

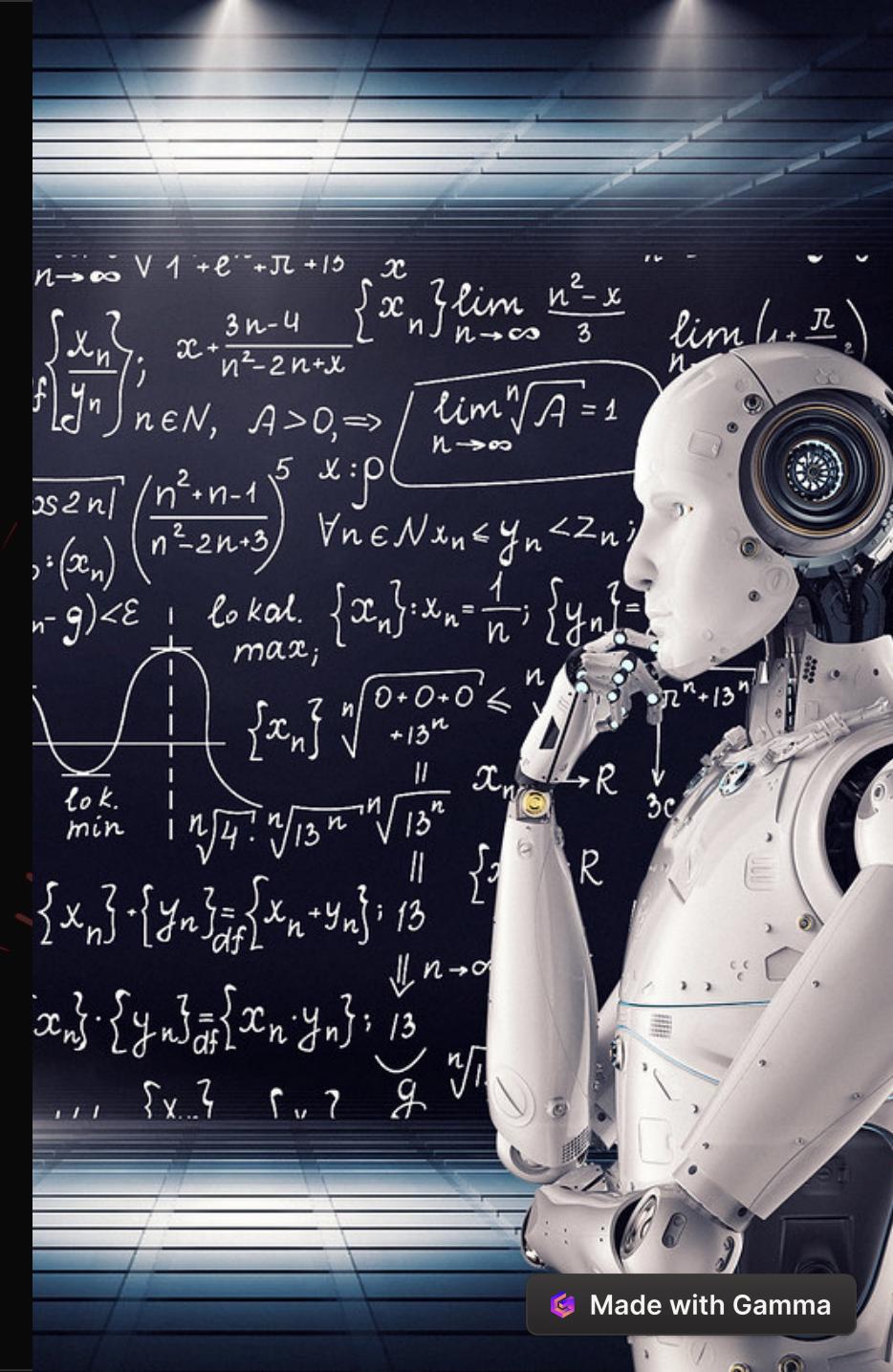
# Regresión Logística Binaria con Numpy y Pandas en Python

Bienvenidos a la **Semana 2** del Curso de Ciencia de Datos. En esta sesión, aprenderemos a aplicar el proceso de código de Regresión Logística Binaria con Numpy y Pandas en Python, a través de un ejemplo práctico usando un dataset de diabetes.

**Cuerpo DOCENTE:** Migue Reyes, Jose Jaime, Vazquez  
Victor

Argentina Programa 4.0 - Tramo 3 -

Universidad Nacional de Salta



# Diferencias entre Regresión Lineal y Regresión Logística

## Propósito distinto

La regresión lineal se usa para predecir valores continuos, mientras que la regresión logística se utiliza para predecir la probabilidad de un resultado binario.

## Función de costo diferente

La regresión lineal utiliza la Suma de Errores Cuadráticos como función de costo, mientras que la regresión logística utiliza la entropía cruzada.

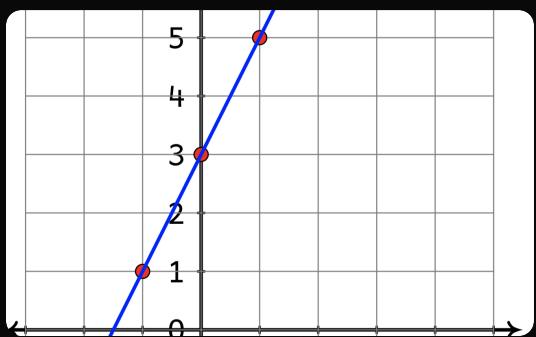
## Curva de ajuste diferente

La regresión lineal da lugar a una línea recta de ajuste, mientras que la regresión logística da lugar a una curva sigmoidea.

## Cálculo de error distinto

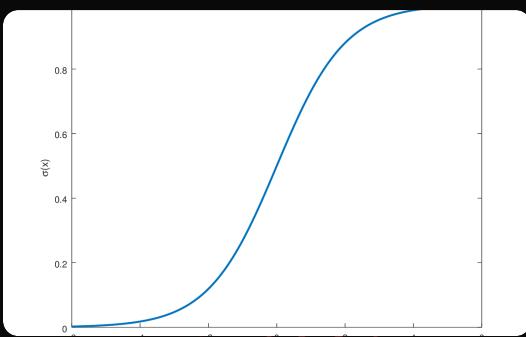
La regresión lineal utiliza el Error Cuadrático Medio para evaluar la calidad de ajuste, mientras que la regresión logística utiliza medidas como la precisión, la sensibilidad y la especificidad.

# La Función Lineal en Regresión Logística



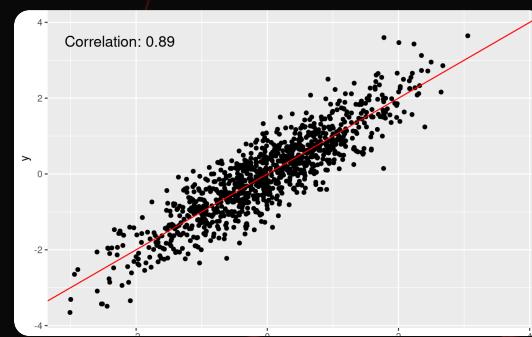
## Función Lineal

La función lineal en la Regresión Logística es la sumatoria ponderada de las variables predictoras. Esta función es transformada luego en una función sigmoidea para obtener la probabilidad de éxito.



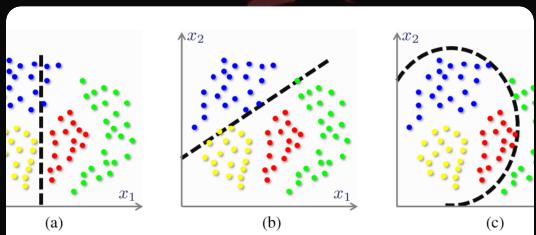
## Función Sigmoidea

La función sigmoidea es una curva en forma de "S" que se acerca al 0 o 1 pero nunca los alcanza. Esta función se utiliza para transformar la función lineal en una función de probabilidad.



## Diagrama de Dispersion

El Diagrama de Dispersion (o Scatter Plot) es una herramienta visual que permite representar las variables en un gráfico. Esencial para entender la relación y correlación entre las variables predictoras.



4 Example weak learners. (a) Axis-aligned hyperplane. (b) General oriented hyperplane. (c) Quadratic (conic in 2D). For ease of visualization here we have  $\mathbf{v} = (x_1 \ x_2)$  and  $\phi(\mathbf{v}) = (\mathbf{v} \cdot \mathbf{v}, 1)$  in homogeneous coordinates. In general data points  $\mathbf{v}$  may live in higher dimensionality and  $\phi$  still a dimensionality of  $\leq 2$ .

## Límite de Decisión

El Límite de Decisión es la línea que separa las regiones donde la probabilidad de éxito es del 50%. Es un resultado importante de la Regresión Logística que ayuda a clasificar los datos.

# Data Set de Diabetes

## Variables Predictoras

Las variables X e Y en el Data Set de Diabetes son respectivamente la glucosa alta y la glucosa baja. Estas dos variables serán utilizadas para predecir la probabilidad de éxito.



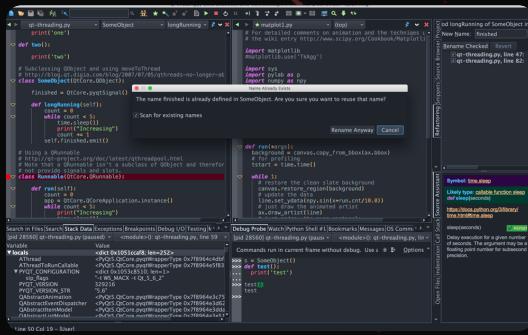
## Descripción del Data Set

El Data Set de Diabetes contiene información médica de 768 pacientes con diabetes, incluyendo variables como la edad, la cantidad de insulina y la presión sanguínea.

## Preprocesamiento del Data Set

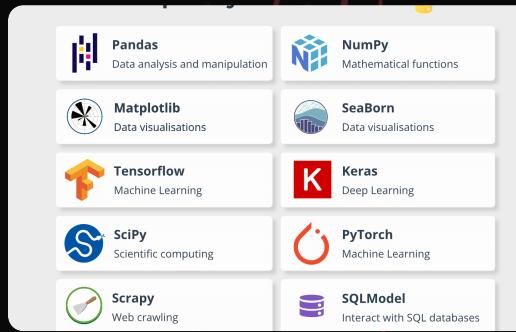
Antes de aplicar la Regresión Logística, es necesario realizar tareas de preprocesamiento de los datos, como eliminar valores nulos o escalar las variables numéricas.

# Ejemplo Práctico: Regresión Logística Binaria con Python



## Interfaz de Desarrollo de Python

Utilizaremos una herramienta de programación como PyCharm o Jupyter Notebook para desarrollar el código necesario para aplicar la Regresión Logística Binaria.



## Librerías de Python

Utilizaremos las librerías de Numpy y Pandas, entre otras, para trabajar con los datos y aplicar la Regresión Logística.

```
first_tiny_script.py
1 # This program says hello and asks for my name
2
3 print('Hello world!')
4 print('What is your name?') # ask for their name
5 myName = input()
6 print('It is good to meet you, ' + myName)
7 print('The length of your name is:')
8 print(len(myName))
9 print('What is your age?') # ask for their age
10 myAge = input()
11 print('You will be ' + str(int(myAge) + 1) + ' in a year.')
```

## Código de Python

El código de Python para aplicar la Regresión Logística Binaria con Numpy y Pandas no es muy complicado, pero requiere de un conocimiento previo de Python.

# ARTIFICIAL INTELLIGENCE

## Conclusión y Puntos Clave

**1** La Regresión Logística es una técnica de aprendizaje supervisado utilizada para predecir resultados binarios.

**2** La Función Lineal es la sumatoria ponderada de las variables predictoras y es transformada en una función sigmoidea para obtener la probabilidad de éxito.

**3** Es importante entender las diferencias entre la Regresión Lineal y la Regresión Logística, y saber cuál utilizar en función del problema que se quiere solucionar.

**4** El Data Set de Diabetes es un buen ejemplo para aplicar la Regresión Logística, pero esta técnica puede ser utilizada en muchos otros problemas.

**5** El conocimiento previo de Python y de las librerías de Numpy y Pandas es fundamental para aplicar la Regresión Logística Binaria con éxito.