

# PROYECTO FINAL COMPUTACIÓN CIENTÍFICA I

PREDICCIÓN DE PATRONES CRIMINALES EN LOS ÁNGELES CON  
MODELOS DE ENSAMBLE BAGGING & BOOSTING

OSCAR DANIEL ARCOS REALPE

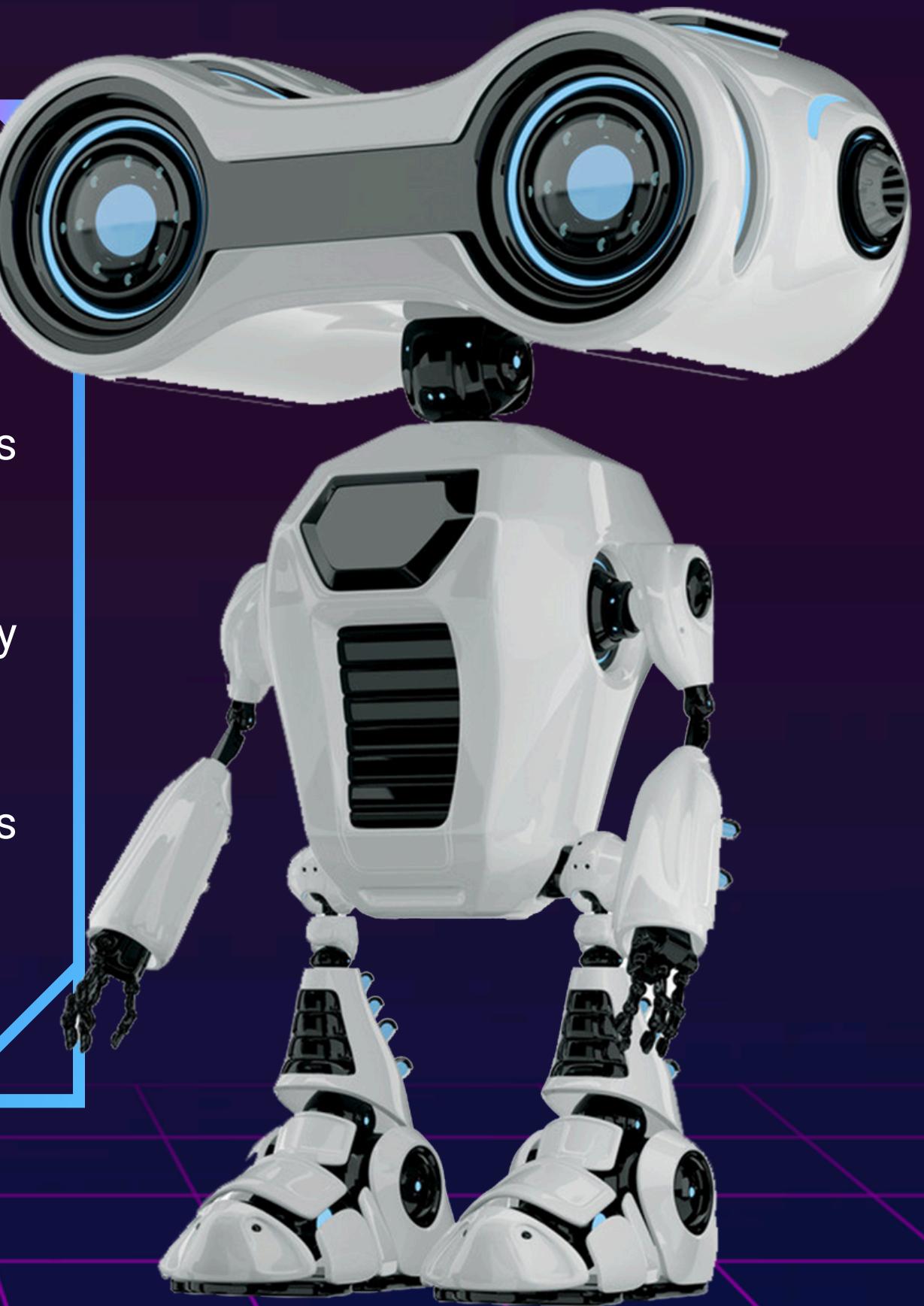
SEMESTRE 2025-I

# CONTEXTO DEL PROYECTO

Se utilizó un conjunto de datos reales del Departamento de Policía de Los Ángeles con información de crímenes ocurridos entre 2020 y 2024.

El **objetivo** fue aplicar técnicas de ensamble como Bagging, Random Forest y Boosting para resolver tareas de clasificación y regresión.

Todo el desarrollo se realizó en Google Colab utilizando Python y bibliotecas como *scikit-learn*, *pandas*, *matplotlib* y demás vistas en clase.



# BAGGING & BOOSTING

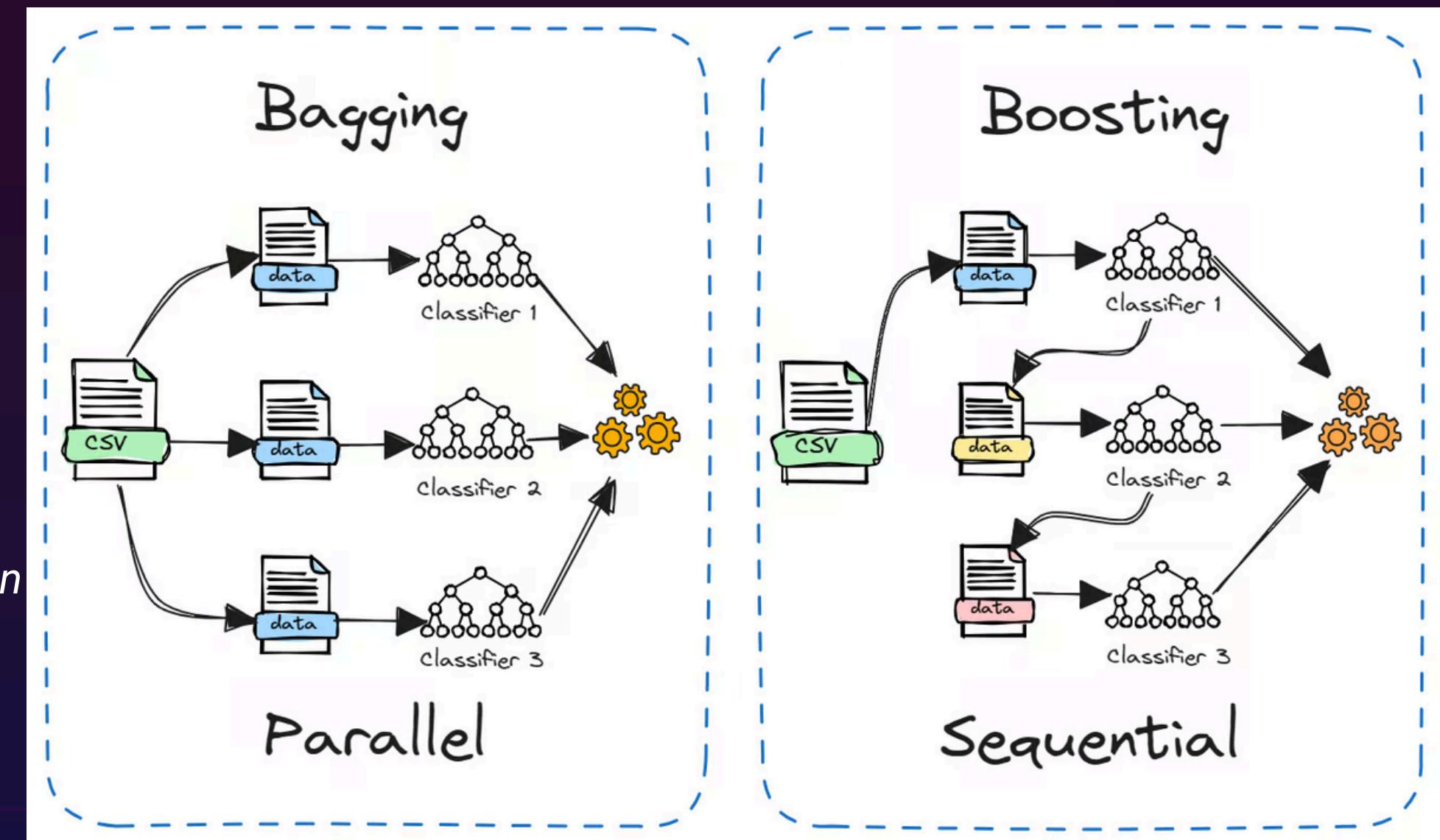
El **Bagging** es una técnica de ensamble que entrena múltiples modelos sobre subconjuntos aleatorios (con reemplazo) del conjunto de entrenamiento.

## Ventajas:

- Reduce la varianza:
- Paralelizable:
- Robusto a outliers:
- Simple de implementar:

## Desventajas:

- No corrige sesgo:
- Menos interpretable:
- Pérdida de información en datos pequeños:



El **Boosting** entrena modelos de forma secuencial. Cada nuevo modelo intenta corregir los errores del anterior, enfocándose en los datos mal clasificados. Un ejemplo clásico es el Gradient Boosting y AdaBoost.

## Ventajas:

- Reduce sesgo y varianza:
- Alta precisión:
- Enfocado en errores:
- Flexibilidad:

## Desventajas:

- Propenso a overfitting:
- Secuencial (lento):
- Sensible a outliers:
- Requiere ajuste fino:

# BAGGING

1. **Muestreo Bootstrap:** Bagging usa muestreo aleatorio con reemplazo. Suponga que se tiene un conjunto de datos  $D$  con  $n$  muestras. Generamos  $B$  subconjuntos de datos  $D_1, D_2, \dots, D_B$ , donde cada  $D_i$  con tiene  $n$  muestras seleccionadas con reemplazo de  $D$ . Así, algunos datos se repiten y otros no aparecen en cada subconjunto.
2. **Entrenamiento Independiente:** Cada subconjunto  $D_i$  entrena un modelo independiente  $M_i$ . Si  $B$  es suficientemente grande, el error de los modelos individuales se promedia, reduciendo la varianza.
  - Fórmula general del Bagging (clasificación):

$$\bar{y} = \text{modo}(\{M_1(x), M_2(x), \dots, M_B(x)\})$$

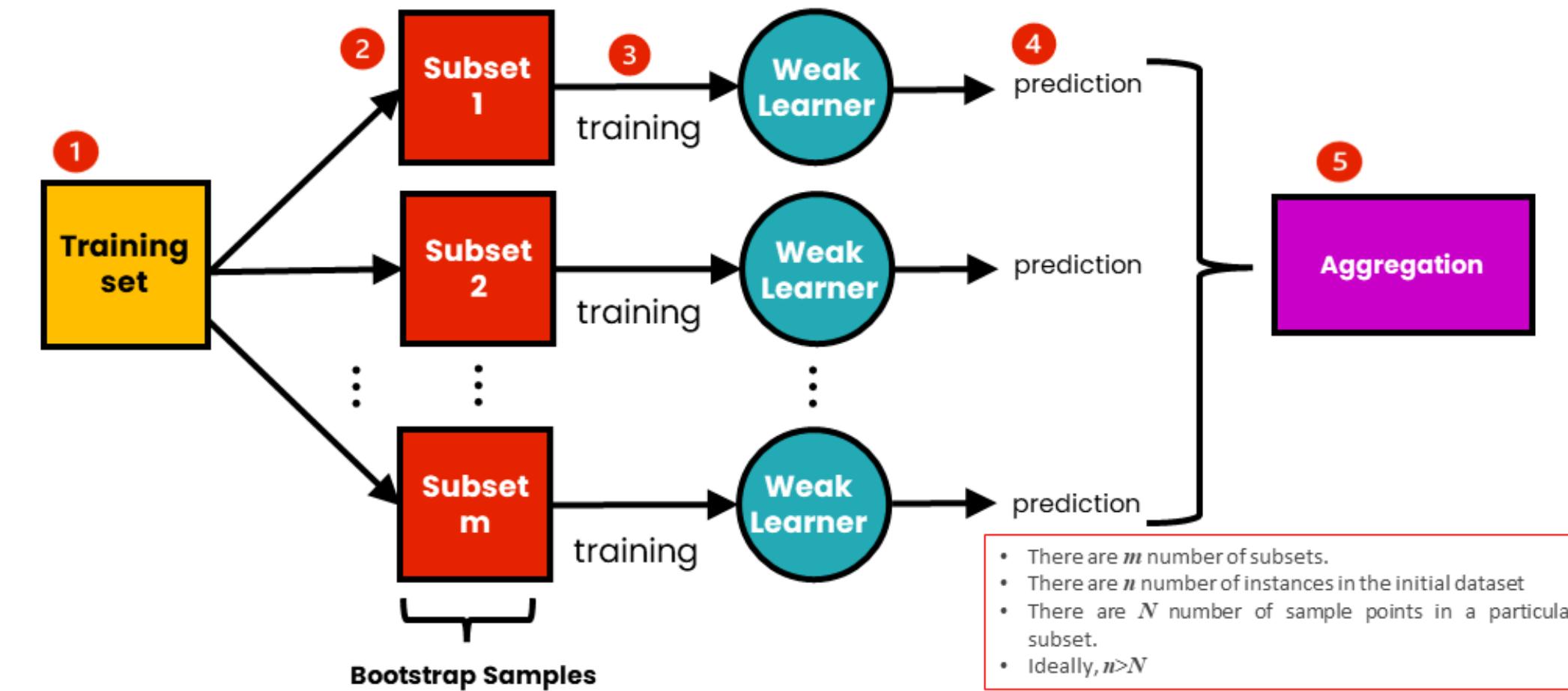
donde  $M_i(x)$  es la predicción de cada modelo sobre el dato  $x$ .

- Fórmula general del Bagging (regresión):

$$\text{Var}\left(\frac{1}{B} \sum_{i=1}^B M_i(x)\right) = \frac{\sigma^2}{B}$$

Aquí,  $B$  es el número de modelos, y  $\sigma^2$  es la varianza de cada modelo base.

## The Process of Bagging (Bootstrap Aggregation)



# BOOSTING

El objetivo principal de Boosting es mejorar el sesgo (bias) y la precisión combinando modelos débiles (ej. árboles poco profundos)

Fórmula del modelo de Boosting (idea general):

Predictión Final (Clasificación/Regresión) en Boosting:

$$y = \sum_{i=1}^T \alpha_t \cdot M_t(x)$$

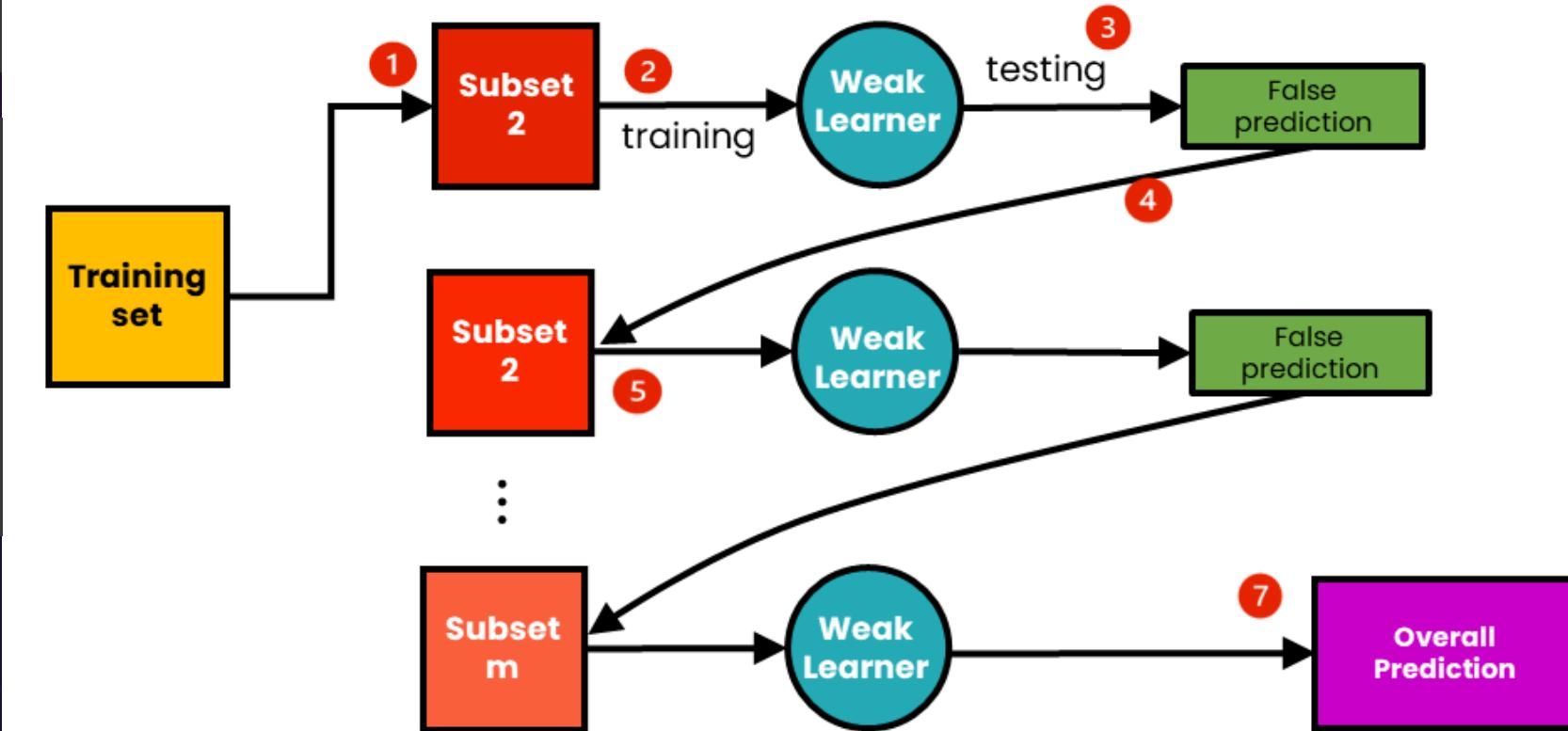
- $T$ : Número total de modelos (iteraciones).
- $\alpha_t$ : Peso del modelo  $M_t$  (calculado con  $\alpha_t = \frac{1}{2} \ln \left( \frac{1-\text{err}_t}{\text{err}_t} \right)$ ).
- $M_t(x)$ : Predicción del modelo  $t$ -ésimo para el dato  $x$ .
- Clasificación: Usar  $\text{sign}(y)$  para decidir la clase.
- Regresión: Valor directo de  $y$ .

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x)$$

Donde:

- $F_m(x)$ : es el modelo acumulado en la iteración  $m$
- $h_m(x)$ : es el nuevo modelo entrenado sobre los errores de  $F_{m-1}(x)$
- $\gamma_m$ : es la tasa de aprendizaje (learning rate)

## The Process of Boosting



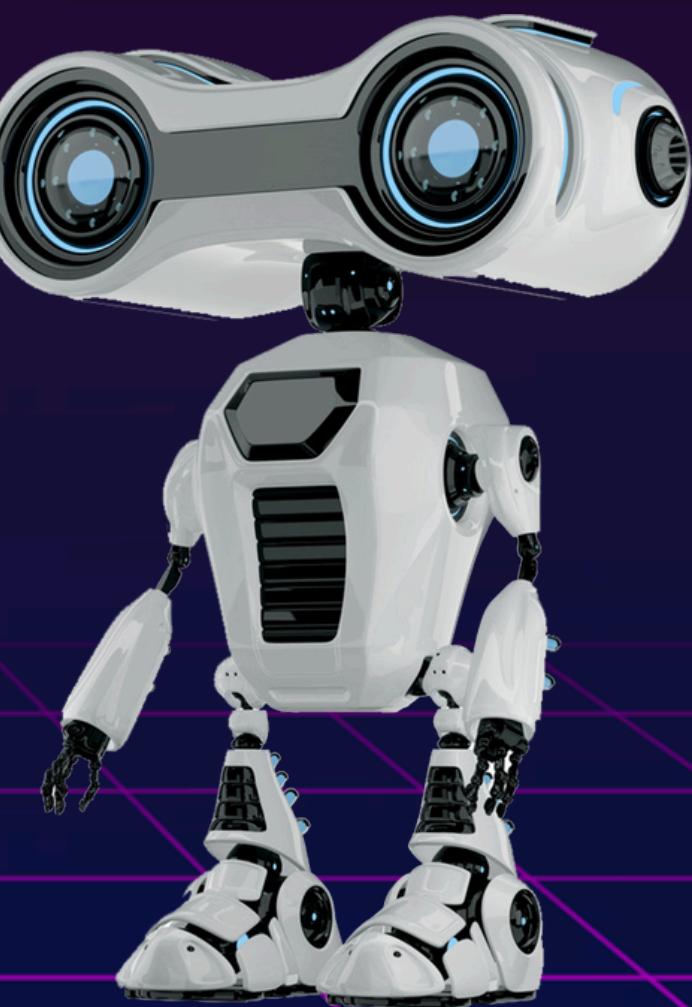
# AREAS DE APLICACIÓN

## Áreas comunes de aplicación Bagging:

- **Predicción de fraudes bancarios:** Donde los datos son muy variables, y se necesita estabilidad.
- **Diagnóstico médico:** Para evitar que modelos se sobreajusten a ruido en datos clínicos.
- **Reconocimiento de voz o imagen:** Donde se trabaja con datos ruidosos y complejos.
- **Evaluaciones educativas:** Clasificación de estudiantes en niveles usando muchos datos inconsistentes.

## Áreas comunes de aplicación Boosting:

- **Finanzas:** Predicción de riesgo crediticio, scoring de clientes
- **Marketing:** Segmentación de clientes y predicción de abandono
- **Detección de spam o malware:** Donde hay que captar patrones sutiles y mejorar cada vez más.
- **Competencias de Machine Learning (como Kaggle):** Boosting (como XGBoost, LightGBM) domina los rankings.



# OBJETIVO GENERAL

Aplicar y comparar técnicas de ensamble para predecir variables relacionadas con crímenes en Los Ángeles:

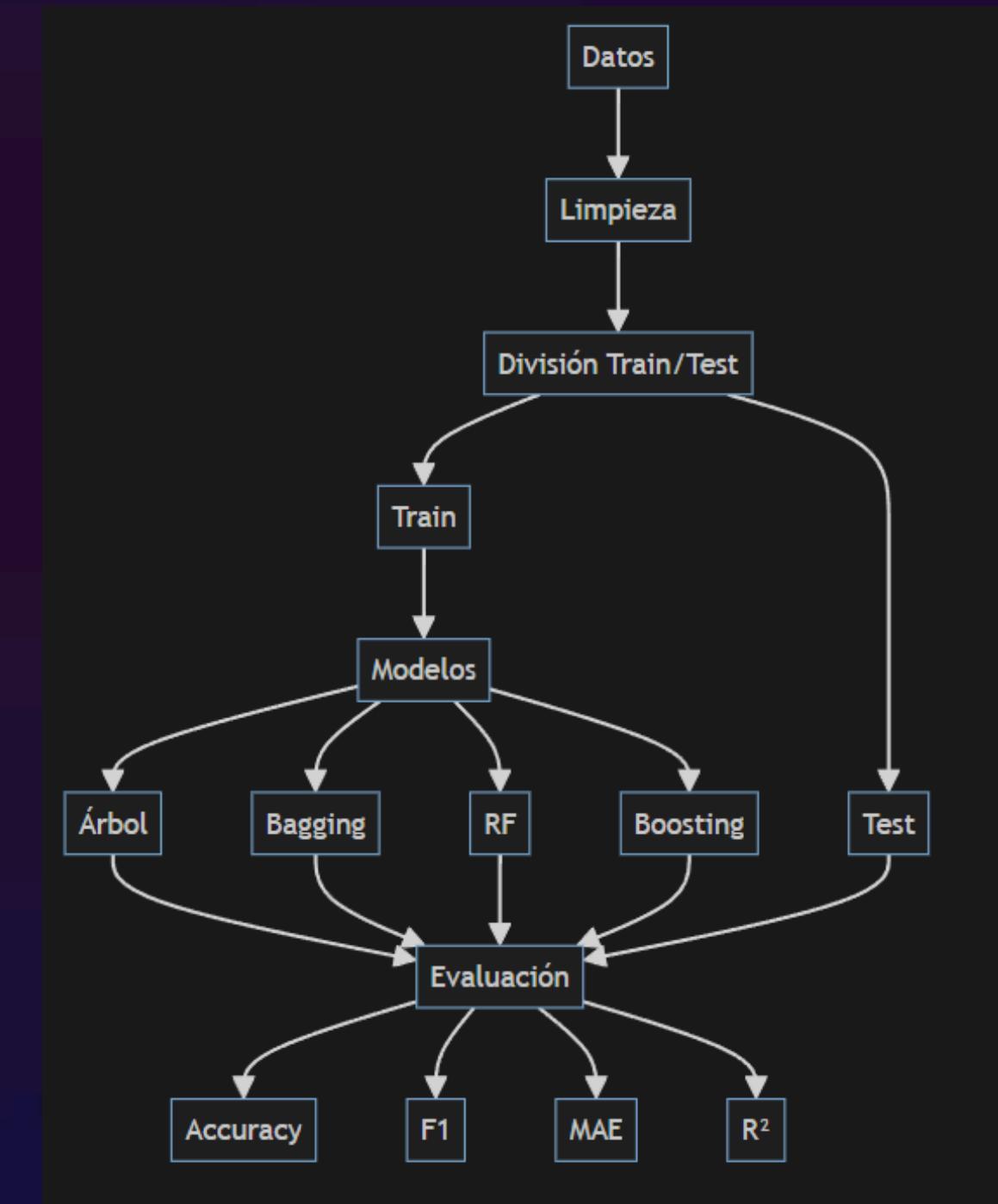
- Estado del caso (Status)
- Gravedad del crimen (Part 1-2)
- Edad de la víctima (Vict Age)

DATE OCC	Día del mes en que ocurrió el crimen
TIME OCC	Hora del día en que ocurrió el crimen (formato 24h)
AREA	Código del área o división policial
Rpt Dist No	Número de distrito del reporte
Part 1-2	Clasificación del crimen: Parte 1 = grave, Parte 2 = menor
Crm Cd	Código específico del tipo de crimen
Vict Age	Edad de la víctima
Premis Cd	Código del lugar donde ocurrió el crimen (ej. calle, casa)
Status	Estado del caso (ej. arresto realizado, pendiente, cerrado)



# METODOLOGÍA

1. Carga y limpieza de datos
2. Codificación de variables
3. División en entrenamiento/test (80/20)
4. Entrenamiento con 4 modelos:
  - Árbol base
  - Bagging
  - Random Forest
  - Boosting
5. Evaluación con métricas:
  - Clasificación: Accuracy, F1
  - Regresión: MAE, MSE, R<sup>2</sup>

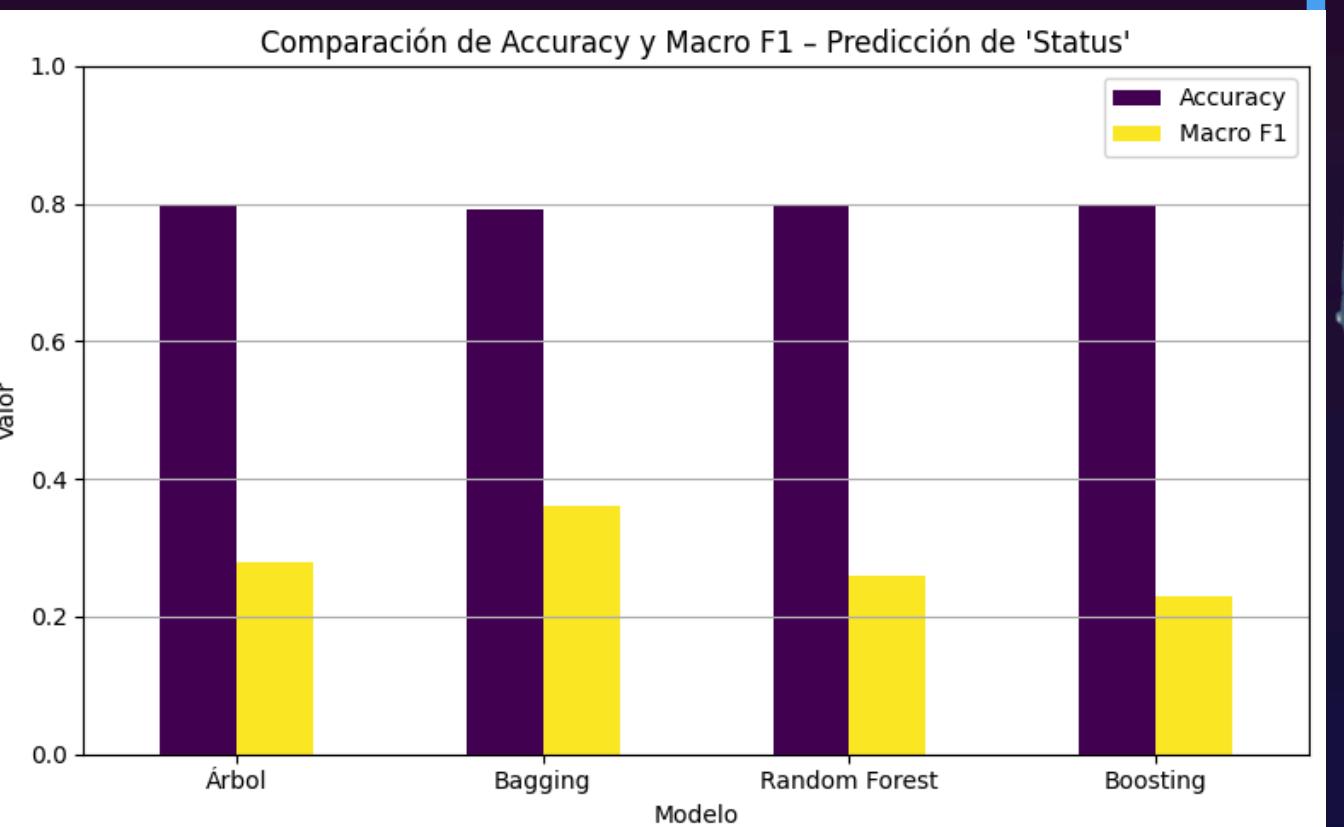


# RESULTADOS - SECCIÓN 1

Predicción del estado del caso (Status)

- Precisión global  $\approx 80\%$
- Alto desbalance en clases
- Bagging logró mejor equilibrio entre clases
- Boosting no superó modelos previos

Bagging fue el más robusto en este escenario



# RESULTADOS - SECCIÓN 2

Predicción del tipo de crimen (Part 1-2)

- Variable binaria y balanceada
- Todos los modelos alcanzaron precisión  $\approx 1.0$
- Modelos aprendieron con facilidad

	Accuracy	F1 Clase 1 (grave)	F1 Clase 2 (menor)	Macro F1
Árbol de Decisión	1.0000	1.0000	1.000	1.0000
Bagging	1.0000	1.0000	1.000	1.0000
Random Forest	0.9981	0.9982	0.998	0.9981
Boosting	1.0000	1.0000	1.000	1.0000



# RESULTADOS - SECCIÓN 3

Predicción de la edad de la víctima (Regresión)

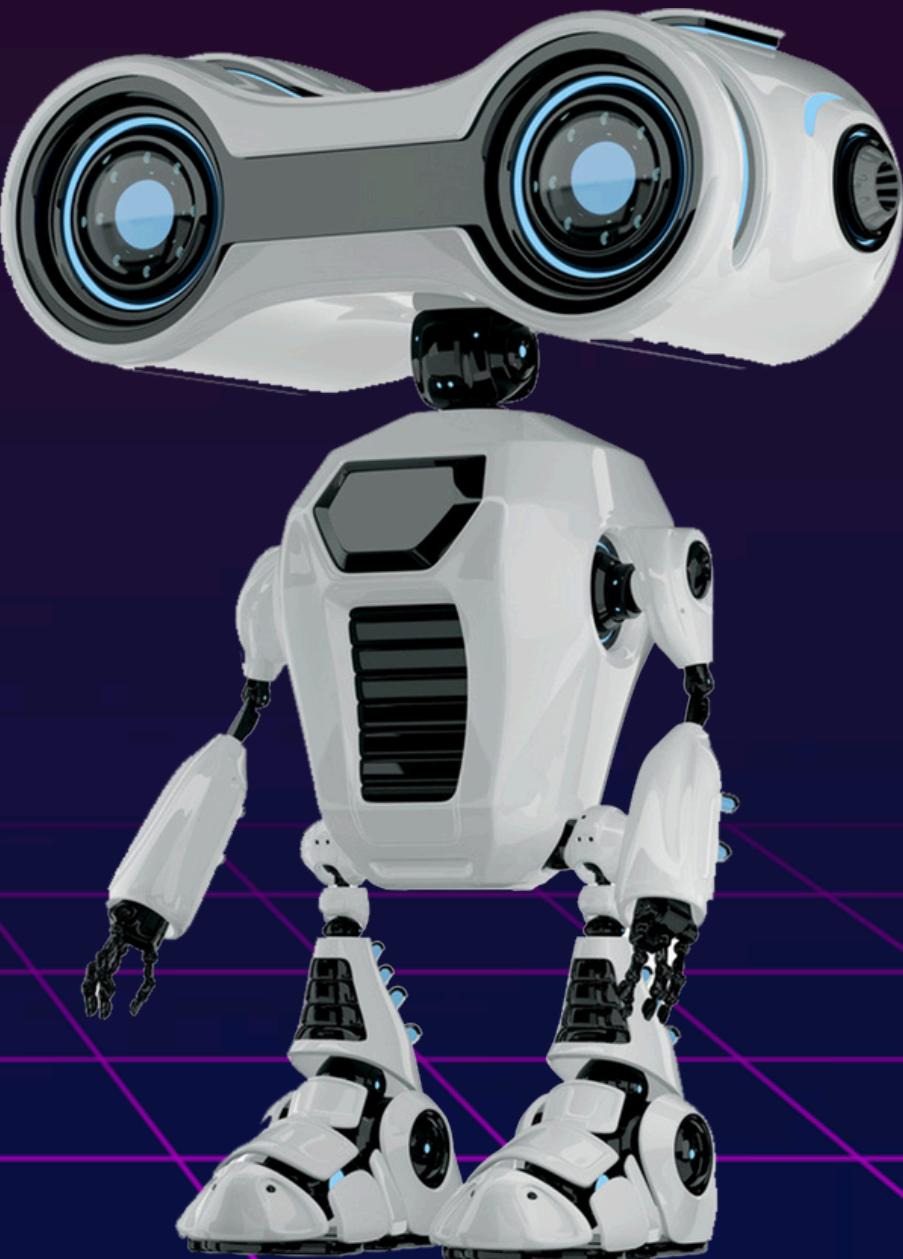
- $R^2 \approx 0.37$
- Mejor modelo: Random Forest

	MAE	MSE	R <sup>2</sup> Score
Árbol de Regresión	0.06	0.03	0.3522
Bagging	0.06	0.03	0.3140
Random Forest	0.06	0.03	0.3680
Boosting	0.06	0.03	0.3407

Predicción de la edad de la víctima  
clasificación por rangos

- 4 rangos definidos
- Modelos obtuvieron precisión y F1 = 1.0
- Boosting excluido por baja representación en una clase

	Accuracy	Macro	F1
Árbol de Decisión	1.0	1.0	1.0
Bagging	1.0	1.0	1.0
Random Forest	1.0	1.0	1.0

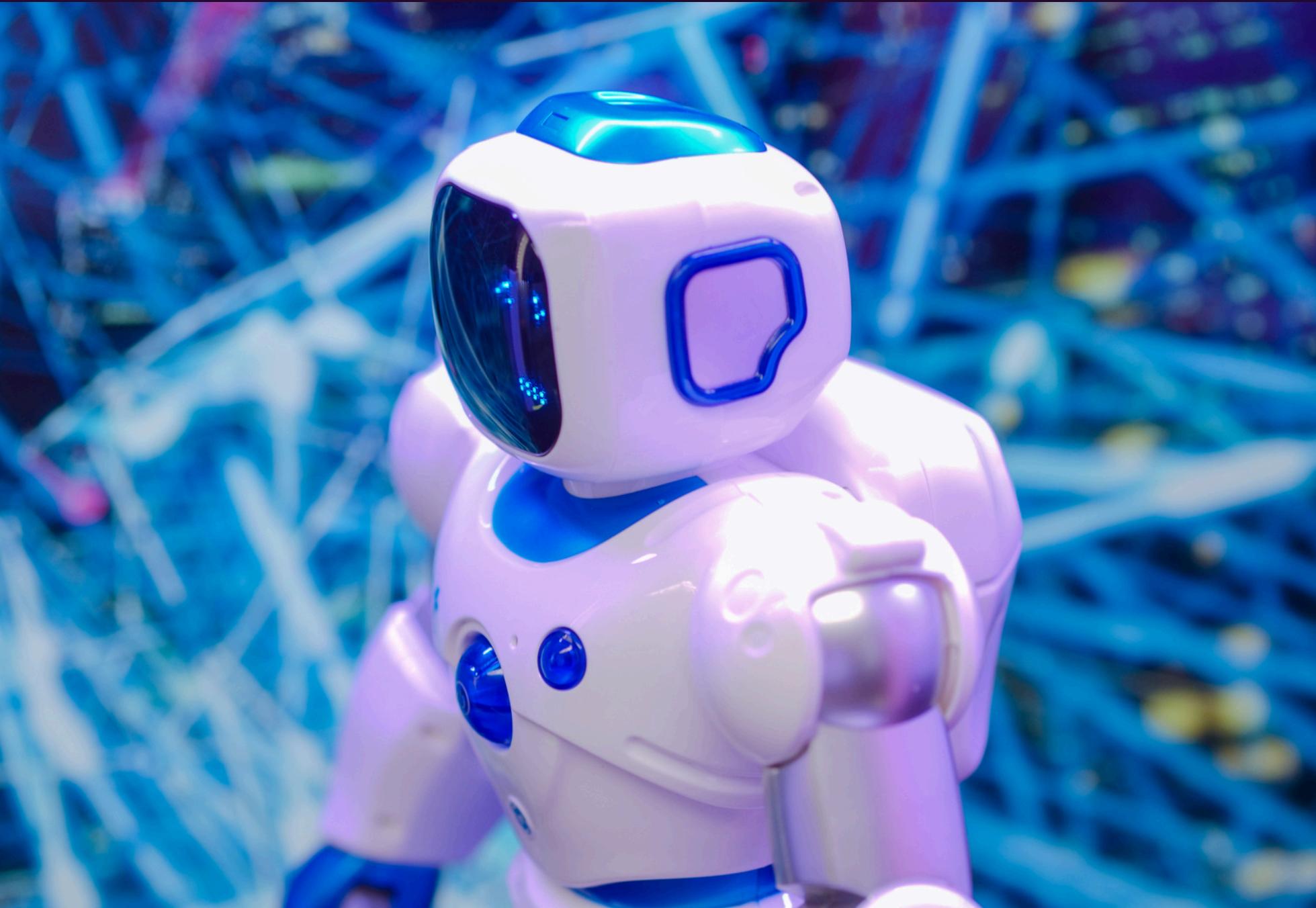


# CONCLUSIONES

- Bagging fue el más confiable en casos desbalanceados
- Boosting mostró potencia, pero no ventajas claras
- La elección de la variable objetivo influye mucho en los resultados
- Clasificación por rangos mejora interpretabilidad

## REFLEXIÓN

- Los modelos de ensamble no solo funcionan en teoría, también se pueden aplicar a contextos del mundo real.
- Este proyecto mostró cómo los datos pueden ayudar a entender patrones complejos como el crimen.
- Las herramientas del curso de computación científica permitieron conectar matemática, programación y problemas sociales reales.



**THANK YOU**