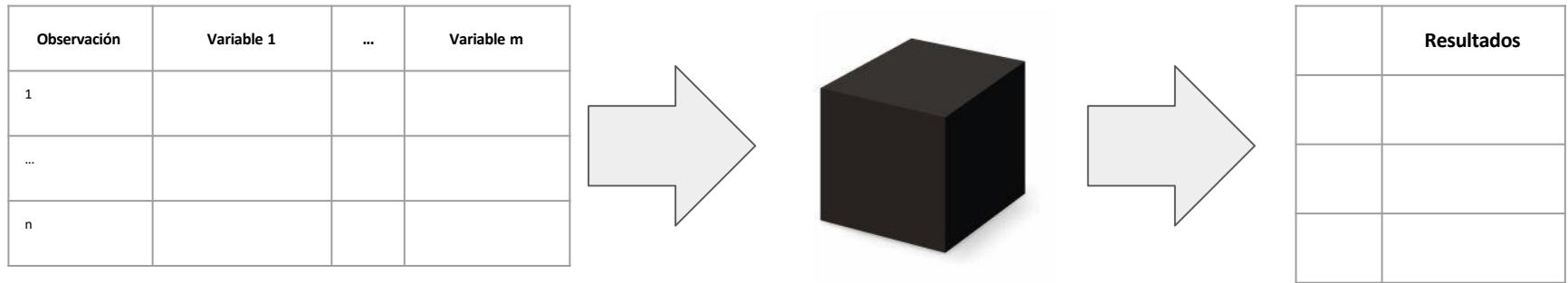


# **MACHINE LEARNING - PYTHON**

# Que és Machine Learning?

## Que es Machine Learning (ML)?

- Es una rama de la **inteligencia artificial** cuyo objetivo es desarrollar técnicas que permitan a las computadoras **aprender**.
- Los algoritmos generalizan comportamientos con base a ejemplos proporcionados.



- Los algoritmos de Machine Learning toman como input diferentes variables de los individuos y arrojan un output.

# Que és Machine Learning?

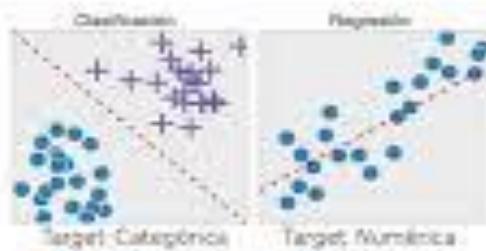
El ML se divide en dos grandes tipos de aprendizaje, supervisado y no supervisado:

- Los de **tipo supervisado** son aquellos que tienen una “clase” que se intenta predecir.
- Los de **tipo no supervisado** son aquellos que no tienen una “clase”, sino que intentan agrupar casos en base a similitud entre ellos.

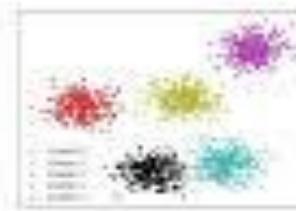
## SUPERVISADO VS NO SUPERVISADO



Se conoce la variable target

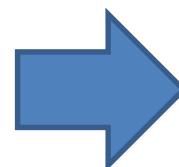


No se conoce la variable target



## Los algoritmos de aprendizaje supervisado pueden ser:

- **Clasificación:** Se predice un evento con una cantidad de valores discretos
  - Binarios (“Acepta” / “No acepta”)
  - Multiclasificación (“Montevideo”, “Rocha”, “La Paloma”)
- **Regresión:** Se predice un valor continuo

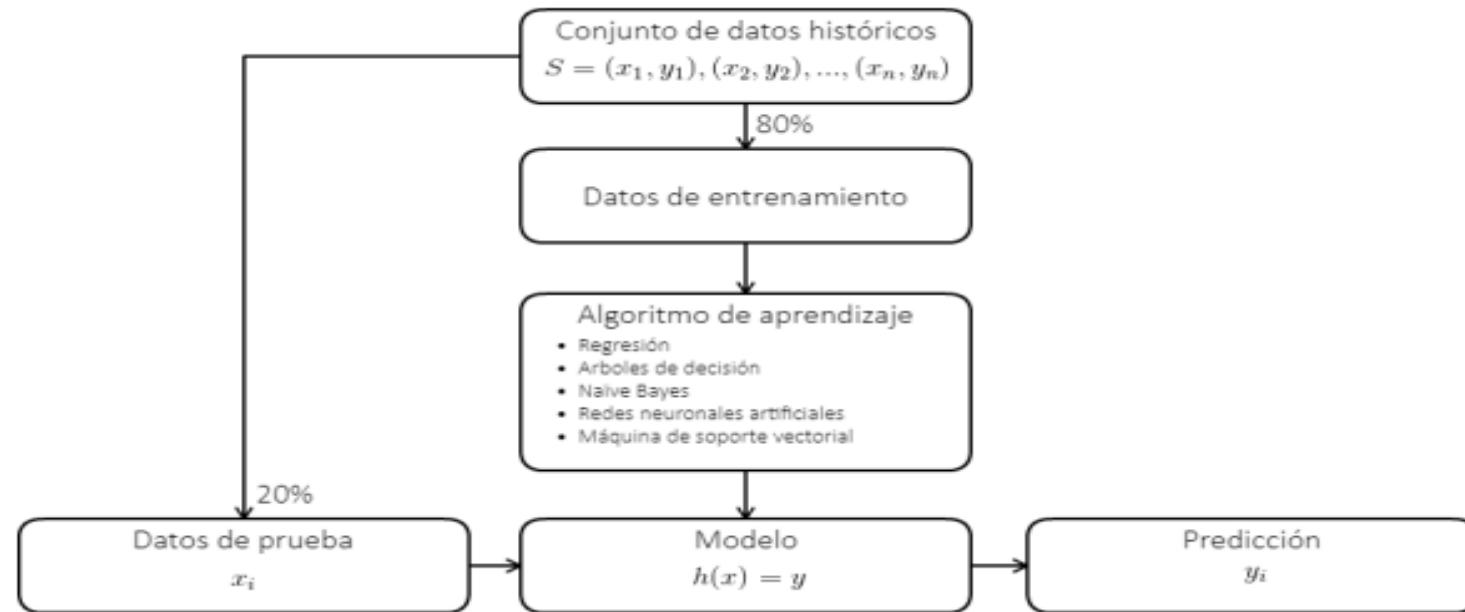


En síntesis, lo que vamos a intentar hacer es predecir:

- Un evento futuro en base a lo que conocemos del pasado.
- Un evento en una población en base a lo que conocemos de otra población

Cuando la variable **output “y”** es **continua** se habla de un problema de regresión, mientras que cuando es **nominal o discreta**, se habla de un problema de clasificación.

# Aprendizaje Supervisado



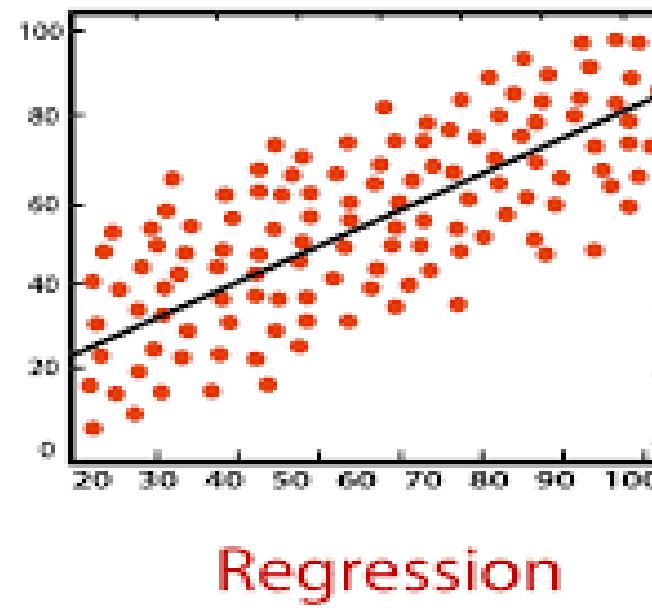
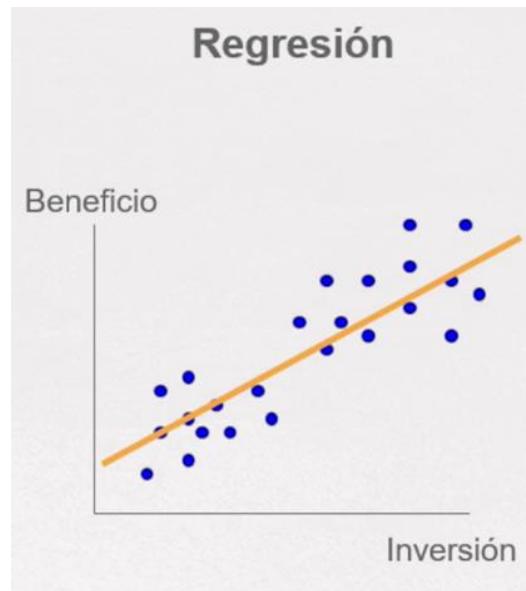
Un ejemplo de este tipo de aprendizaje en la vida cotidiana es un grupo de alumnos en una clase, el profesor imparte un tema con ejemplos, da ejercicios y los alumnos intentan reproducir lo aprendido para llegar al mismo resultado.

**El aprendizaje supervisado realiza tareas predictivas las cuales se dividen en dos tipos: regresión y clasificación.**

# Aprendizaje Supervisado - Regresión

El objetivo del **aprendizaje supervisado de tipo regresión** es predecir valores numéricos. En ese tipo de tarea los atributos son valores continuos.

Por ejemplo, estimar el precio de la gasolina a partir del histórico de precios del petróleo y la demanda del combustible registrado en años anteriores



# Aprendizaje Supervisado - Clasificación

---

El **aprendizaje supervisado de tipo clasificación** consiste en predecir la etiqueta o categoría de un elemento o individuo a partir de los atributos que los distinguen, se trata de eventos discretos. **El problema de clasificación se divide en binaria y multiclas.**

**Binaria:** sólo existen dos categorías.

$$y \in \{0, 1\}$$

Un **ejemplo de este tipo de clasificación** es el correo electrónico que con su filtro etiqueta los correos como “spam” y “no spam”

# Algoritmos de aprendizaje supervisado

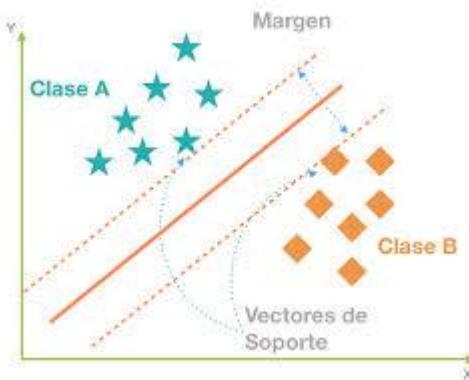
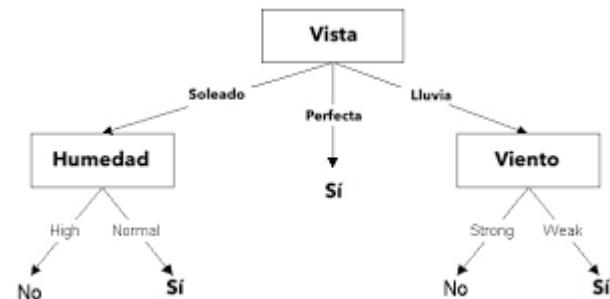
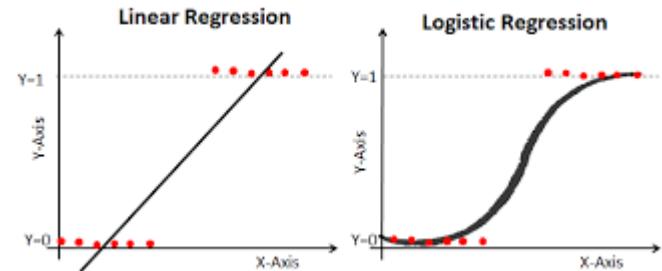
Hay muchos tipos diferentes de algoritmos de aprendizaje supervisado, pero algunos de los más comunes incluyen:

**Regresión lineal:** Se utiliza para predecir un valor continuo, como el precio de una casa o la temperatura.

**Regresión logística:** Se utiliza para predecir un valor binario, como si un cliente va a comprar un producto o no.

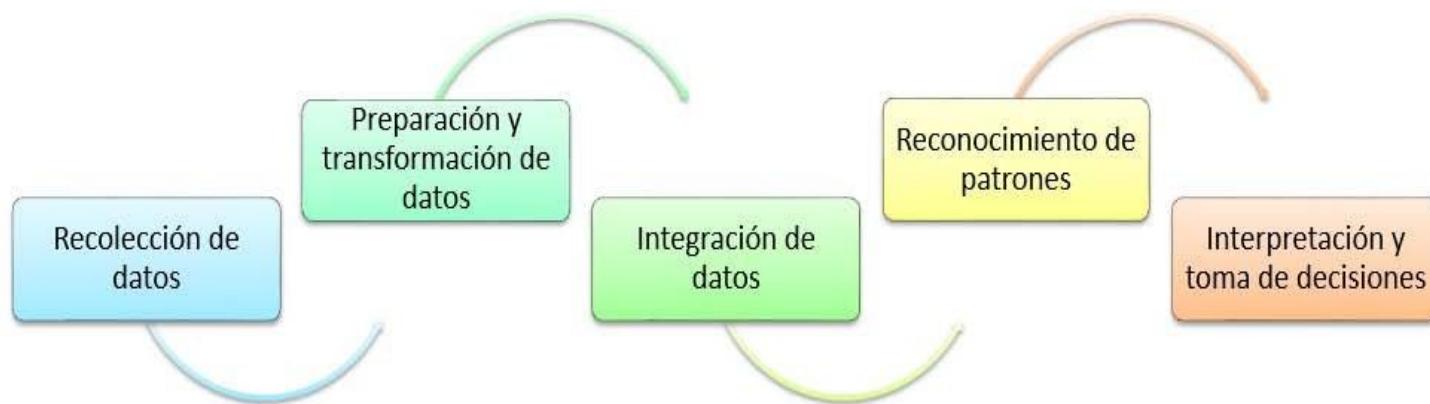
**Árboles de decisión:** Se utilizan para clasificar datos en diferentes categorías.

**Máquinas de vectores de soporte:** Se utilizan para encontrar el mejor límite entre dos categorías de datos.



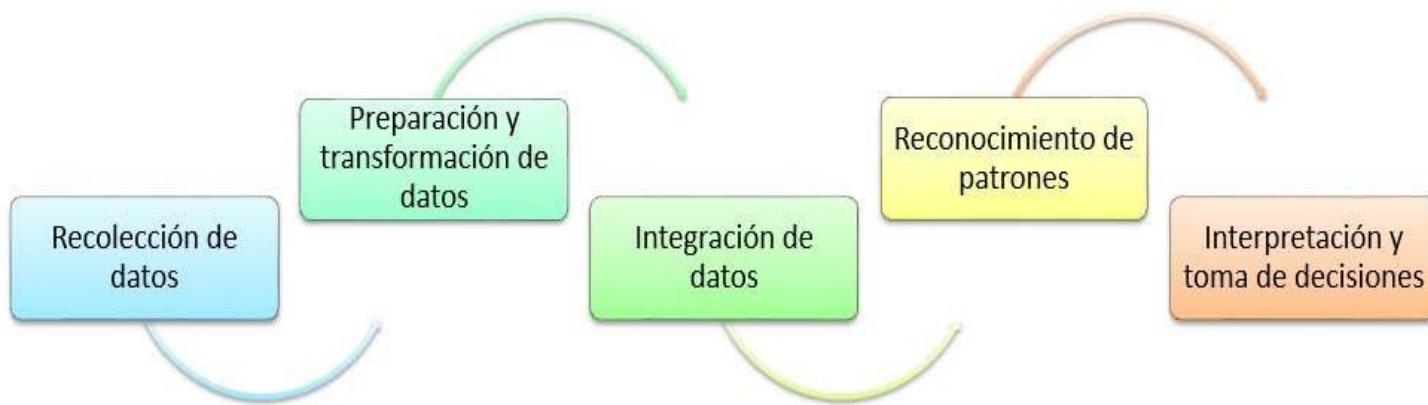
# Metodología para Machine Learning

A continuación, se describe la metodología a implementar en este trabajo de investigación, planteada en una serie de pasos que involucra desde la recolección de datos hasta la generación de conocimiento



**Recolección de datos:** En esta etapa se identifican las fuentes de información que cumplen con los requerimientos del problema. Se extraen los datos relevantes de su medio ambiente origen desde una o varias fuentes como bases de datos públicas o bases de datos privadas.

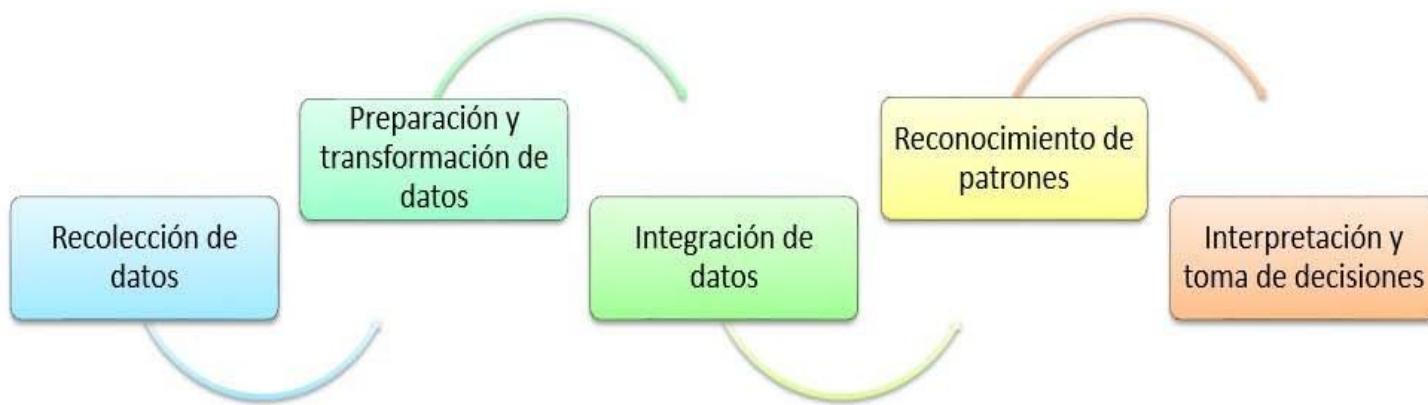
# Metodología para Machine Learning



**Preparación y transformación de datos.** Se le da tratamiento a los datos faltantes, anómalos o inconsistentes al aplicar técnicas de análisis de datos que incluyen medidas estadísticas y gráficas utilizadas para visualizar el comportamiento y mejorar la calidad en los datos.

**Integración de datos.** Los datos que provienen de múltiples fuentes con frecuencia son combinados en un almacén de datos para que todos los datos tengan un formato en común. Aunque no es imprescindible. En algunos casos, se suele trabajar con los datos originales en archivos de texto plano u hojas de cálculo.

# Metodología para Machine Learning



**Reconocimiento de patrones.** Este paso es fundamental es donde se decide el tipo de tarea de aprendizaje a realizar (regresión, clasificación o agrupamiento) así como el algoritmo de Machine Learning más apropiado para la construcción del modelo.

**Interpretación y toma de decisiones.** Al obtener un modelo que sea óptimo para resolver el problema se pasa a la etapa de interpretación. En esta etapa se hace uso de la estadística descriptiva para presentar los datos de forma gráfica, facilitando el análisis del conocimiento que se ha generado y ponerlo a nivel de usuario. Una vez que el modelo está en uso el científico de datos se elige soluciones estratégicas basadas en los resultados obtenidos en la etapa anterior.

**Lenguajes de programación:** Python se ha convertido en el lenguaje de programación líder en el campo del Machine Learning (ML), gracias a su sintaxis sencilla, su gran comunidad y la gran cantidad de bibliotecas y frameworks disponibles.

**Bibliotecas y frameworks:** Python cuenta con una gran cantidad de bibliotecas y frameworks especializados en ML, como Scikit-learn, TensorFlow y PyTorch, que te permiten implementar algoritmos y modelos de manera eficiente..

En Python, el ML se sostiene sobre cuatro pilares :

- **Pandas:** Para manipular datos.
- **NumPy:** Para operaciones matemáticas rápidas.
- **Matplotlib / Seaborn:** Para graficar y entender qué dicen los datos.
- **Scikit-Learn:** La librería principal para el ML tradicional (regresiones, clasificaciones).

## 1. Carga de datos: Leer archivos CSV o bases de datos.

```
import pandas as pd  
# Cargar un archivo CSV  
df = pd.read_csv('datos.csv')
```

## 2. Limpieza (EDA): Manejar valores nulos o datos ruidosos.

```
# Ver si hay valores nulos  
print(df.isnull().sum())  
  
# Borrar filas con datos faltantes  
df = df.dropna()  
  
# Ver estadísticas básicas (promedio, min, max)  
print(df.describe())
```

### 3. División de datos: Train/Test Split (entrenar con una parte, evaluar con otra).

```
from sklearn.model_selection import train_test_split

# Definir variables
(X = características, y = objetivo)
X = df.drop('columna_objetivo', axis=1)
y = df['columna_objetivo']

# Dividir: 80% para entrenar, 20% para evaluar

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=42)
```

## 4. Entrenamiento: (entrenar con una parte, evaluar con otra).

```
from sklearn.linear_model import LinearRegression  
  
modelo = LinearRegression()  
  
modelo.fit(X_train, y_train)
```

## 5. Evaluación:

```
from sklearn.metrics import mean_squared_error, r2_score

# Realizar predicciones con los datos que el modelo NO conoce

predicciones = modelo.predict(X_test)

# Si es Regresión (¿Cuánto?)

print("Error Medio:", mean_squared_error(y_test, predicciones))
r2 = r2_score(y_real, y_pred)
```

# Ejemplo comparativo en Python – algoritmo supervisados

---

Opción A: Regresión Lineal

```
from sklearn.linear_model import LinearRegression  
modelo = LinearRegression()
```

Opción B: Árbol de Decisión (¡Solo cambias estas dos líneas!)

```
from sklearn.tree import DecisionTreeRegressor  
modelo = DecisionTreeClassifier()
```

# EL RESTO DEL CÓDIGO NO CAMBIA:

```
modelo.fit(X_train, y_train)
```

```
predicciones = modelo.predict(X_test)
```

---

Como vimos, scikit-learn nos **permite cambiar de un modelo simple (Regresión Lineal) a uno más complejo (Árboles de Decisión) casi sin tocar el código**. Esta estructura de Instanciar, Entrenar, Predecir es el estándar de la industria.

**El Verdadero Examen:** La verdadera potencia de un modelo no está en qué tan bien recuerda lo que ya vio, sino en qué tan bien predice el futuro con datos que NO conoce ( $X_{\text{test}}$ ).