

UNIVERSIDAD TECNOLÓGICA DE CHIHUAHUA

Desarrollo y Gestión de Software



Extracción de Conocimiento en Bases de Datos

**I.3. Reporte de investigación de los tipos de aplicaciones,
procesamiento y herramientas para inteligencia artificial,
machine learning, data mining y big data.**

IDGS91N

PRESENTA:

T.S.U. Hugo Uriel Chaparro Estrada

DOCENTE:

Enrique Mascote

Chihuahua, Chih., 6 jun 2025

Introducción.....	3
1. Inteligencia Artificial (IA).....	4
Tipos de aplicaciones.....	4
Procesamiento: batch vs. streaming en IA.....	5
Herramientas y tecnologías (IA).....	6 2.
Machine Learning (ML).....	7 Tipos
de aplicaciones.....	7
Procesamiento: batch vs. streaming en ML.....	8
Herramientas y tecnologías (ML).....	8 3.
Minería de Datos (DM).....	9 Tipos
de aplicaciones.....	9
Procesamiento: batch vs. streaming en DM.....	10
Herramientas y tecnologías (DM).....	10 4. Big
Data.....	12 Tipos de
aplicaciones.....	12 Procesamiento:
batch vs. streaming en Big Data.....	13 Herramientas y
tecnologías (Big Data).....	13
Conclusión.....	15
Referencias Bibliográficas.....	16

Introducción

La **inteligencia artificial (IA)**, **aprendizaje automático (ML)**, **minería de datos (DM)** y **big data** son cuatro dominios estrechamente relacionados que comparten objetivos de extracción de conocimiento a partir de los datos, pero se diferencian en su enfoque y escala. En este reporte compararemos cada dominio: sus definiciones, las aplicaciones más frecuentes (con ejemplos reales), los modos de procesamiento (batch vs. streaming) y las herramientas principales asociadas. La investigación se basa en múltiples fuentes actualizadas, incluyendo artículos de IBM, AWS, blogs técnicos y medios especializados.

1. Inteligencia Artificial (IA)

Definición breve: La IA es la tecnología que permite a las máquinas simular capacidades cognitivas humanas, como el aprendizaje, la resolución de problemas o la toma de decisiones. Según IBM, la IA es una tecnología que permite a ordenadores y máquinas simular capacidades humanas. Por ejemplo, un coche autónomo es una aplicación paradigmática de IA que integra visión, planificación y control.

Tipos de aplicaciones

- **Visión por computador:** Uso de IA para procesar e interpretar imágenes o vídeo. Por ejemplo, **Tesla** emplea redes neuronales para que sus vehículos detecten carriles, peatones y señales de tránsito en tiempo real. En general, los sistemas de visión por computador analizan datos de cámaras y sensores para guiar robots o vehículos de forma autónoma. (*Imagen sugerida: diagrama de un coche autónomo con sensores de visión*).
- **Procesamiento del Lenguaje Natural (PLN):** Análisis y generación de lenguaje humano por máquinas. Por ejemplo, **Apple Siri** y **Amazon Alexa** utilizan IA para convertir voz en texto y comprender comandos, proporcionando respuestas útiles mediante reconocimiento de voz y análisis semántico. Un ejemplo concreto es que los asistentes virtuales entienden el tono y el contexto para dirigir consultas al servicio adecuado.
- **Sistemas de recomendación:** Plataformas que sugieren productos o contenidos basados en preferencias del usuario. Empresas como **Netflix** y **Amazon** analizan el historial de visualización o compras para personalizar recomendaciones. Netflix, por ejemplo, “entiende matices y ofrece recomendaciones como si un amigo te conociera muy bien”, mientras que Amazon ajusta dinámicamente precios y sugiere productos según patrones de compra.
- **IA generativa:** Creación de contenido original mediante IA. Un caso emblemático es **DALL·E de OpenAI**, que genera imágenes desde descripciones de texto.

Xataka describe: “DALL·E es un sistema de IA que genera imágenes a partir de texto”, permitiendo, por ejemplo, crear ilustraciones a pedido. De modo similar, modelos como **ChatGPT** generan texto coherente, aunque en este reporte nos centramos en la perspectiva de herramienta.

- **Robótica autónoma:** Robots y sistemas autónomos que perciben y actúan sobre el mundo físico. Por ejemplo, **Amazon Robotics** utiliza robots móviles (como Proteus y Sparrow) que navegan almacenes, identifican y manipulan productos sin intervención humana. Wired destaca que el robot *Sparrow* puede “tomar productos individuales de los cubículos de almacenamiento” automatizando tareas manuales. Estos robots incorporan visión por computador y planificación basados en IA.

Procesamiento: batch vs. streaming en IA

En IA las operaciones intensivas en datos se gestionan en modo *batch* o en tiempo real (*streaming*), según el caso. El **entrenamiento de modelos** suele hacerse en batch: se procesan grandes volúmenes de datos históricos de manera offline (por ejemplo, usando GPUs en centros de cómputo). En cambio, la **inferencia** (uso del modelo entrenado) puede ser *en streaming* cuando se requiere respuesta inmediata. IBM describe varios modos de inferencia: la inferencia dinámica (online) produce respuestas de baja latencia para casos como ChatGPT, mientras que la inferencia por lotes (batch) genera predicciones offline para grandes conjuntos de datos (p.ej., informes periódicos de marketing). La inferencia en streaming se emplea en canalizaciones continuas de datos (por ejemplo, IoT o monitorización), donde la IA predice el vuelo basándose en flujos de sensores. En resumen, el *batch* en IA se usa típicamente para entrenamiento masivo o análisis histórico, y el *streaming* para inferencia o detección en tiempo real.

Herramientas y tecnologías (IA)

- **TensorFlow** – Biblioteca de código abierto de Google para construir y entrenar modelos de redes neuronales. TensorFlow proporciona herramientas flexibles para

aprendizaje profundo y se ejecuta en CPU/GPU. Según Unite.AI, fue creado por Google Brain y ofrece “una amplia gama de herramientas, librerías y recursos” para modelos de deep learning. Se usa en visión por computador, NLP y visión industrial. (*Imagen sugerida: logo de TensorFlow*).

- **PyTorch** – Framework open source (de Meta/Facebook AI) para deep learning. PyTorch destaca por su flexibilidad y facilidad de prototipado de redes neuronales. Unite.AI señala que PyTorch es muy popular y abierto, pensado para aprendizaje profundo desde 2016. Es muy usado en investigación y producción (p.ej., en sistemas de traducción automática). (*Imagen sugerida: logo de PyTorch*).
- **OpenCV** – Biblioteca de visión artificial con cientos de algoritmos (detección de caras, clasificación de objetos, calibración de cámaras, etc.). Se emplea para procesar imágenes y video en tiempo real. Es clave en IA de visión embarcada y aplicaciones móviles.
- **Hugging Face Transformers** – Plataforma y librería (Python) especializada en modelos de lenguaje (BERT, GPT, etc.). Ofrece modelos pre-entrenados de última generación para PLN, facilitando construir chatbots, análisis de sentimientos y traducción automática.
- **IBM Watson (Watsonx)** – Plataforma de IA en la nube de IBM que integra múltiples servicios (NLU, visión, asistentes virtuales). Por ejemplo, **Watson Assistant** permite crear chatbots empresariales multilingües que responden automáticamente, como se ve en aplicaciones de atención al cliente bancario.
- **Cloud AI Services** – Servicios gestionados en la nube (Google Vertex AI, Azure Cognitive Services, AWS SageMaker) que ofrecen APIs de visión, lenguaje, recomendación y entrenamiento de modelos escalables para empresas.

2. Machine Learning (ML).

Tipos de aplicaciones

- **Recomendaciones y analítica de marketing:** Como IA, los sistemas de recomendación también se categorizan en ML. Amazon, Netflix y otros sitios usan ML para analizar gustos y comportamientos, ofreciendo sugerencias personalizadas. Por ejemplo, en marketing, el ML permite *predecir la preferencia de un cliente* y adaptar campañas. IBM destaca que empresas de e-commerce usan el ML para contactar a usuarios que abandonan carritos de compra, y que los motores de recomendación de Netflix y Amazon se basan en ML.
- **Servicio al cliente y chatbots:** El ML impulsa asistentes virtuales en soporte al cliente. Por ejemplo, muchos bancos y tiendas en línea emplean chatbots que usan PLN (una técnica de ML) para entender consultas de texto, detectar sentimientos y dirigir usuarios al agente adecuado. IBM documenta casos de chatbots que responden eficientemente a cerca del 96% de las preguntas de los clientes.
- **Asistentes de voz/personales:** Sistemas como **Alexa** y **Siri** son casos de ML en PLN. Al igual que en IA, estos asistentes convierten voz en texto y generan respuestas. IBM señala que estos asistentes utilizan ML para realizar tareas, reconociendo el tono y el contenido de la petición. El ML también se usa en reconocimiento de imágenes (p.ej., identificar objetos o rostros en fotos) y en detección de fraudes bancarios (no citado explícitamente aquí).
- **Análisis predictivo/finanzas:** Instituciones financieras usan ML para predecir tendencias (tasa de rotación de clientes, precios de activos) y detectar anomalías (fraude). Aunque no lo cubre directamente una fuente citada aquí, es uno de los usos más comunes de ML en la industria.

Procesamiento: batch vs. streaming en ML

En ML, el **entrenamiento de modelos** suele hacerse en modo batch con datasets históricos completos. Esto significa procesar offline grandes cantidades de datos, actualizar el modelo periódicamente. Sin embargo, también existe *aprendizaje online* (en streaming), donde el modelo se va actualizando incrementalmente con datos nuevos sin reentrenar desde cero. Por ejemplo, en escenarios donde los datos cambian rápidamente, se aplican variantes de ML online que aprenden sobre la marcha.

La **inferencia (predicción)** en ML puede configurarse en batch o streaming, al igual que en IA. El procesamiento por lotes se emplea para generar predicciones masivas offline (por ejemplo, segmentar clientes usando datos acumulados). En cambio, el streaming ML se usa cuando se requieren predicciones en tiempo real, como detección de fraude en transacciones al instante o motores de recomendación que reaccionan en vivo. IBM describe que en *inferencia dinámica* (en línea) se exige baja latencia (e.g., chatbots con ML), y que el procesamiento de flujos (streaming) se usa para canalizar datos de sensores e IoT en tiempo real. En resumen, **batch** es común para análisis masivo o entrenamientos regulares, mientras que **streaming** permite respuestas inmediatas y adaptación continua.

Herramientas y tecnologías (ML)

- **Scikit-learn:** Biblioteca de Python de código abierto para ML clásico. Ofrece implementaciones eficientes de algoritmos de regresión, clasificación, clustering y reducción de dimensionalidad. Según datascientest, “proporciona acceso a versiones eficaces de muchos algoritmos comunes” y una API uniforme. Se usa ampliamente para prototipos y competencias de ML.
- **TensorFlow:** (Ver IA arriba) También es clave en ML general, no solo IA. Además de su capacidad para deep learning, permite crear pipelines de datos con *TensorFlow Data* y modelado de redes neuronales clásicas.
- **PyTorch:** (Ver IA) Framework preferido en investigación ML y prototipado rápido.
- **XGBoost:** Biblioteca optimizada de boosteo de árboles de decisión. Según su documentación, “es una biblioteca distribuida de gradient boosting diseñada para ser muy eficiente, flexible y portátil”, adecuada para problemas de clasificación y regresión. XGBoost es popular en competiciones de ML por su alta precisión y velocidad.
- **R (RStudio):** Lenguaje y entorno estadístico open source. R incluye paquetes de ML (caret, randomForest, etc.) y visualización avanzada. Como señalan expertos, R es un estándar en análisis exploratorio y modelado estadístico. Por ejemplo, bancos y universidades lo usan para análisis predictivo y minería de datos básicos.

- **Otras herramientas:** pandas (Python, para manipulación de datos), Jupyter Notebooks (entorno interactivo), MATLAB (en entornos académicos), y plataformas cloud ML (Azure ML, AWS SageMaker). Estas facilitan la experimentación de modelos ML y la gestión de datos.

3. Minería de Datos (DM)

Definición breve: La minería de datos es el proceso asistido por computadora de **explorar grandes volúmenes de datos** para descubrir patrones, relaciones ocultas o conocimiento útil. Según AWS, la DM “transforma datos en bruto en conocimiento práctico” al identificar correlaciones y anomalías. No debe confundirse con ML: la minería de datos abarca técnicas (incluyendo ML) enfocadas en **descubrimiento** y segmentación de la información.

Tipos de aplicaciones

- **Predicción y modelado estadístico:** Por ejemplo, predecir las ventas de un nuevo producto en función del presupuesto publicitario o la evolución del mercado de valores. Estas tareas usan regresión, redes neuronales u otros modelos para estimar valores continuos..
- **Detección de anomalías/fraude:** Un uso clásico es en finanzas: identificar transacciones sospechosas analizando patrones anómalos en grandes bases de datos. ITmasters destaca que la DM “identificar patrones inusuales” es útil para la detección de fraudes, ya que las anomalías suelen indicar actividad fraudulenta.
- **Segmentación y personalización:** La minería de datos ayuda a segmentar clientes y optimizar ofertas. En telecomunicaciones, por ejemplo, se analizan patrones de uso del cliente para **personalizar planes y servicios**. De forma similar, los datos históricos de compra se usan para crear perfiles y agrupar usuarios con comportamiento parecido.
- **Análisis de series temporales:** Predecir tendencias de consumo o eventos según

datos temporales (e.g., demanda eléctrica o afluencia de clientes). Las técnicas de minería incluyen análisis de series y clustering secuencial.

- **Recomendaciones comerciales:** Aunque clasificado a veces como ML, se puede ver como DM cuando se explora la base de datos de compras para encontrar *reglas de asociación* (market basket analysis) que expliquen qué productos se compran juntos frecuentemente.

Procesamiento: batch vs. streaming en DM

Tradicionalmente, la minería de datos se realiza en modo *batch*. Se analizan offline grandes conjuntos históricos para extraer patrones globales. Por ejemplo, generar informes de segmentación de clientes a partir de la base de datos anual de ventas. El procesamiento por lotes aprovecha herramientas como Hadoop o Data Warehouses para computar sobre los datos enteros. No obstante, existe *minería de datos en streaming* para flujos continuos: se analiza en tiempo real datos de sensores o logs para detectar rápidamente eventos relevantes. IBM explica que el streaming se utiliza en pipelines de datos (IoT, tráfico de sensores) con predicciones continuas. En términos generales, el procesamiento por lotes realiza tareas de minería escalada y programada, mientras que el streaming permite *minería incremental*, actuando sobre cada nuevo registro en cuanto llega. La elección depende de la necesidad de latencia; por ejemplo, detección de fraudes en tiempo real usaría streaming, mientras que análisis histórico de clientes usaría batch.

Herramientas y tecnologías (DM)

- **Weka:** Plataforma Java open source de la Universidad de Waikato. Weka incluye una colección gráfica de algoritmos de DM (clustering, clasificación, reglas de asociación). Su interfaz facilita experimentar con preprocesamiento y modelado sin programar. Wikipedia indica que “Weka soporta varias tareas estándar de minería de datos, especialmente preprocesamiento, clustering y clasificación”. Se usa para educación e investigación en minería de datos.

- **RapidMiner**: Herramienta comercial/ open source de análisis de datos. Según su wiki, permite desarrollar procesos de minería encadenando “operadores” en un entorno gráfico. RapidMiner provee cientos de operadores para preprocesamiento y visualización, e incluso integra algoritmos de Weka. Es muy popular en empresas para prototipado rápido de flujos de DM.
- **KNIME**: Plataforma visual de análisis y minería de datos open source. Similar a RapidMiner, permite diseñar flujos de trabajo con componentes para ETL, machine learning y reporting. Es extensible con plugins y se integra bien con Python/R.
- **Orange**: Herramienta Python de minería visual. Ofrece “widgets” de arrastrar y soltar para análisis, gráficos y ML. Es útil para visualización interactiva y aprendizaje automático básico.
- **SAS Enterprise Miner** y **IBM SPSS Modeler**: Suites comerciales ampliamente usadas en corporaciones para DM. Incluyen interfaces gráficas y algoritmos avanzados de minería (regresión, árboles, redes neuronales) con soporte empresarial.
- **R y Python (scikit-learn, pandas)**: Lenguajes de análisis de datos con librerías de minería (ej. `arules` en R, `mlxtend` en Python). Son ampliamente usados para scripts de minería de datos personalizados.
- **Bases de datos analíticas**: Herramientas como **RapidMiner**, **Orange** y software BI (Tableau, Power BI) se emplean en el contexto de minería de datos para explorar resultados.

4. Big Data

Definición breve: *Big Data* se refiere a volúmenes de datos tan grandes, variados y con tanta velocidad de generación que las técnicas tradicionales no son suficientes. Se define por las “3V”: *Volumen* (grandes cantidades de datos), *Variedad* (orígenes heterogéneos) y

Velocidad (datos que llegan rápidamente). Según Nucleoo, “los big data son datos que contienen gran variedad, volúmenes crecientes y alta velocidad”, y en conjunto “big data son conjuntos de datos muy grandes y complejos... que las herramientas tradicionales no pueden manejarlos”.

Tipos de aplicaciones

- **Analítica de usuarios y recomendación (Netflix, Amazon):** Grandes plataformas de streaming y comercio analizan comportamientos de decenas de millones de usuarios. Netflix procesa terabytes de datos de visionado diario para personalizar recomendaciones de contenido. Amazon, por su parte, utiliza *big data* para segmentar usuarios y ajustar precios dinámicamente cada pocos minutos. Estos sistemas basan sus modelos en infraestructuras big data escalables para manejar el flujo continuo de interacción (por ejemplo, “el 35% de lo que compran en Amazon proviene de recomendaciones basadas en algoritmos”).
- **Personalización y retail (Starbucks):** Compañías con operaciones masivas usan big data para optimizar la experiencia del cliente. Starbucks analiza **100 millones de transacciones semanales** a través de su app y programa de recompensas. Con ello adapta menús por ubicación y preferencias, y envía ofertas personalizadas a clientes inactivos. El análisis incluye factores externos (clima, hora) para predecir comportamientos.
- **Sector financiero:** Los bancos aplican big data para detección de fraude masivo, análisis de riesgos y cumplimiento. Procesan enormes logs de transacciones en tiempo real. Empresas fintech también usan big data para crear perfiles crediticios alternativos.
- **Internet de las Cosas (IoT) y redes sociales:** Big data se usa en smart cities y manufactura (sensores que generan datos en continuo) y en plataformas sociales (análisis de millones de tuits o fotos al instante).
- **Investigación científica:** Campos como la genómica o astronomía generan petabytes diarios. Big data tools permiten procesar esas cantidades para avances

en salud o clima.

Procesamiento: batch vs. streaming en Big Data

El procesamiento de big data puede ser **batch** o **streaming** según la tecnología. El procesamiento *batch* (por lotes) procesa grandes volúmenes acumulados, típico de sistemas basados en Hadoop MapReduce o consultas en data warehouses. Por ejemplo, usar Hive o Pig para ejecutar análisis sobre datos históricos. La computación en batch es “programada en intervalos” y eficaz para cálculos a gran escala. En contraste, el procesamiento *streaming* (o tiempo real) es continuo. Herramientas como **Apache Kafka** (sistema de mensajería), **Apache Flink** o **Spark Streaming** permiten analizar flujos de datos a medida que llegan. DataCamp explica que “a diferencia del procesamiento por lotes, que recopila y procesa datos en trozos discretos, el procesamiento por flujos trata los datos de forma continua e incremental”. En big data, se elige el modo según la aplicación: p.ej., registro de logs en tiempo real o detección de anomalías en streaming, vs. informes globales offline. Apache Spark ejemplifica esta dualidad: es un motor unificado de **cálculo rápido** en clústeres, diseñado tanto para *streaming* como para procesamiento por lotes.

Herramientas y tecnologías (Big Data)

- **Apache Hadoop:** Framework open source para almacenar y procesar grandes datos distribuidos. Incluye HDFS (almacenamiento distribuido) y MapReduce (cómputo paralelo). AWS describe Hadoop como un marco para datos de *gigabytes hasta petabytes*, creando clústeres de computadoras para análisis masivo en paralelo. Hadoop es la base de muchas arquitecturas big data.
- **Apache Spark:** Motor de análisis en memoria ultra-rápido. Spark soporta procesamiento batch y streaming, además de SQL, machine learning y análisis gráfico. IBM señala que “Spark es un motor de procesamiento de datos de código abierto ultrarrápido” diseñado para escalabilidad en big data, particularmente para datos en streaming y ML. Spark procesa datos mucho más rápido que MapReduce al mantenerlos en memoria.
- **Apache Kafka:** Plataforma de streaming distribuido para pipelines de datos. Permite

publicar, almacenar y procesar flujos de eventos en tiempo real. Se usa junto con Spark o Flink para crear aplicaciones big data en tiempo real (p.ej., monitoreo de métricas).

- **Apache Flink:** Framework open source para procesamiento de stream de eventos con alta tolerancia y baja latencia. Ideal para análisis continuos en clústeres de producción.
- **Bases NoSQL distribuidas:** Como **Cassandra**, **HBase** o **MongoDB**, diseñadas para almacenar grandes volúmenes de datos con alta disponibilidad. Se usan para *data lakes* o *data warehouses* en entornos de big data. Por ejemplo, Cassandra es usada por empresas que requieren lectura/escritura de alta escala (Netflix, Instagram).
- **Hive / Impala:** Sistemas de query SQL sobre Hadoop, que permiten hacer consultas analíticas en big data. Facilitan la migración de conocimientos de SQL tradicional al ecosistema Hadoop.
- **Elasticsearch / Solr:** Motores de búsqueda de texto usados como almacenes analíticos secundarios para big data (logs de aplicaciones, auditoría). Permiten búsquedas rápidas en volúmenes masivos de registros.

Conclusión

Estos cuatro dominios presentan solapamientos importantes, pero también diferencias claras. IA es el campo más amplio; ML es un subconjunto de IA dedicado a modelos que aprenden de datos, mientras que minería de datos se enfoca en descubrir patrones ocultos en los datos, usando a menudo técnicas de ML para análisis exploratorio. Por ejemplo, un algoritmo de clustering utilizado en minería de datos es esencialmente un método de ML. Big Data, en cambio, no es tanto una técnica sino una condición de escala: se ocupa de la infraestructura y las herramientas necesarias para manejar datos que desafían las capacidades tradicionales.

Así, los dominios se solapan en las técnicas: muchas herramientas (p.ej., TensorFlow, Spark) sirven tanto para ML como para IA, y los algoritmos de ML se usan en DM. Pero

difieren en los objetivos: IA incluye razonamiento y creatividad automatizada (e.g., IA generativa), ML se centra en predicciones basadas en datos, DM busca insights y reglas desconocidas, y Big Data presta atención especial a la arquitectura de datos y al rendimiento.

La elección de la herramienta y del modo de procesamiento adecuado es crítica. Usar procesamiento *batch* o *streaming* depende de los requisitos de latencia y volumen: como advierte DataCamp, batch y streaming son enfoques complementarios para aprovechar sus puntos fuertes en diferentes escenarios. Por ejemplo, para entrenar grandes modelos de ML con historial de datos se prefiere batch, mientras que para inferencia en tiempo real (asistentes virtuales o detección de fraudes) se requiere streaming. Igualmente, elegir entre Hadoop, Spark o bases NoSQL depende del caso de uso; Spark puede ser mejor para computaciones rápidas en memoria, mientras que Hadoop MapReduce es robusto para análisis en lote masivo. Una mala elección (por ejemplo, procesar en batch datos que requieren acción inmediata) puede resultar en latencias inaceptables o recursos desaprovechados.

En resumen, aunque IA, ML, DM y Big Data se benefician mutuamente y a menudo emplean tecnologías similares, cada uno enfatiza aspectos distintos del ciclo del dato. Entender sus fronteras y elegir apropiadamente el procesamiento (*batch* vs. *streaming*) y la plataforma (frameworks y herramientas) es esencial para maximizar el rendimiento y obtener resultados útiles de los datos masivos de hoy.

Referencias Bibliográficas

Amazon Web Services. *¿Qué es la minería de datos?* [En línea]. Disponible en: <https://aws.amazon.com/es/what-is/data-mining/> (consultado abril 2025).

Amazon Web Services. *¿Qué es Hadoop?* [En línea]. Disponible en: <https://aws.amazon.com/es/what-is/hadoop/> (consultado abril 2025).

BBVA (2024, 15 abril). *Cinco ejemplos de “big data” que inspiran a otras empresas.* BBVA Innovación.

DataCamp. *Procesamiento por lotes frente a procesamiento por flujos: ¿Cuándo utilizar cada uno y por qué es importante?* [Blog].

IBM (2024, 9 agosto). *¿Qué es la inteligencia artificial (IA)?* IBM

THINK. IBM (2024, 18 junio). *¿Qué es la inferencia de la IA?* IBM

THINK.

IBM (s.f.). *Ejemplos, aplicaciones y casos de uso de machine learning.* IBM

THINK. IBM (2024). *¿Qué es Apache Spark?* IBM (página web).

InteractiveChaos. *Aprendizaje batch y online.* [Tutorial de Machine Learning].

Knight, W. (2023, 27 junio). *Los nuevos robots de Amazon revolucionan la automatización.* Wired (edición España).

Nucleoo. *3 Vs de Big Data: Variedad, Volumen, Velocidad* [Blog].

Perez, O. (2025, 7 mayo). *¿Qué es DALL·E y por qué es tan relevante en IA?* Xataka.

Tableau. *Ocho ejemplos cotidianos del procesamiento de lenguaje natural (PLN) y machine learning.* Tableau Blog.

Unite.AI (2025). *Las 10 mejores bibliotecas de Python para el aprendizaje profundo.*