HENRY

Data Science

Hands on - M4L1 | Introducción al Machine Learning

Objetivos

- Distinguir los conceptos clave de IA, ML, DL y GenAI, diferenciando sus aplicaciones y alcances en la resolución de problemas complejos.
- Clasificar los tipos de aprendizaje en machine learning, explorando supervisado, no supervisado y por refuerzo, junto con sus metodologías clave.
- Aplicar técnicas de ingeniería de características, evaluando estrategias de transformación, selección e imputación para mejorar la calidad de los modelos.

Agenda

COMENCEMOS →

#TEMAS

.01 IA vs ML vs DL vs GenAI

.02 Tipos de tareas de aprendizaje

.03 Elementos básicos en un modelo de ML

.04 Ingeniería de características

.05 Homework

<-->

¿Qué vimos en la lecture?

<01>

IA vs ML vs DL vs GenAI

IA vs ML vs DL vs GenAI

Idea clave: en Data Science se usan IA, ML, DL y GenAI como sinónimos, pero **no lo son**.

- DS: limpieza, EDA, visualización, modelos; no todo es IA.
- IA ⊃ ML ⊃ DL; **GenAI** es una especialización de **DL** que **genera contenido**.
- Aclarar conceptos evita aplicar técnicas inadecuadas.

Definiciones y ejemplos: IA y ML

★ **IA:** disciplina amplia que imita procesos de la inteligencia humana (razonar, planificar, decidir). **No todo es aprendizaje automático.**

Ejemplos:

- Sistemas expertos (diagnóstico con reglas)
- Chatbots con respuestas programadas
- Robótica autónoma
- Reglas de negocio para recomendar

★ **ML:** subcampo de IA; **aprende patrones a partir de datos** para predecir, clasificar o detectar anomalías. Está enfocado en **datos tabulares/estructurados**.

Ejemplos:

- churn, fraude, spam, scoring crediticio.

Definiciones y ejemplos: DL y GenAI

★ **DL:** redes neuronales profundas; datos **no estructurados** (imágenes, voz, texto).

Ejemplos:

- Visión médica, traducción, asistentes de voz, speech-to-text

★ **GenAI (dentro de DL): crea** contenido (texto, imágenes, audio, video, código).

Ejemplos:

- Informes automáticos, imágenes de marketing, copilotos, chatbots avanzados, síntesis de voz/video.

ML tradicional vs GenAI

Son complementarios entre sí.

- **ML tradicional:** datos tabulares; **explicabilidad** alta para contextos sensibles (salud/finanzas).
    - Ej.: regresión logística para explicar riesgo de diabetes.
- **GenAI:** generar contenido y trabajar con **datos heterogéneos**; ideal para interacción (chatbots, informes).
    - Ej.: carta personalizada al paciente con recomendaciones.

Comparativa: IA, ML, DL y GenAI

	IA	ML	DL	GenAI
Nivel Jerárquico	Marco	Subcampo	Subcampo de ML	Aplicación de DL para generar
Objetivo	Imitar inteligencia	Aprender patrones	Resolver con múltiples capas	Generar contenido
Datos	Variados	Tabulares	Masivos/no estructurados	Multimodales
Costo/ explicabilidad	De simple a complejo	Más económico/ interpretables	Alto cómputo/menos explicable	Cómputo intensivo/difícil de explicar

<02>

Tipos de tareas de aprendizaje

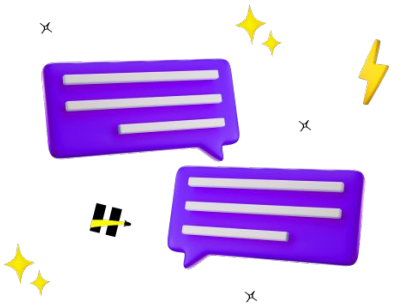
Tipos de tareas en ML

Supervisado, no supervisado y por refuerzo: tres enfoques según etiquetas, estructura y feedback.

Resuelven problemas distintos y producen resultados diferentes.

Aprendizaje Supervisado (regresión vs. clasificación)

- **Regresión (valor numérico):** precios de viviendas, demanda de energía, ventas futuras.
- **HealthPredict:** predecir presión sistólica (p. ej., 138 mmHg) con IMC, glucosa, edad.
  Clasificación (etiqueta): diagnóstico, spam, fraude.

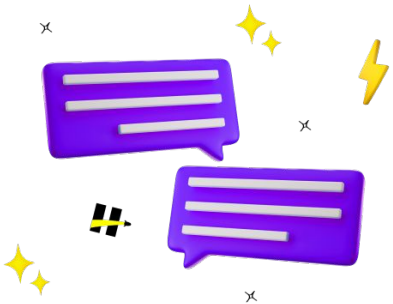


Aprendizaje No supervisado

- Sin etiquetas; descubre **patrones**.
- Clustering (segmentos de pacientes), reducción de dimensionalidad (PCA), recomendadores.
- **HealthPredict:** perfiles de pacientes; PCA para visualizar 15→2 variables.

Aprendizaje por refuerzo

- Agente aprende por **recompensas/penalizaciones**.
- Robótica, AlphaGo, precios dinámicos.
- **HealthPredict:** aprende qué plan de ejercicio funciona mejor por perfil.



<03>

Elementos básicos en un modelo de ML

Elementos básicos de un modelo

Idea guía: predecir precio de un apartamento → formalizar intuiciones.

- ★ **Features:** m^2, habitaciones, ubicación
- ★ **Target:** precio.
- ★ **Muestra:** cada apartamento en el dataset.
- ★ **Algoritmo:** p. ej., regresión lineal.
- ★ **Función de costo:** error entre real y estimado.
- ★ **Parámetros:** pesos e intercepto (se aprenden).
- ★ **Hiperparámetros:** p. ej., profundidad de árbol.

Modelos + ajuste

	Paramétricos	No paramétricos
Parámetros	Fijo	Crece con datos
Complejidad/velocidad/ interpretabilidad	Simple/rápido/alto	Flexible/lento/bajo
Ejemplos	Regresión/Naive Bayes	k-NN/árboles/random forest/SVM con kernels
Costo/ explicabilidad	De simple a complejo	Más económico/ interpretables

Underfitting vs. Overfitting

- **Subajuste:** modelo **simple** (ej., solo habitaciones) → error alto.
- **Sobreajuste:** modelo **excesivamente complejo** (árbol muy profundo) → memoriza; falla en nuevos casos.

<04>

Ingeniería de características

Ingeniería de características

Ciclo de datos → ML: ingesta → transformación → almacenamiento (lakes/warehouses) → consumo en features.

Transformaciones numéricas

- **MinMax** [0,1] (k-NN, SVM, redes; rango acotado).
- **Standard** (media 0, var 1; regresión/logistic, PCA; sin outliers extremos).
- **Robust** (mediana/IQR; con outliers).
- **log1p** (variables muy sesgadas; usar antes/junto con escalado).

Categóricas: one-hot, label encoding, **target encoding** (cuidar leakage: solo en train/validación cruzada).

Selección: conocimiento de dominio; Lasso; importancia en Random Forest; selección progresiva/retro; filtros (correlación alta, baja variabilidad).

Imputación: media/mediana/moda; forward/backward fill; modelos (KNN/regresión); si faltantes ~40% y sistemáticos, considerar descartar.

<DATA SCIENCE/>

Vayamos a la práctica

Homework

Consigna

La empresa **RetailBoost** quiere construir un modelo que prediga si un cliente realizará una compra (`compró=1` / `no compró=0`) en función de información básica como edad, ingresos, región y frecuencia de visitas al sitio web.

Tu tarea consiste en realizar un **EDA (Exploratory Data Analysis)** con Python (usando **Pandas + Numpy**) para preparar y seleccionar las características más relevantes que serán utilizadas en un futuro modelo de clasificación.

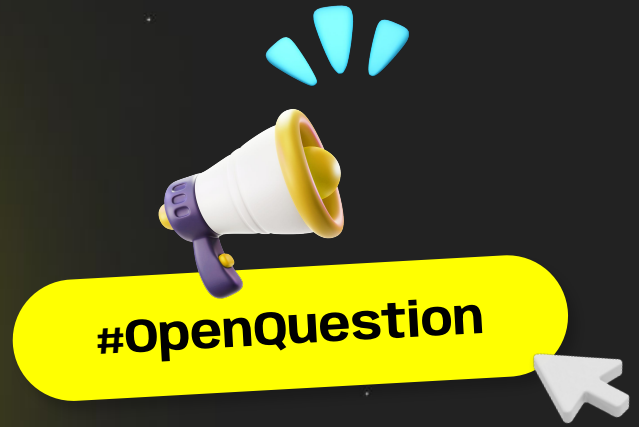
Tareas a realizar

1. **Carga y exploración inicial del dataset** (`retailboost_customers.csv`).
2. **Identificación de distribuciones** (histogramas o descripciones estadísticas de variables numéricas y categóricas).
3. **Limpieza de datos nulos**: imputar valores o descartar columnas según corresponda.
4. **Correlaciones**: calcular la matriz de correlación entre variables numéricas y discutir posibles redundancias.
5. **Transformaciones de características**:
    - Normalización o estandarización en variables numéricas.
    - Codificación de variables categóricas (One-Hot Encoding o Label Encoding).
6. **Selección preliminar de features** en base al análisis anterior.

Extra credits

- Aplicar un método de **selección automática de características** (ej. `SelectKBest`, `Lasso` o importancia de variables con árboles).
- Documentar hallazgos principales en un **informe breve** (Markdown o PDF).

HENRY



¿Preguntas?

HENRY

¡Muchas gracias!

→ soyhenry.com