

# Analítica de datos y Machine Learning para las ciencias políticas

Definiciones y ruta de trabajo

Sebastián Emilio Henao Bedoya

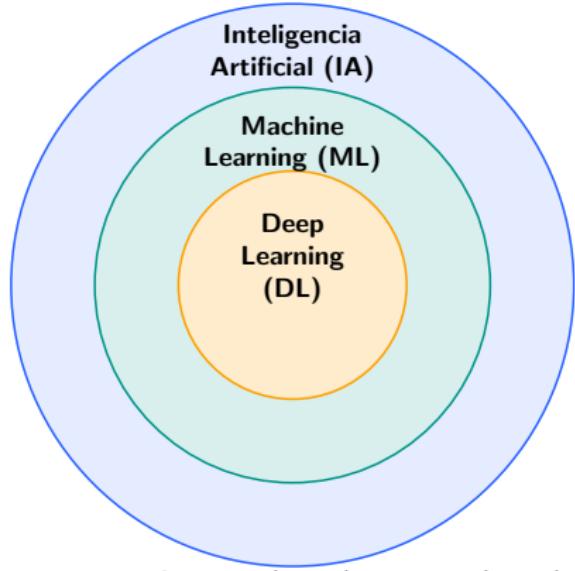
Universidad de Antioquia

1 de octubre de 2025

# Definiciones clásicas

- **Arthur Samuel (1959)**: *dar a los computadores la capacidad de aprender sin ser programados.*
- **Tom Mitchell (1997)**: un programa aprende de la **experiencia (E)** respecto a una **tarea (T)** y una **medida de desempeño (P)** si mejora en T con E.
- **Luis Serrano (2020)**: *ML es sentido común hecho por una computadora.*

# Relación entre IA, ML y DL

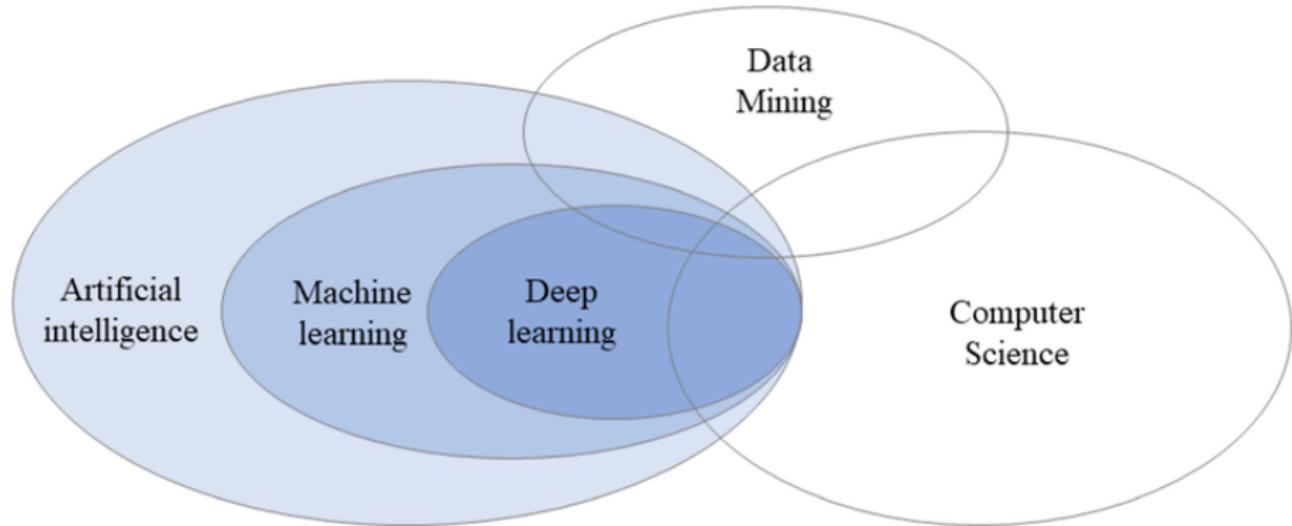


- **IA:** Toma de decisiones por cualquier método (reglas, búsqueda, lógica, datos).
- **ML:** Subcampo de la IA que aprende patrones a partir de **datos**.
- **DL:** Subcampo de ML basado en **redes neuronales profundas** (visión, texto, audio).

Basado en Serrano (2020) y Géron (2021).

- **Inteligencia Artificial (IA)**: campo general que busca que las computadoras realicen tareas cognitivas humanas (razonamiento, lenguaje natural, visión).
- **Machine Learning (ML)**: subcampo de la IA que permite a los sistemas **aprender de datos** en lugar de ser programados con reglas explícitas.
- **Ciencia de Datos**: integra estadística, computación y conocimiento del dominio para **extraer conocimiento útil de los datos**. Incluye: limpieza, visualización, modelado, comunicación de resultados.
- **Minería de Datos**: técnicas para identificar patrones en grandes volúmenes de información (clustering, asociaciones, anomalías). Fue el antecedente de muchos algoritmos de ML modernos.

# Relación IA, ML, DL, DS y DM



Fuente: Zhang et al. (2021).

# ML vs Programación tradicional

- **Programación tradicional:** el programador escribe reglas explícitas (if...then).
  - Ejemplo: para detectar spam, escribir todas las condiciones: “si el correo contiene ganaste, márcalo como spam”.
  - Problema: hay demasiadas variaciones, difícil de mantener.
- **Machine Learning:** el sistema *aprende* las reglas a partir de datos.
  - Ejemplo: entrenar con miles de correos etiquetados como spam/ham.
  - El modelo descubre automáticamente patrones (palabras, frecuencia, remitente).
- **Ventaja de ML:** más flexible y adaptable a cambios.
  - Si aparecen nuevas formas de spam, el modelo se actualiza con datos recientes.
  - En política: predecir encuestas o analizar tweets sin escribir reglas para cada caso.

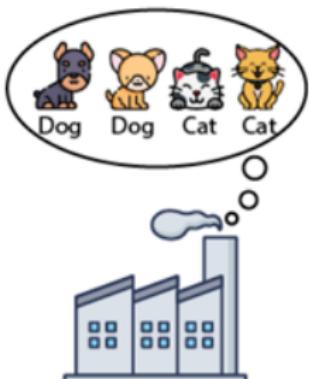
# Ejemplos cotidianos de ML

- **Recomendaciones personalizadas (Netflix, Spotify):** sistemas de filtrado colaborativo aprenden de tus elecciones y las de otros usuarios para sugerir series, películas o música.
- **Diagnóstico médico asistido:** algoritmos de clasificación analizan imágenes (radiografías, resonancias) o datos clínicos para detectar enfermedades como cáncer o COVID-19.
- **Autos autónomos:** redes neuronales profundas procesan en tiempo real cámaras y sensores, distinguiendo peatones, semáforos y otros vehículos.
- **Aplicaciones políticas:**
  - Encuestas y modelos de predicción electoral.
  - Análisis de discursos y narrativas en el Congreso.
  - Detección de desinformación en redes sociales (fake news, bots).

# ¿Cuándo usar ML?

- **Muchas reglas o ajustes manuales:** cuando la solución requiere mantener docenas de reglas operativas, *ML* simplifica el código y se adapta mejor.
- **Problemas complejos sin óptimo claro:** cuando el enfoque tradicional no ofrece una solución cerrada, *ML* encuentra aproximaciones efectivas.
- **Ambientes cambiantes:** con datos que evolucionan, *ML* puede reentrenarse y adaptarse rápidamente.
- **Grandes volúmenes de información:** para extraer patrones y perspectivas en datasets extensos y de alta dimensionalidad.

# Aprendizaje supervisado



Remember



Formulate



Predict

Fuente: Serrano (2020).

# Aprendizaje supervisado: Modelos de regresión

- Contiene datos etiquetados y su objetivo es **predecir valores numéricos**.
- El resultado de la regresión es un número basado en los **features** o características.
- **Ejemplo:** predecir el *precio de una casa*.
- **Modelos de regresión más usados:**
  - ① Regresión lineal.
  - ② Árboles de decisión.
  - ③ Random Forests.
  - ④ AdaBoost.
  - ⑤ Gradient Boosted Trees.
  - ⑥ XGBoost.
- **Flujo:**
  - Datos de entrenamiento → el modelo aprende patrones.
  - Datos de prueba → se valida la exactitud del modelo.

# Modelos de regresión

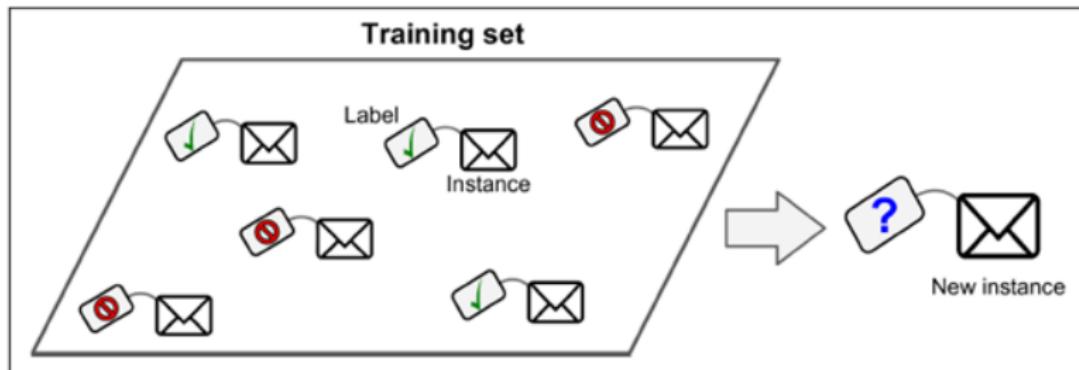


Fuente: Geron (2019).

# Aprendizaje supervisado: Modelos de clasificación

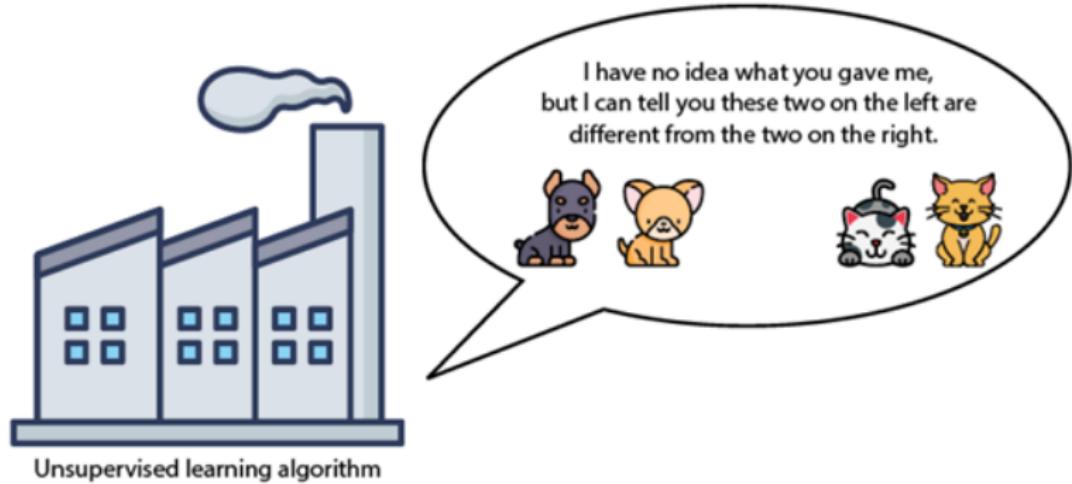
- Contiene datos etiquetados y su objetivo es **predecir categorías**.
- El resultado del modelo es una **clase o estado discreto**.
- **Ejemplo:** detección de spam en correos electrónicos.
- **Aplicaciones frecuentes:**
  - Análisis de sentimientos.
  - Tráfico web.
  - Redes sociales.
  - Recomendación de videos.
- **Modelos más usados:**
  - 1 Clasificación logística.
  - 2 Näive Bayes.
  - 3 Árboles de decisión.
  - 4 Redes neuronales.
  - 5 Support Vector Machines (SVM).
  - 6 Métodos de ensamble.

# Modelos de clasificación



Fuente: Geron (2019).

# Aprendizaje no supervisado

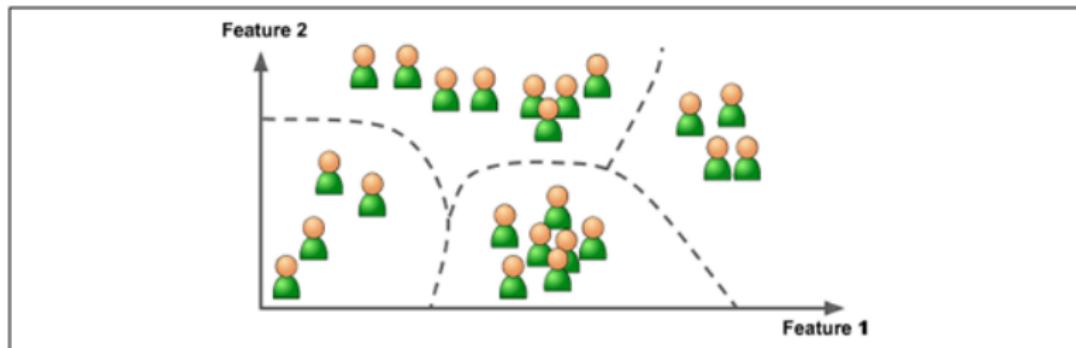


Fuente: Serrano (2020).

# Aprendizaje no supervisado: Clusterización

- Trabaja con **datos no etiquetados**, buscando patrones y estructuras ocultas.
- Objetivo: extraer la mayor cantidad de información posible y agrupar elementos similares, incluso cuando no sabemos de antemano qué categorías existen.
- **Algoritmos de clusterización:** agrupan datos en **clusters** o grupos según su similaridad.
- **Aplicaciones:**
  - Segmentación de mercado (agrupar consumidores).
  - Genética (descubrir familias de genes).
  - Medicina (clasificar pacientes por síntomas).
  - Recomendación de videos o productos.
- **Métodos más usados:**
  - ① K-means clustering.
  - ② DBSCAN.
  - ③ Gaussian Mixture Models.

# Clusterización



Fuente: Gerion (2019).

# Aprendizaje no supervisado: otras ramas

- **Reducción de dimensionalidad:**

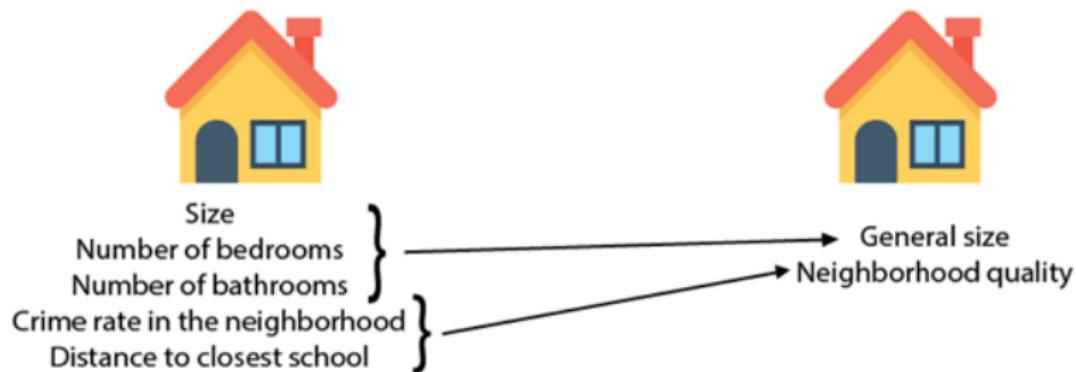
- Simplifica los datos describiéndolos con menos variables sin perder información clave.
- Ejemplos: *PCA*, *t-SNE*, *UMAP*.
- Útil para visualizar bases de datos muy grandes o con miles de variables.

- **Algoritmos generativos:**

- Crean nuevos datos que simulan los originales.
- Usos actuales: generación de **imágenes**, **vídeos**, **música**.
- Ejemplos más conocidos:
  - GANs (Generative Adversarial Networks).
  - Autoencoders variacionales.
  - Restricted Boltzmann Machines (RBMs).
- Nota: este es un campo avanzado, pero importante para entender aplicaciones como **deepfakes** o generación de contenido sintético.

# Reducción de dimensionalidad

## Dimensionality reduction



Fuente: Gerion (2019).

# Aprendizaje por refuerzo

- Objetivo: maximizar la **recompensa acumulada** a lo largo del tiempo.
- Un **agente** aprende mediante prueba y error.
- Recibe **recompensas** o **castigos** según sus acciones.
- Se busca **maximizar** la recompensa acumulada.
- **Ejemplo:** enseñar a un robot a caminar o jugar fútbol.

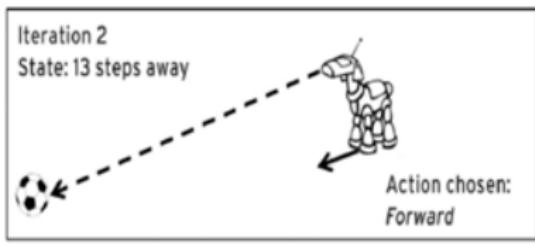
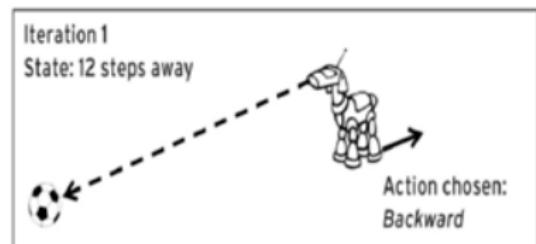
# Aprendizaje por refuerzo: ejemplos (I)

Agente	Entorno	Recompensa	Objetivo
<b>AlphaGo (bot)</b>	Tablero Go/Ajedrez	+1 ganar, 1 perder	Maximizar victorias
<b>Agente RL (Atari)</b>	Videojuego	Puntaje del juego	Maximizar punaje acumulado
<b>Robot bípedo</b>	Superficie física/simulada	Avanzar sin caer	Caminar estable y eficiente
<b>Dron autónomo</b>	Espacio 3D con obstáculos	Trayecto sin colisión / energía	Navegar seguro y eficiente

## Aprendizaje por refuerzo: ejemplos (II)

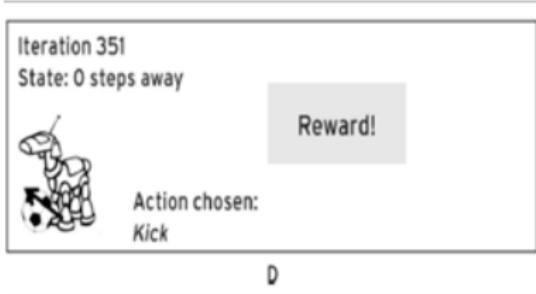
Agente	Entorno	Recompensa	Objetivo
<b>Empresas/consumidores</b>	Mercado (oferta–demanda)	Beneficio, utilidad	Maximizar utilidad esperada
<b>Semáforo inteligente</b>	Red vial urbana	Menor tiempo de espera	Reducir trancos
<b>Tutor virtual</b>	Estudiante/curso	Aciertos, progreso	Maximizar aprendizaje
<b>Planificador de salud</b>	Población/sistema salud	Menos contagios/costo	Minimizar impacto de epidemias

# Aprendizaje por refuerzo



A

B



D

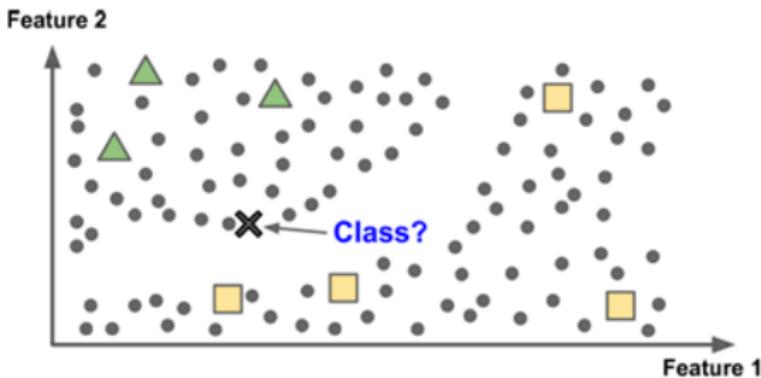
Q-table:

State	0 steps away	1 step away	...	10 steps away	...
Action	Forward	Forward	...	Forward	...
Forward	0	8	...	0	...
Backward	0	0	...	0	...
Kick	10	0	...	0	...

# Aprendizaje semi-supervisado

- Combina pocos datos etiquetados con muchos no etiquetados.
- Útil cuando etiquetar es costoso o lento.
- **Ejemplo:** clasificar imágenes médicas con anotaciones limitadas.

# Aprendizaje semi-supervisado



Fuente: Geron (2019).

# Flujo de un proyecto de ML (I)

- ① **Definir el problema:** ¿Clasificación o regresión? ¿Qué queremos predecir?
- ② **Recolectar los datos:**
  - Desde archivos, bases de datos, APIs, sensores, etc.
  - *Ejemplo: descargas de datos de viviendas, registros médicos, etc.*
- ③ **Explorar y visualizar:**
  - Análisis exploratorio para entender la distribución y relaciones.
  - Visualizaciones para detectar anomalías o patrones.
- ④ **Preparar los datos:**
  - Limpieza (nulos, outliers).
  - Transformaciones (escalado, encoding, imputación).

# Flujo de un proyecto de ML (II)

## ⑤ Entrenar el modelo:

- Aplicar un algoritmo para ajustar parámetros con los datos de entrenamiento.
- *Ejemplos: regresión lineal, random forest, neural networks*

## ⑥ Evaluar el rendimiento:

- Usar métricas apropiadas según el tipo de tarea:
  - Clasificación: *accuracy, precision, recall, F1*.
  - Regresión: *RMSE, MAE, R<sup>2</sup>*.
- Validación cruzada para estimar generalización.

## ⑦ Afinar el modelo (opcional):

- Búsqueda de hiperparámetros: *grid search, random search*.
- Ensamblado de modelos.

## ⑧ Predecir sobre nuevos datos:

- Aplicar el modelo entrenado en producción o en casos reales.
- Monitorear desempeño y actualizar si es necesario.

# Conceptos clave (I): Datos y objetivo

- **Features ( $X$ ):**

- Variables de entrada o características conocidas del problema.
- También llamadas *atributos*.
- Ejemplo: edad, ingresos, historial de compras.
- *Nota: en aprendizaje supervisado, estas se usan para predecir el objetivo.*

- **Target ( $Y$ ):**

- Variable que queremos predecir.
- También conocida como *etiqueta* (clasificación) o *salida* (regresión).
- Ejemplo: ¿hará clic en un anuncio?, ¿precio de una vivienda?

# Conceptos clave (II): Modelo y algoritmo

- **Algoritmo:**

- Procedimiento para entrenar el modelo a partir de los datos.
- Aprende patrones a partir del conjunto de entrenamiento.
- Ejemplos: gradiente descendente, árboles de decisión, KNN.

- **Modelo:**

- Representación matemática de la relación entre  $X$  y  $Y$ .
- Puede ser simple (regresión lineal) o complejo (red neuronal).
- *Ejemplo:*  $\hat{Y} = w_0 + w_1X_1 + w_2X_2$

# Métricas de ajuste en regresión (I)

## Errores comunes

- **Error Absoluto Medio (MAE):** en promedio, ¿cuánto se equivoca el modelo?
  - Ejemplo: digo que una casa vale 105 millones, pero en realidad vale 100. El error es de 5 millones.
- **Error Cuadrático Medio (MSE):** mide el error, pero castigando más los grandes.
  - Ejemplo: equivocarme por 20 millones pesa mucho más que equivocarme por 2.
- **Raíz del Error Cuadrático Medio (RMSE):** similar al anterior, pero expresado en las mismas unidades de la variable.
  - Ejemplo: si predigo ingresos en pesos, el RMSE también estará en pesos.

## Coeficiente de determinación ( $R^2$ )

- Nos dice qué tan bien el modelo logra **explicar la variabilidad de los datos**.
- Se interpreta como un porcentaje entre 0 y 100.
- **Ejemplo:**
  - Si  $R^2 = 0,80$ , significa que el modelo explica el 80 % de los cambios observados en los datos.
  - El otro 20 % queda sin explicar (ruido, factores no incluidos, azar).
- **Idea clave:** mientras más alto sea  $R^2$ , mejor el modelo representa la realidad.

# Matriz de confusión: clasificación

Predicción del modelo		
	Positivo	Negativo
Casos positivos	TP	FN
Casos negativos	FP	TN

- **TP (True Positives)**: casos positivos que el modelo predijo bien.
- **TN (True Negatives)**: casos negativos que el modelo predijo bien.
- **FP (False Positives)**: negativos que el modelo predijo erróneamente como positivos.
- **FN (False Negatives)**: positivos que el modelo predijo erróneamente como negativos.

# Métricas de evaluación: clasificación

- **Accuracy (Exactitud):**

- Porcentaje de predicciones correctas.
- *Fórmula:* Accuracy =  $\frac{TP+TN}{Total}$
- Útil cuando las clases están balanceadas.

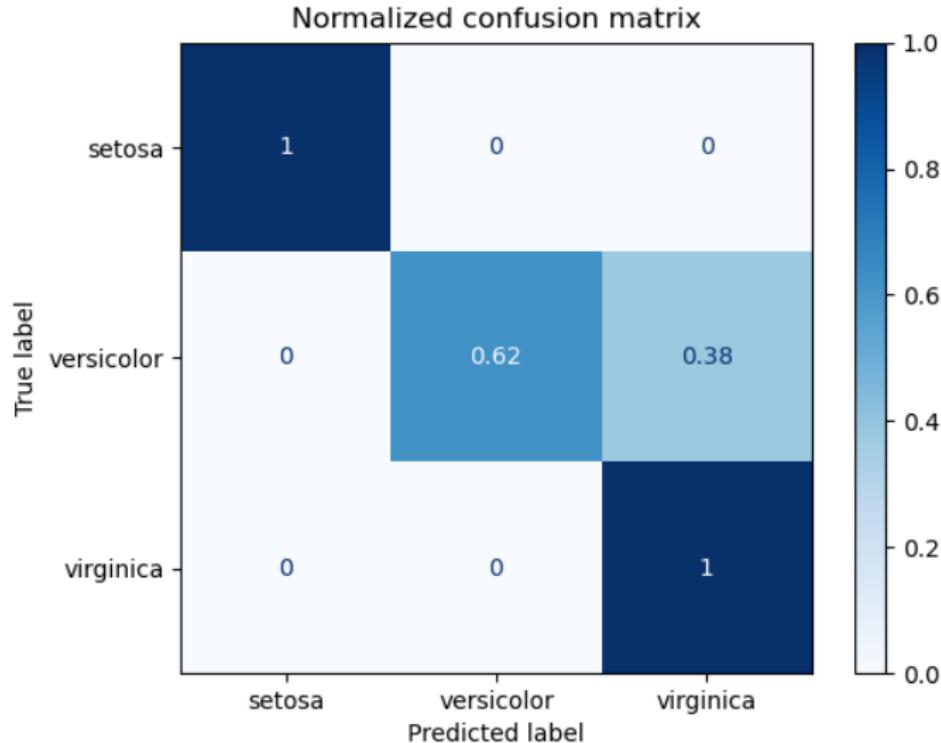
- **Precisión (Precision):**

- Proporción de predicciones positivas que fueron correctas.
- *Fórmula:* Precision =  $\frac{TP}{TP+FP}$
- Alta precisión = pocos falsos positivos.

- **Recall (Sensibilidad):**

- Proporción de positivos reales que fueron detectados.
- *Fórmula:* Recall =  $\frac{TP}{TP+FN}$
- Alto recall = pocos falsos negativos.

# Matriz de confusión en Python



# 2024 presidential election model methodology update

- **Entradas:** encuestas estatales y nacionales ponderadas (tamaño muestral, *pollster rating*).
- **Ajustes a encuestas:** *likely vs. registered voters, house effects, recencia temporal.*
- **Modelo por estado:** mezcla encuestas + **partisan lean, elasticidad** y **Cost of Voting Index.**
- **Fundamentales:** economía e incumbencia con **peso decreciente** (0 % en la víspera).
- **Incertidumbre:** deriva nacional, error nacional del día D, **errores correlacionados entre estados** e idiosincráticos.
- **Simulación Monte Carlo:** miles de elecciones → **probabilidad de victoria**, distribución de *Electoral Votes* y *tipping-point states*. (*p.ej., 30 % 0 %*)

# Fake News Classification and Topic Modeling in Brazilian Portuguese

- **Contexto:** mensajes de *grupos públicos* de WhatsApp durante la campaña presidencial 2018 (Brasil).
- **Identificación de desinformación:** *matching* con un corpus de **notas verificadas** de **6** sitios brasileños de fact-checking.
- **Método (texto):** análisis lingüístico + **clasificación** (SVM / Naive Bayes) sobre mensajes de texto.
- **Difusión:** **más rápida** dentro de cada grupo y **más lenta** entre grupos (fricción intercomunidades).
- **Narrativas:** desinformación **concentrada en menos tópicos** y con lenguaje tipo “cadena” (*alerta, pásalo*).
- **Limitación clave:** solo grupos **públicos**; detección basada en verificados (no cubre *deepfakes*/audio inédito).

# Crimen y violencia urbana

- **Datos:** reportes geolocalizados + fecha/hora (delitos contra la propiedad, etc.).
- **Modelo:** procesos de punto *self-exciting* (Hawkes) que capturan **clustering espacio-temporal** (riesgo sube cerca de eventos recientes).
- **Salida operativa:** “**celdas calientes**” (heatmaps/cuadrículas) para asignar patrullaje preventivo.
- **Evaluación:** ensayos de campo **aleatorizados** comparando despliegue tradicional vs. predictivo → mayor **tasa de aciertos** en zonas priorizadas.
- **Advertencias:** riesgo de **feedback loops** (datos de detenciones delito real), sesgos por sub-reportaje y focalización; necesidad de **auditoría y métricas de equidad**.

Fuente: Mohler, G. et al. (2015). *JASA*.

- **Contexto:** elecciones a la Alcaldía de Medellín (2023); se analizaron **anuncios pagos en Meta** (Facebook e Instagram) de **14 candidatos** y páginas no oficiales.
- **Método:** recopilación vía *Meta Ad Library* + scraping; preprocesamiento de textos e imágenes; clasificación automática con **regresión logística**, Random Forest y XGBoost (precisión ≈97–98 %).
- **Hallazgos:**
  - Los anuncios con alta probabilidad de **desinformación** usaban tono confrontativo, emocional y negativo.
  - Gran parte de la inversión publicitaria se destinó a **microsegmentación** sociodemográfica.
- **Implicaciones:** riesgo de fragmentación del debate público; necesidad de regulación de la publicidad digital y alfabetización mediática.

Fuente: Marín Álvarez, M. (2025). *Publicidad política en redes sociales: Desinformación paga y su impacto en las elecciones a la Alcaldía de Medellín 2023*. Universidad de Antioquia.

# Segmentación de votantes

- **Datos:** encuestas (socio-demo, territorio, actitudes, medios, intención/recuerdo de voto).
- **Método:** estandarizar/imputar → reducción (PCA/UMAP para visualizar) → **clustering** (k-means/mixto).
- **Elección de  $k$ :** *silhouette*, gap, estabilidad por *bootstrap*.
- **Salida:** tipologías interpretables (p.ej., *liberales urbanos*, *conservadores rurales*, *indecisos*).
- **Uso de campaña:** *microsegmentación* de mensajes, targeting territorial, diseño de encuestas de seguimiento.
- **Validación/alertas:** robustez de clústeres en submuestras/ondas; resultados dependen de variables y  $k$ ; riesgo de *overclaiming*.

Fuente: Grimmer, J., Roberts, M. & Stewart, B. (2021). *Annual Review of Political Science*.

# Discursos parlamentarios (EE. UU.)

- **Datos:** 1M intervenciones (Congressional Record); preprocesamiento: tokenizar, *stopwords*, lematizar.
- **Método: topic modeling** (LDA/STM); elegir  $K$  y etiquetar temas por *top-words*; series de *topic proportions* en el tiempo.
- **Resultados:** agendas dominantes y picos por eventos (p.ej., seguridad post-9/11, salud/Obamacare, crisis económica).
- **Análisis comparado:** diferencias por partido/cámara/periodo; co-ocurrencia de temas y ciclos mediáticos.
- **Validación:** coherencia temática (UMass/ $C_v$ ) + juicio experto; sensibilidad a  $K$  y priors.
- **Limitaciones:** mezcla de temas en discursos largos, cambios léxicos, no captura ironía/sarcasmo.

Fuente: Quinn, K. et al. (2010). *American Journal of Political Science*.

## ¿Por qué usar ML?

- **Detecta patrones complejos:** identifica relaciones no evidentes para el ser humano (e.g., predicción de fallos en maquinaria).
- **Escala a grandes volúmenes de datos:** puede analizar millones de registros rápidamente (e.g., análisis de redes sociales, sensores IoT).
- **Adapta soluciones a nuevos contextos:** el modelo puede actualizarse con datos recientes sin reprogramar todo el sistema (e.g., recomendaciones personalizadas).
- **Automatiza tareas repetitivas:** libera tiempo humano para tareas creativas (e.g., clasificación de correos, reconocimiento de imágenes).
- **Aprende con la experiencia:** mejora conforme ve más datos (e.g., modelos de predicción meteorológica, traducción automática).

# Limitaciones y riesgos del Machine Learning

## Desafíos y consideraciones éticas

- **Sesgos en los datos:**
  - Si los datos están sesgados, el modelo perpetúa y amplifica injusticias.
  - Ejemplo: discriminación algorítmica en sistemas judiciales o bancarios.
- **Falta de interpretabilidad:**
  - Modelos complejos (como redes neuronales profundas) son cajas negras.
  - Esto dificulta auditoría, confianza y uso en contextos sensibles.
- **Sobreajuste (overfitting):**
  - El modelo aprende demasiado bien los datos de entrenamiento, pero falla con nuevos datos.
- **Riesgo de mal uso político o social:**
  - Manipulación de información, vigilancia masiva, propaganda automatizada.

## Pregunta final

¿Dónde ves oportunidades para usar ML en tu campo de investigación en ciencias políticas?