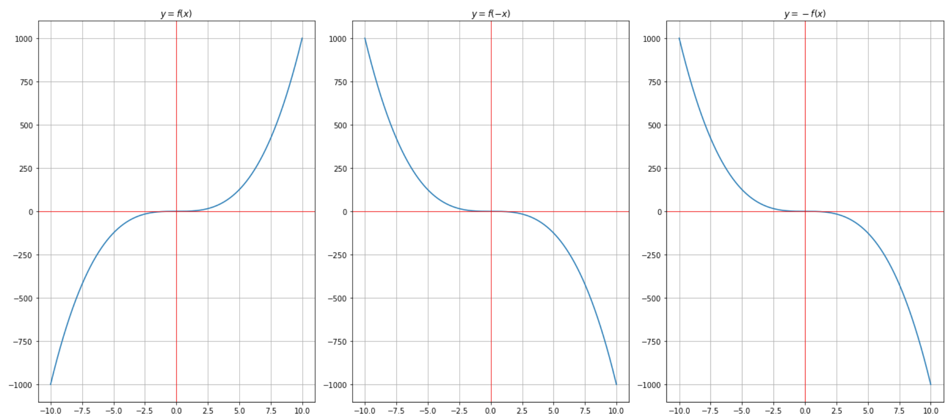




Reflexiones

- $y = -f(x)$ refleja la gráfica respecto al eje x.
- $y = f(-x)$ refleja la gráfica respecto al eje y.

```
# Ejemplos Reflexion
N = 1000

def f(x):
  return x**3

x = np.linspace(-10,10, num=N)

y1 = f(x)
y2 = -f(x)
y3 = f(-x)

fig, ax = plt.subplots(1,3, figsize=(18,8))
graficar2((0),y1,'$y=f(x)$')
graficar2((1),y2,'$y=f(-x)$')
graficar2((2),y3,'$y=-f(x)$')
fig.tight_layout()
```



Características de las funciones

¿Por qué se llaman funciones reales?

Se llaman funciones reales porque tanto su **dominio** como el **codominio** (recuerda que al **codominio también se llama rango o imagen**) están contenidos en el conjunto de los **números reales**. Es decir, el **conjunto que contiene** a los **números racionales e irracionales**. En otras palabras cualquier número que no sea imaginario.

Función par

Una función es par si cumple la siguiente relación a lo largo de su dominio:

$$f(-x) = f(x)$$

Si lo notaste, esta relación nos dice que una función es par si es simétrica al eje vertical (eje Y). Por ejemplo, una parábola es una función es par.

Función impar

Una función es impar si cumple la siguiente relación a lo largo de su dominio:

$$f(-x) = -f(x)$$

Desde un punto de vista geométrico, una función impar posee una simetría rotacional con respecto al origen de coordenadas, lo que quiere decir que su gráfica no se altera luego de una rotación de 180 grados alrededor del origen.

Función acotada

Una función es acotada si su codominio (también conocido como rango o imagen) se encuentra entre dos valores, es decir, está acotado. Esta definición se define como que hay un número m que para todo valor del dominio de la función se cumple que:

$$-m \leq f(x) \leq m$$

Video Explicativo funciones acotadas

Por ejemplo, la función seno o coseno están acotadas en el intervalo [-1, 1] dentro de su co-dominio.

Funciones monótonas

Estas funciones son útiles de reconocer o analizar debido a que nos permiten saber si una función crece o decrece en alguno de sus intervalos. Que algo sea monótono significa que no tiene variaciones. Entonces las funciones monótonas son aquellas que dentro de un intervalo I, perteneciente a los números reales, cumple alguna de estas propiedades:

1. La función es monótona y estrictamente creciente:

$$\text{si para todo } x_1, x_2 \in I : x_1 < x_2 \Leftrightarrow f(x_1) < f(x_2)$$

La forma correcta de leer esto es "**si para todo x1 y x2 que pertenecen al intervalo I, tal que x1 sea menor a x2, si y solo si f(x1) sea menor a f(x2)**". En palabras mucho más sencillas, lo que nos dice esta definición es que x1 siempre tiene que ser menor que x2 en nuestro intervalo I, y que al evaluar x2 en la función el resultado de esto siempre será mayor que si evaluamos la función en x1. Para las siguientes tres definiciones restantes no cambia mucho la forma en la que se interpretan.

2. La función es monótona y estrictamente decreciente:

$$\text{si para todo } x_1, x_2 \in I : x_1 < x_2 \Leftrightarrow f(x_1) > f(x_2)$$

3. La función es monótona y creciente:

$$\text{si para todo } x_1, x_2 \in I : x_1 < x_2 \Leftrightarrow f(x_1) \leq f(x_2)$$

4. La función es monótona y decreciente:

$$\text{si para todo } x_1, x_2 \in I : x_1 < x_2 \Leftrightarrow f(x_1) \geq f(x_2)$$

Funciones periódicas

Las funciones periódicas son aquellas que se repiten cada cierto periodo, este periodo se denomina con la letra T. La relación que debe cumplir la función para ser periódica es la siguiente.

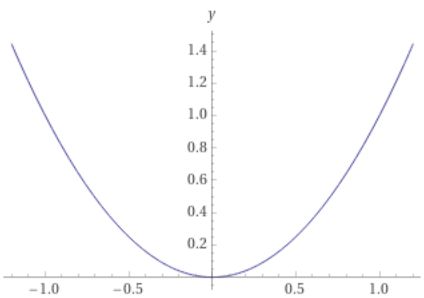
$$f(x) = f(x+T), T \neq 0$$

Por ejemplo, la función seno y coseno son funciones periódicas con un periodo $T = 2\pi$. Es decir que si nosotros calculamos $f(x)$ y calculamos $f(x+2\pi)$ en la función seno el valor que nos den ambas expresiones es el mismo.

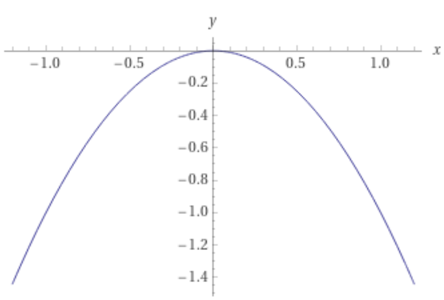
Funciones cóncavas y convexas

La forma de demostrar la concavidad de una función se puede hacer a través del análisis de derivadas consecutivas, pero aún no llegamos a eso, así que no te preocupes. A continuación, te dejo una forma súper intuitiva de ver si una función es cóncava o convexa.

Se dice que una función dentro de un intervalo es convexa si la función "abre hacia arriba". Es decir si se ve la siguiente manera:



Ahora, ¿qué sería una función cóncava? Pues así es, lo contrario de una convexa. Se dice que una función dentro de un intervalo es concava si la función "abre hacia abajo". Es decir si se ve la siguiente manera:



Reto

Como ves, identificar si una función es cóncava o convexa a través de su gráfica es muy sencillo. Pero si quieres enfrentar un reto, te dejo que cuando sepas que es una derivada regreses a esta clase y busques la forma de comprobar si una función es cóncava o convexa de forma analítica. Como pista, se hace a través del análisis de la segunda derivada.

Criterios de concavidad y convexidad

Sea $f(x)$ una función dos veces derivable en $x = x_i$, podemos determinar su curvatura a partir de los siguientes criterios:

- Sí $f''(xi) < 0$, entonces la función es cóncava en xi
- Sí $f''(xi) > 0$, entonces la función es convexa en xi

Funcion Ejemplo

$f(x) = 2x^3 + x^2 - 3x$

Regla de derivadas usadas

Derivada de x^n

$$x^n = \frac{dx}{dx} = n * x^{n-1}$$

Derivada de n

$$n = \frac{dy}{dx} = 0$$

Resolucion

Hacemos la derivada doble de la funcion dada:

$$f(x) = 2x^3 + x^2 - 3x$$

$$f'(x) = 3 * 2x^{3-1} + 2 * 1x^{2-1} - 1 * 3x^{1-1}$$

$$f'(x) = 6x^2 + 2x^1 - 3x^0$$

$$f'(x) = 6x^2 + 2x^1 - 3 * 1$$

$$f''(x) = 2 * 6x^{2-1} + 2 * x^{1-1} - 3$$

$$f''(x) = 12x^1 + 2 * x^0 - 0$$

$$f''(x) = 12x + 2 * 1$$

$$f''(x) = 12x + 2$$

Igualamos la ecuacion a 0 para encontrar el punto de inflexion:

$$12x + 2 = 0$$

$$12x + 2 - 2 = 0 - 2$$

$$12x = -2$$

$$\frac{12x}{12} = \frac{-2}{12}$$

$$x = -\frac{1}{6}$$

Los intervalos conformados son:

$$(-\infty, -\frac{1}{6})$$

$$(-\frac{1}{6}, +\infty)$$

Para corroborar si el intervalo es concavo o convexo, tenemos que ver el signo de la derivada doble de la funcion:

- Si segunda derivada es positiva: la funcion es **convexa**
- Si segunda derivada es negativa: la funcion es **concava**

Elegimos un numero que respete los intervalos que conforman la funcion $f''(x) = 12x + 2$:

$$(-\infty, -\frac{1}{6}) \longrightarrow f''(-\frac{1}{7}) = 12 * (-\frac{1}{7}) + 2 > 0 \longrightarrow \text{el intervalo es convexo}$$

$$(-\frac{1}{6}, +\infty) \longrightarrow f''(-\frac{1}{5}) = 12 * (-\frac{1}{5}) + 2 < 0 \longrightarrow \text{el intervalo es concavo}$$

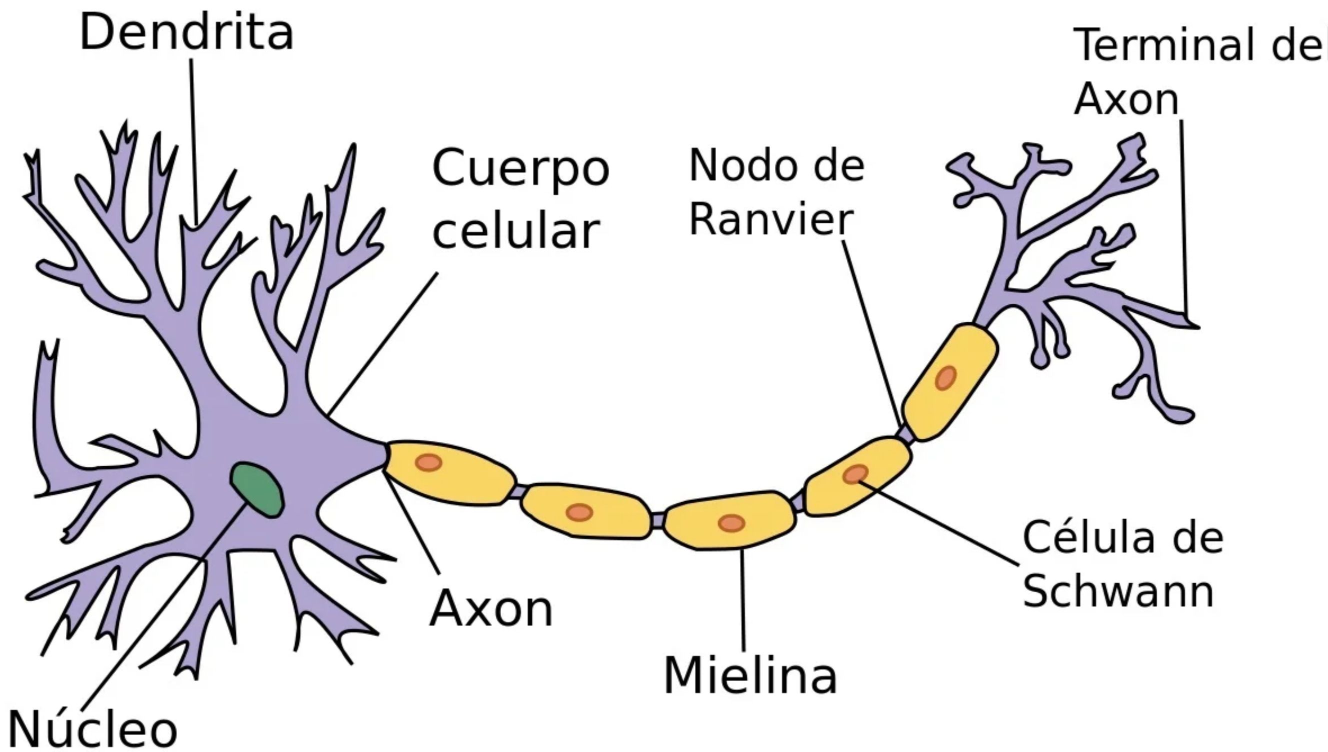
Tema 3: Funciones en ciencia de datos

Conoce al perceptrón

Que es un perceptron?

Es un algoritmo de aprensidazje supervisado para clasificadores binarios (0 - 1). Este algoritmo permite a alas neuronas aprender y procesar elementos en el conjunto de entrenamiento uno a la vez.

Antes de iniciar a explicar el concepto de como esta modelada una **Neurona Artificial**, explicaremos como funciona una neurona biologica de una forma muy sencilla (Dendrita, Axon y Terminal de Axon)

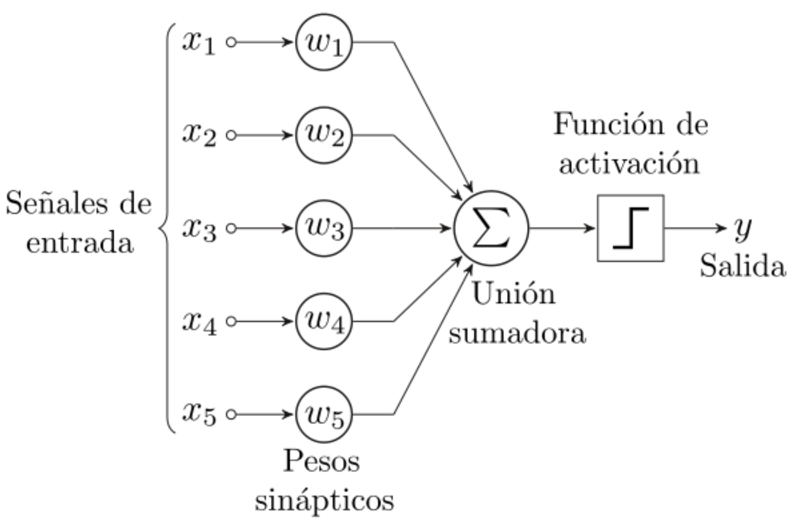


Estructura de una neurona

Empezamos las **Dendritas**, que son mecanismos receptores de impulsos electricos, el **Axon** recoge todos los impulsos que electricos que pasan a traves del nucleo, recorriendo a travez del axon, la mielena, surgiendo diferentes procesos a lo largo de este recorrido, acabando en la **Terminal del Axon**.

Podemos compararlo con una *funcion*, donde entran varios impulsos (valores de entrada), sucede un proceso (la propia funcion), y al final tenemos un mecanismo de salido con el resultado de los procesos dentro de la funcion.

Si queremos modelar esto de una forma matematica, podriamos hacerlo de esta forma.



Tenemos diferentes **Señales de entrada**, **Pesos sinapticos**, **Union sumadora**, **Funcion de activacion** que nos resulta en una **Salida**

[Video Expicativo]

Señal de entrada

Basicamente en la neurona biologica lo podriamos comparar como un impulso electrico, pero con las neuronas artificiales interpretarian caracteristicas o *Feactures*. Por ejemplo **un producto comercial** podria tener como

- x_1 : El color,
- x_2 : El tamano,
- x_3 : El precio,
- x_4 : La calidad,
- x_5 : El fabricante

Los valores del las señales son 1 o 0

Peso sináptico

Son numeros encargados de ponderan el valor de la señal de entrada, le ponen un valor de que tan importante es la señal de entrada que acaba de entrar. Por ejemplo:

- $w_1 : 1$
- $w_2 : 10$
- $w_3 : 100$
- $w_4 : 1000$
- $w_5 : 10000$

Unión sumadora

Suma todos los valores de las Señales de entradas multiplicadas por sus respectivos Pesos sinápticos, y generamos una combinación lineal

$$\sum_{i=1}^{n} w_i * x_i$$

Función de activación

[Video Explicativo]

u es el 'umbral', el cual representa el grado de inhibición de la neurona, es un término constante que no depende del valor que tome la entrada.

El valor de $f(x)$ (0 o 1) se usa para clasificar x como un caso positivo o un caso negativo, en el caso de un problema de clasificación binario. El umbral puede entenderse como una manera de compensar la función de activación, o una forma de fijar un nivel mínimo de actividad a la neurona para considerarse como activa. La suma ponderada de las entradas debe producir un valor mayor que u para cambiar la neurona de estado 0 a 1.

$$f(x) = \begin{cases} 1 & \text{si } w * x - u > 0 \\ 0 & \text{en otro caso} \end{cases}$$

Una sola neurona no tiene sentido de. Su labor especializada toma valor en la medida en que se asocia a otras neuronas, formando una red. Normalmente, el axón de una neurona entrega su información como "señal de entrada" a una dendrita de otra neurona y así sucesivamente. El perceptrón que capta la señal en adelante se extiende formando una red de neuronas, sean éstas biológicas o de sustrato semiconductor (compuertas lógicas).

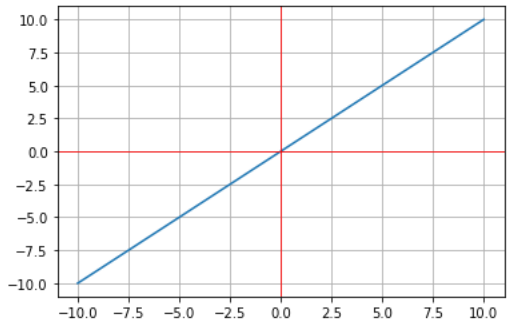
Funciones de activación

[Lectura sobre Funciones de activación]

Función lineal

Aprovechamos la funcion lineal como Funcion de activacion cuando queremos mantener valores a lo largo de un proceso, como predecir el valor de una venta. $y = mx + b$

```
#Funcion Lineal
def f(x):
    return x

y = x
graficar(x,y)
```

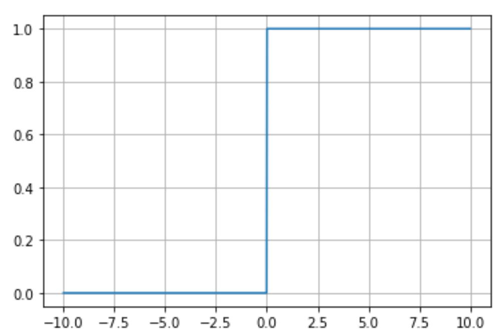


Función escalón o de Heaviside

Podriamos usar la Funcion de escalon de Heaviside para hacer **clasificaciones categóricas** en modelos que solo tenemos 2 valores, como por ejemplo estados binarios.

$$H(x) = \begin{cases} 0, & \text{para, } x < 0 \\ 1, & \text{para. } x \geq 0 \end{cases}$$

```
# Función escalón o de Heaviside

def H(x):
  Y = np.zeros(len(x))
  for idx,x in enumerate(x):
    if x>=0:
      Y[idx]=1
  return Y

y = H(x)

plt.plot(x,y)
plt.grid()
```

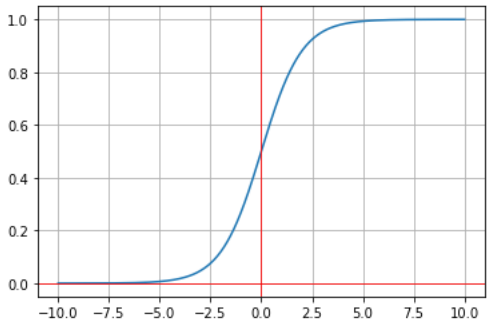


Función sigmoide

para el modelo de regresión logística, para probabilidades, ya que su rango va de 0 a 1.

$$f(x) = \frac{1}{1 + e^{-x}}$$

```
# Función sigmoide

def f(x):
    return 1/(1 + np.exp(-x))

y = f(x)

graficar(x,y)
```

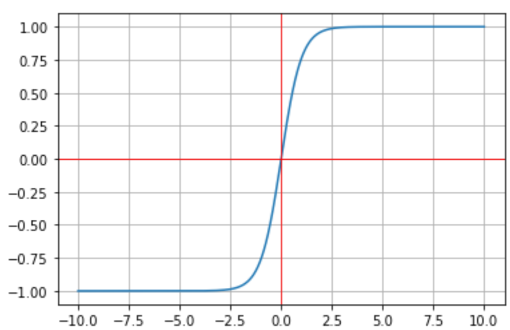


Función tangente hiperbólica

Llega a ser más exacta que la función sigmoide. Permite más posibilidades. Ya que sirve para ver pequeñas variaciones.

$$f(x) = \frac{2}{1+e^{-2x}} - 1$$

```
def f(x):
  return np.tanh(x)


N=1000

y = f(x)

graficar(x,y)
```



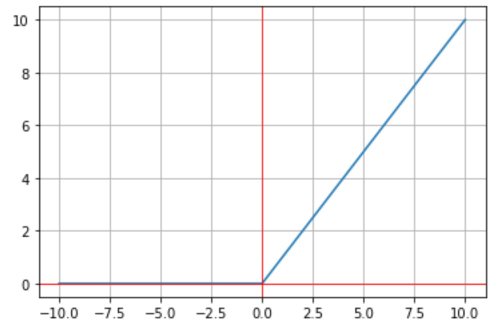
Función ReLU

Mantiene un limite entra el cero y números positivos. Es util para filtrar valores nulos. Sirve para hacer un seguimiento de perceptrones muertos.

[Diferencias entre ELU, ReLU, LeakyRelu]

$$R(x) = max(0, x)$$

```
def f(x):
  return np.maximum(x,0)


N=1000

y = f(x)

graficar(x,y)
```



Tema 4: Modela tu primer función

Enunciado

Estas trabajando en una compania como Cientifico/a de Datos, llega tu gerente y te da la siguiente informacion.

En el ultimo mes se ha gastado X de publicidad, y obtuvieron Y de retorno en ventas.

Se te pide que elabores un modelo o funcion que te trate de predecir, cuanto se va a vender dependiendo de cuanto se vaya a gastar.

🔔Publicidad🔔	🤑Ganancias🤑
1.200000	2.000000
2.000000	3.000000
3.200000	3.400000
2.500000	3.100000
5.000000	4.000000
6.000000	4.700000
4.000000	3.800000
8.000000	7.000000

Resolucion

[Video Informativo]

La regresión lineal permite explicar la relación de una variable dependiente (y), con respecto a otras variables independiente (x). Permite explicar como se afecta la variable dependiente (y) por los cambios que tenga la variable independiente (x).

$$m = \frac{n \sum xy - n \sum x * n \sum y}{n \sum x^2 - n \sum x^2}$$

$$b = \frac{\sum y - a * \sum x}{n}$$

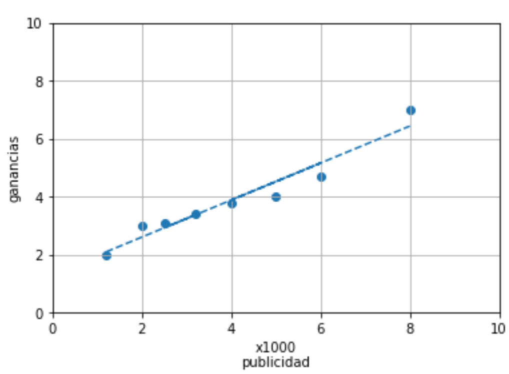
$$y = m * x + b$$

```
import numpy as np
import matplotlib.pyplot as plt

x = np.array([1.2, 2, 3.2, 2.5, 5, 6, 4, 8])
y = np.array([2, 3, 3.4, 3.1, 4, 4.7, 3.8, 7])
n = len(x)

xy = np.sum((x*y),axis=0)

x_2 = np.sum((x**2), axis=0)

sum_x = np.sum(x, axis=0)
sum_y = np.sum(y, axis=0)
```

```
def m():
    numerador = (n*xy) - (sum_x * sum_y)
    denominador = (n*x_2) - (sum_x**2)
    return np.divide(numerador,denominador)
m = m()

def b():
    numerador = sum_y - (m * sum_x)
    denominador = n
    return np.divide(numerador, denominador)
b = b()

def f():
    return (m*x)+b

y_prima = f()

df = pd.DataFrame.from_dict({
    "xy"      :xy,
    "x_2"     :x_2,
    "sum_x"   :sum_x,
    "sum_y"   :sum_y,
    "m"       :m,
    "b"       :b,
    "y_prima" :y_prima,
},)
df.head(1)
```

	xy float64	x_2 float64	sum_x float64	sum_y float64	m float64	b float64	y_prima float64
0	146.43	162.93	31.9	31	0.6386313543015079	1.3284574747227373	2.0948150998845465

```
# Graficar
plt.scatter(x,y)
plt.plot(x,y_prima, '--')

plt.xlabel('x1000\npublicidad')
plt.ylabel('ganancias')

plt.grid()

plt.xlim(0,10)
plt.ylim(0,10)
```



Error Cuadratico Medio (ECM o MSE)

$$ECM = \frac{1}{n} * \sum_{i=1}^{n} (y_i - \hat{y_i}^2)$$

```
sum_y_prima = np.sum(y_prima, axis=0)

def MSE():
```

```
    return (1/n) * np.sum((y - y_prima)**2, axis=0)
mse = MSE()

print(f"""
mse {mse}
""")
```

```
mse 0.12537863415316783
```