

## 🌟 Introducción a la Misión 2: Modelos de Machine Learning 🌟

### 🎯 Meta de la Misión 2

Que los estudiantes entiendan qué es un modelo, qué tipos existen y por qué las transformaciones dependen del modelo.

### 🧩 ¿Qué es un modelo?

Un modelo es una herramienta que aprende patrones a partir de los datos para hacer predicciones.

Le damos variables (X), aprende la relación con la variable objetivo (Y), y luego predice valores nuevos.

### 📌 Ejemplos:

- Predecir salario  $\rightarrow$  Y: salary\_in\_usd
- Clasificar correo  $\rightarrow$  Y: spam / no spam
- Predecir ventas  $\rightarrow$  Y: ventas

✨ Frase clave: Cada modelo entiende el mundo de forma diferente; por eso debemos preparar los datos según el modelo.


### 📊 Modelos Lineales

📐 Ven el mundo como líneas rectas

#### ✨ Características:

- Necesitan datos limpios
- Sufren con outliers
- Prefieren variables cercanas a la normalidad
- Requieren encoding para variables categóricas


## Modelos Basados en Árboles

 Funcionan como un árbol de decisiones: 'si salario > 100k...'

✨ Características:

- No requieren normalidad
- Toleran outliers
- Funcionan bien con categorías numéricas
- Encuentran relaciones no lineales

## Modelos Basados en Distancia (KNN, SVM)

 Miden qué tan cerca están los puntos

✨ Características:

- Muy sensibles a outliers
- Necesitan normalización
- No entienden categorías sin encoding

## Redes Neuronales


 Aprenden patrones complejos

✨ Características:

- Requieren escalamiento
- Necesitan encoding
- No exigen normalidad perfecta

## Relación entre Modelos y Transformaciones

Ahora que sabemos cómo piensan los modelos, vamos a preparar los datos según lo que necesitan.

 Tabla general:

- Log-transform → Modelos lineales
- One-Hot Encoding → Todos (menos árboles que toleran Label)
- Normalización → KNN, SVM, Redes Neuronales

- Eliminar outliers → Lineales y KNN
- Sin normalidad → Árboles



### Actividad Inicial

Usando lo que vimos en el EDA de salarios:

- 1 ¿Qué modelo funcionaría mejor?
- 2 ¿Requeriría normalización?
- 3 ¿Necesitaría encoding?
- 4 ¿Sería sensible a outliers?