



# Guía Rápida de Modelos de Machine Learning

Aplicaciones, ventajas, desventajas y cómo elegir el modelo correcto

# ¿Qué aprenderás en esta guía?



## Tipos de problemas

Regresión, clasificación y series temporales



## Selección de modelos

Qué algoritmo usar según tu variable objetivo



## Casos reales

Ejemplos prácticos de cada tipo de modelo



## Pros y contras

Ventajas y limitaciones de cada enfoque

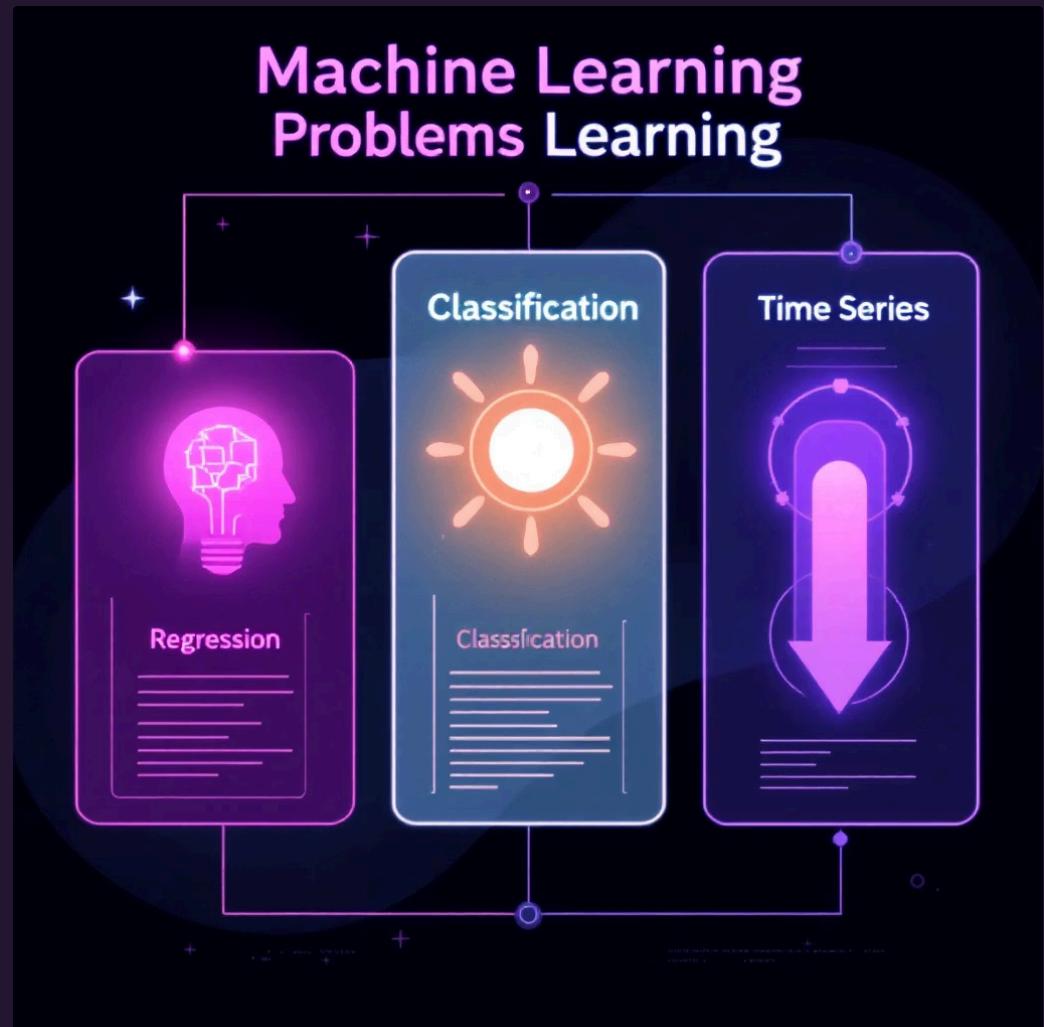


## Requisitos de datos

Qué necesitas preparar antes de modelar

# Tipos de problemas en Machine Learning

Elegir el enfoque correcto comienza por identificar qué tipo de problema estás resolviendo. Cada categoría tiene características únicas que determinarán tu estrategia de modelado.



1

## Regresión

Predecir valores numéricos continuos como precios, temperaturas o ingresos

2

## Clasificación binaria

Distinguir entre dos categorías mutuamente excluyentes (sí/no, fraude/legítimo)

3

## Clasificación multiclas

Asignar observaciones a una de múltiples categorías posibles

4

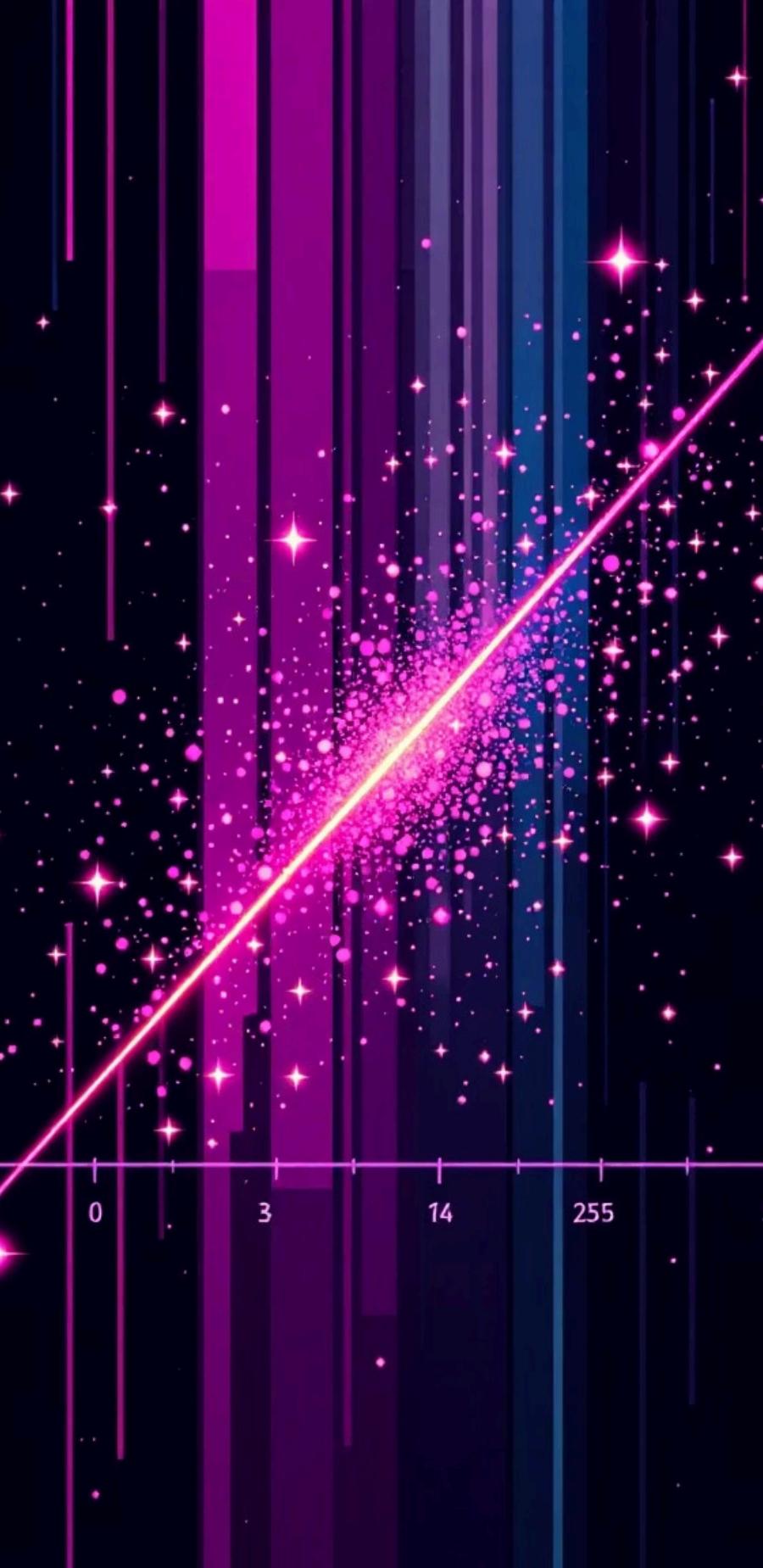
## Clasificación multietiqueta

Asignar múltiples etiquetas simultáneas a cada observación

5

## Series de tiempo

Analizar y predecir patrones temporales con datos secuenciales



# Regresión: predecir valores numéricos

## Ejemplos prácticos

- Estimar salarios según experiencia
- Predecir precios inmobiliarios
- Proyectar temperaturas futuras
- Calcular ventas esperadas

## Modelos recomendados

### Regresión Lineal

Simple y interpretable para relaciones lineales

### Random Forest

Robusto ante outliers, captura no linealidades

### Regresión Polinómica

Para relaciones curvas entre variables

- Requisitos clave:** Variable objetivo numérica continua, datos limpios con pocos valores atípicos extremos, y suficiente variabilidad en el rango de valores.

# Clasificación binaria: decisiones entre dos opciones



## Aprobación de créditos

¿Aprobar o rechazar una solicitud bancaria?



## Detección de fraude

¿Es esta transacción legítima o sospechosa?



## Diagnóstico médico

¿Está presente o ausente la enfermedad?

## Modelos más utilizados

1

### Regresión Logística

Probabilidades interpretables

2

### Árbol de Decisión

Reglas claras y visualizables

3

### Random Forest

Mayor precisión, menos interpretable

4

### SVM

Efectivo con datos de alta dimensión

**Requisitos:** Variable objetivo binaria (0/1), codificación adecuada de variables categóricas, balance razonable entre clases para evitar sesgos.

# Clasificación multiclas: más de dos categorías

## Ejemplo: segmentación por edad

Clasificar personas en grupos etarios permite personalizar estrategias de marketing, servicios de salud o políticas públicas según las necesidades de cada segmento poblacional.

### Jóvenes (15-44 años)

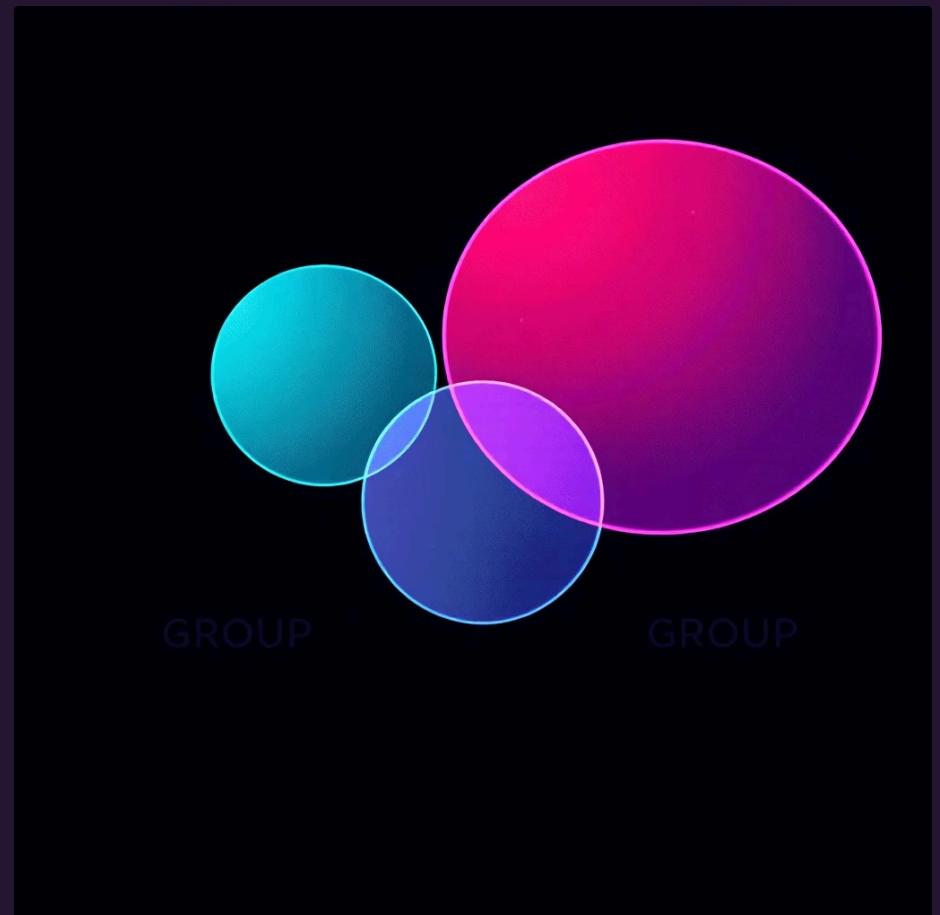
Prioridad en tecnología y desarrollo profesional

### Adultos (45-64 años)

Enfoque en estabilidad y planificación financiera

### Mayores (65+ años)

Servicios de salud y bienestar prioritarios



## Modelos efectivos

- **Random Forest:** Maneja bien múltiples clases sin configuración especial
- **Árbol de Decisión:** Interpretable pero propenso a sobreajuste
- **k-NN:** Simple y efectivo con buenos datos de entrenamiento

- **Requisito crítico:** Dataset balanceado entre clases para evitar que el modelo favorezca las categorías más frecuentes.

# Clasificación multietiqueta: múltiples características simultáneas

A diferencia de la clasificación multiclas donde cada observación pertenece a una sola categoría, en multietiqueta un mismo elemento puede tener varias etiquetas activas simultáneamente.

## Ejemplo: categorización de productos

Un mismo artículo puede ser clasificado como "popular", "caro" e "importado" al mismo tiempo

## Caso: análisis de contenido

Un artículo de noticias puede etiquetarse como "política", "economía" y "internacional" simultáneamente

## Aplicación: diagnóstico médico

Un paciente puede presentar múltiples condiciones o síntomas al mismo tiempo

## Modelos especializados

1

### Random Forest MultiOutput

Extensión que predice múltiples etiquetas binarias

2

### One-vs-Rest

Entrena un clasificador binario por cada etiqueta posible

**Estructura de datos:** Requiere múltiples columnas binarias (0/1) como variables objetivo, una por cada etiqueta posible.

# Series de tiempo: cuando el orden temporal importa

Las series temporales capturan patrones que se repiten o evolucionan a lo largo del tiempo: tendencias, estacionalidad y ciclos que los modelos tradicionales no detectan.



## Ventas mensuales

Proyectar demanda futura considerando estacionalidad y tendencias de crecimiento



## Predicción climática

Estimar temperaturas basándose en patrones históricos y variaciones estacionales



## Violencia intrafamiliar

Identificar patrones temporales para prevención y asignación de recursos

## Modelos recomendados



### Prophet (Meta)

Robusto ante datos faltantes, detecta automáticamente estacionalidad múltiple



### ARIMA

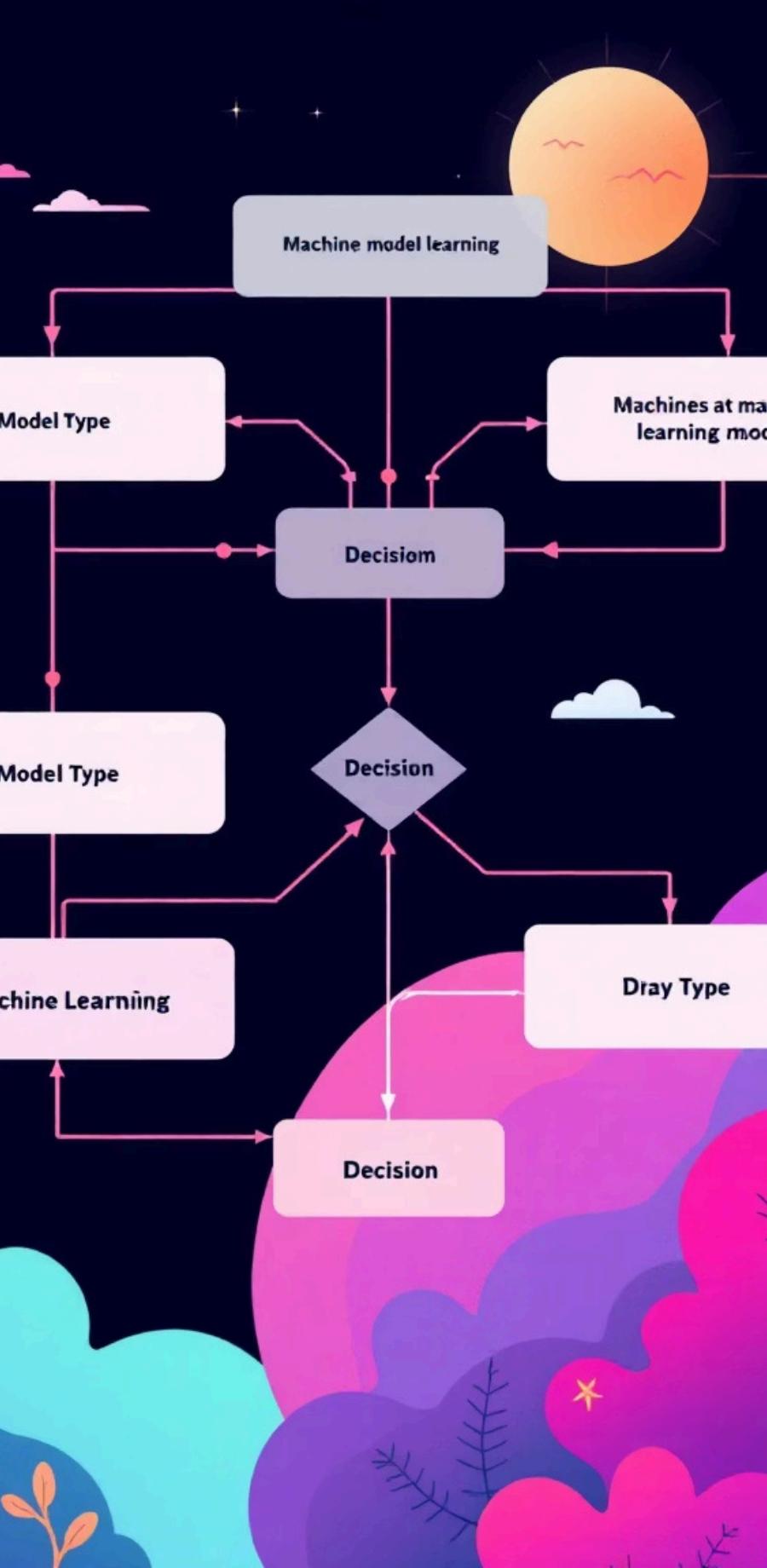
Clásico estadístico, efectivo para series estacionarias sin estacionalidad compleja



### SARIMA

Extensión de ARIMA que incorpora componentes estacionales explícitos

- ❑ **Requisitos esenciales:** Fechas ordenadas cronológicamente, frecuencia consistente (diaria, mensual, etc.), sin huecos temporales significativos en los datos.



# ¿Cómo elegir el modelo correcto?

- 1 ─── ¿Tu objetivo es predecir un número?

→ **Regresión:** Lineal, Random Forest, Polinómica según complejidad
  - 2 ─── ¿Clasificas en categorías mutuamente excluyentes?

→ **Clasificación:** Logística para binaria, Random Forest para multiclase
  - 3 ─── ¿Cada observación puede tener varias etiquetas?

→ **Multietiqueta:** MultiOutput o One-vs-Rest según tu infraestructura
  - 4 ─── ¿El orden temporal es relevante?

→ **Series de tiempo:** Prophet para simplicidad, SARIMA para control fino

Esta decisión inicial determinará el 80% de tu estrategia. El resto es refinamiento: ajuste de hiperparámetros, ingeniería de características y validación rigurosa.

# Principios clave para el éxito en ML



## Simplicidad primero

Comienza con modelos interpretables como regresión lineal o árboles de decisión antes de avanzar a algoritmos complejos



## Preprocesamiento riguroso

La calidad de tus datos determina el éxito: limpia outliers, imputa valores faltantes y normaliza cuando sea necesario



## Variable objetivo clara

Define exactamente qué estás prediciendo y elige el modelo que mejor se adapte a ese tipo de variable



## Evaluación continua

Usa métricas apropiadas (RMSE para regresión, F1-score para clasificación desbalanceada) y valida con datos no vistos

**Recuerda:** No existe un modelo perfecto universal. El mejor modelo es aquel que resuelve tu problema específico con datos disponibles, es interpretable para tus stakeholders y se mantiene en producción de manera confiable.