

2. Identificación de las ramas de la IA y sus algoritmos.

La **inteligencia artificial** no forma un bloque único y homogéneo, sino un conjunto amplio de enfoques, técnicas y aplicaciones que conviene ordenar para entenderlos con claridad. Bajo esta denominación se agrupan sistemas orientados al análisis de datos, al tratamiento del lenguaje, al reconocimiento de imágenes, a la automatización física o a la generación de contenido, entre otros. Por eso, antes de profundizar en modelos concretos, resulta necesario disponer de una visión estructurada que permita distinguir las principales ramas del campo y relacionarlas con los algoritmos que suelen emplearse en cada una.

Esta organización ayuda a comprender por qué unas soluciones se apoyan en **aprendizaje automático**, otras en **redes neuronales**, otras en **reglas y representación del conocimiento** y otras en técnicas especializadas para texto, imagen, voz o recomendación. Tener este mapa general permite situar cada técnica dentro de su contexto y valorar con más criterio qué tipo de herramienta puede resultar más adecuada según la naturaleza del problema que una empresa quiera resolver.

2.1. Principales ramas de la inteligencia artificial.

Cuando se intenta clasificar la **inteligencia artificial**, aparece enseguida una dificultad: **no existe una división única, cerrada e idéntica en todos los contextos**. Esto ocurre porque la IA puede organizarse de dos maneras distintas. A veces se clasifica según **cómo funciona el sistema**, es decir, según el enfoque técnico que utiliza para aprender o razonar. Otras veces se clasifica según **qué tipo de problema resuelve**, por ejemplo, trabajar con texto, con imágenes, con voz o con robots. En la práctica, ambas formas de ordenar el campo conviven.

Para entender mejor este panorama, conviene distinguir entre ramas más **técnicas** y ramas más **aplicadas**. Las primeras describen la forma de construir el sistema. Las segundas se refieren al tipo de tarea o de dato sobre el que trabaja. Esta diferencia es importante porque evita una confusión muy frecuente: pensar que todas las ramas están en el mismo plano, cuando en realidad unas son métodos base y otras son campos de aplicación.

Siguiendo este criterio, las principales ramas de la inteligencia artificial pueden organizarse así:

Rama o familia principal	Qué la caracteriza	Ejemplo orientativo de uso
Enfoques basados en conocimiento	Representan información, reglas y relaciones para razonar sobre un problema.	Un sistema que aplica reglas para apoyar decisiones en un dominio muy definido.
Machine Learning	Entrena modelos con datos para hacer predicciones, clasificaciones o generar salidas útiles.	Un modelo que anticipa demanda o detecta riesgo de abandono de clientes.
Deep Learning	Utiliza redes neuronales profundas para captar patrones complejos, sobre todo en datos poco estructurados.	Un sistema que reconoce defectos en imágenes o analiza texto y voz a gran escala.
Procesamiento del lenguaje natural	Permite trabajar con lenguaje humano escrito o hablado.	Clasificar correos, resumir informes o responder consultas por chat.

Visión por computador	Analiza imágenes y vídeo para detectar objetos, patrones o incidencias.	Control de calidad visual o conteo automático de piezas.
Reconocimiento y tratamiento del habla	Trabaja con voz, audio y transcripción.	Dictado, asistentes de voz o análisis de llamadas.
Sistemas inteligentes de apoyo a la decisión	Integran información y modelos para recomendar o priorizar acciones.	Priorización de incidencias o apoyo a decisiones operativas.
Robótica inteligente	Combina percepción, decisión y actuación en entornos físicos.	Robots móviles de almacén o sistemas automáticos de manipulación.

Esta tabla permite fijar una idea esencial: **las ramas de la IA no funcionan como compartimentos aislados**. En muchos sistemas reales aparecen combinadas. Un asistente virtual puede utilizar **procesamiento del lenguaje natural** para interpretar preguntas, **Machine Learning** para clasificar intenciones y, en algunos casos, **Deep Learning** para mejorar el tratamiento del texto o de la voz. De forma parecida, un robot avanzado puede integrar visión por computador, modelos de aprendizaje y sistemas de decisión. Por tanto, hablar de ramas sirve para ordenar el conocimiento, pero no debe llevar a pensar que cada solución empresarial pertenece a una única categoría pura.



También conviene recordar que algunas ramas tienen un carácter más general y otras un carácter más funcional. **Machine Learning** y **Deep Learning** describen, sobre todo, maneras de construir modelos. En cambio, el **procesamiento del lenguaje natural** o la **visión por computador** describen campos de aplicación sobre determinados tipos de datos. Esta diferencia ayuda mucho a interpretar la IA con orden, porque aclara que no siempre hablamos de cosas equivalentes.

Desde una perspectiva empresarial, esta clasificación resulta muy útil. Cuando una empresa quiere trabajar

con datos tabulares para predecir ventas, rotación de clientes o riesgo de impago, suele apoyarse en **Machine Learning**. Cuando necesita analizar imágenes, audio o lenguaje con una complejidad mayor, suele entrar en juego el **Deep Learning**. Cuando el objetivo consiste en procesar contratos, reclamaciones, correos o conversaciones, el campo de referencia suele ser el **procesamiento del lenguaje natural**. La rama elegida depende, por tanto, del problema, del tipo de dato disponible y del nivel de complejidad que exige la tarea.

Ejemplo

Una cadena de tiendas quiere mejorar su funcionamiento en tres frentes. En primer lugar, desea prever mejor las ventas semanales de cada establecimiento; aquí encaja un enfoque de Machine Learning. En segundo lugar, quiere detectar automáticamente productos mal colocados en una estantería mediante cámaras; en este caso entra en juego la visión por computador, normalmente apoyada en Deep Learning. En tercer lugar, busca clasificar y resumir reclamaciones escritas por la clientela; aquí el campo central es el procesamiento del lenguaje natural. El ejemplo muestra que la inteligencia artificial no llega a la empresa como un bloque uniforme, sino como un conjunto de ramas que responden a necesidades distintas.

Sabías que...

Muchas herramientas que hoy parecen muy distintas comparten técnicas de base. Por ejemplo, una aplicación que resume textos y otra que analiza imágenes pueden apoyarse ambas en redes neuronales profundas, aunque trabajen con datos completamente diferentes.

Actividad 11

Relaciona cada rama con el ejemplo que mejor encaja.

- a. Procesamiento del lenguaje natural.
 - b. Visión por computador.
 - c. Robótica inteligente.
 - d. Machine Learning.
-
1. Predecir qué clientes tienen mayor riesgo de darse de baja.
 2. Analizar fotografías para detectar defectos en piezas.
 3. Interpretar y resumir correos electrónicos.
 4. Desplazar un robot por un almacén y decidir por dónde moverse.

2.2. Machine Learning como rama aplicada de la IA.

Dentro de las ramas de la inteligencia artificial, el **Machine Learning** ocupa una posición especialmente importante porque ha sido una de las vías que más ha impulsado la aplicación práctica de la IA en las últimas décadas. En términos básicos, puede entenderse como el proceso de **entrenar un modelo** para que haga **predicciones útiles** o genere contenido a partir de datos. Esta idea es muy importante, porque muestra que el valor del Machine Learning no está en ejecutar reglas fijas, sino en **aprender patrones** a partir de ejemplos previos.

Esta definición permite entender por qué el Machine Learning puede considerarse una **rama aplicada** de la IA. La inteligencia artificial, en sentido amplio, plantea el objetivo de construir sistemas capaces de inferir resultados útiles. El Machine Learning traduce ese objetivo a una metodología concreta: en vez de programar manualmente todas las reglas de decisión, se **entrena un modelo con datos** para que aprenda relaciones, regularidades o comportamientos estadísticos. De este modo, la IA deja de ser solo una idea general sobre sistemas inteligentes y pasa a convertirse en una práctica operativa basada en **entrenamiento, validación e inferencia**.



EDITORIAL TUTOR FORMACIÓN

El elemento central aquí es el **modelo**. Un modelo es la estructura y el conjunto de parámetros que el sistema utiliza para producir una predicción, una clasificación o una salida a partir de datos de entrada. El **entrenamiento** consiste precisamente en ajustar ese modelo para que funcione bien con nuevos datos del mismo tipo que los utilizados durante el aprendizaje. Esta lógica explica por qué el Machine Learning se ha vuelto tan útil en la empresa: permite automatizar tareas donde no resulta viable escribir una regla exacta para cada situación posible.

Para entender mejor su funcionamiento, conviene resumir sus componentes básicos de forma ordenada:

Elemento básico	Función dentro del Machine Learning.
Datos	Proporcionan los ejemplos a partir de los cuales el sistema aprende.
Modelo	Es la estructura matemática o computacional que intenta captar relaciones útiles en esos datos.
Entrenamiento	Ajusta el modelo para mejorar su capacidad de predecir o generar salidas útiles.
Predicción o inferencia	Es el uso del modelo ya entrenado sobre datos nuevos.
Evaluación	Permite comprobar si el modelo funciona con precisión y fiabilidad suficientes para el caso de uso.

Esta estructura parece sencilla, pero representa un cambio profundo respecto al software tradicional. En un programa clásico, la persona programadora define de manera explícita las reglas de funcionamiento. En Machine Learning, muchas de esas relaciones no se escriben una por una, sino que **se aprenden a partir de ejemplos**. Por eso este enfoque resulta tan útil en problemas donde hay suficientes datos, pero no sería razonable redactar manualmente una regla para cada combinación posible de variables.

Además, el Machine Learning no es una técnica única, sino una **familia de enfoques**. Entre las categorías más conocidas se encuentran el **aprendizaje supervisado**, el **aprendizaje no supervisado**, el **aprendizaje por refuerzo** y, en el panorama actual, también los sistemas generativos entrenados a gran escala. Todos comparten una idea básica: **aprender a partir de datos**, aunque lo hagan de maneras diferentes y con objetivos distintos.

Desde el punto de vista empresarial, el Machine Learning resulta especialmente valioso en tareas como **predecir demanda**, **clasificar correos o incidencias**, **detectar fraude**, **recomendar productos**, **estimar riesgos** o **segmentar comportamientos**. En todos estos casos aparece un rasgo común: existen suficientes datos como para entrenar un modelo, pero no resulta razonable programar una regla exacta para cada caso posible. El valor del ML está precisamente en convertir historiales, registros o patrones de uso en una base útil para predecir o decidir mejor.

Ejemplo

Una empresa de logística quiere anticipar retrasos en las entregas. Si intentara resolverlo con reglas fijas, tendría que contemplar innumerables combinaciones de tráfico, meteorología, carga, rutas, incidencias y horas punta. En cambio, con un enfoque de **Machine Learning** puede entrenar un modelo con datos históricos de entregas, tiempos, ubicaciones, condiciones externas y resultados anteriores. A partir de ese aprendizaje, el sistema puede estimar la probabilidad de retraso en nuevas entregas. Aquí no se ha programado manualmente cada posibilidad; se ha entrenado un modelo para inferir una salida útil a partir de los datos.

Sabías que...

En Machine Learning, entrenar un modelo no significa que ya funcione bien por sí solo. Después del entrenamiento hay que **evaluarlo** con métricas adecuadas para comprobar si realmente predice con la calidad necesaria para el caso de uso.

Actividad 12

Completa las frases con las palabras correctas: **modelo, datos, entrenamiento, inferencia.**

- a. El _____ es la estructura que utiliza el sistema para hacer predicciones o generar salidas.
- b. Los _____ aportan los ejemplos a partir de los cuales aprende el sistema.
- c. El _____ ajusta el modelo para mejorar su rendimiento.
- d. La _____ consiste en utilizar el modelo ya preparado sobre casos nuevos.

2.3. Deep Learning y redes neuronales.

Dentro del **Machine Learning**, el **Deep Learning** representa una especialización de gran relevancia. De forma sencilla, puede decirse que trabaja con **redes neuronales profundas**, es decir, con modelos que tienen varias capas intermedias y que son capaces de captar relaciones complejas en los datos. Esta profundidad adicional es lo que permite abordar tareas que resultan difíciles de resolver con reglas simples o con modelos más básicos.

Para comprender bien el **Deep Learning**, primero hay que entender qué es una **red neuronal**. Una red neuronal es un modelo formado por nodos o **neuronas** organizados en capas. Los datos entran por una **capa de entrada**, pasan por una o varias **capas ocultas** y, finalmente, llegan a una **capa de salida**, que produce el resultado. En ese recorrido, cada neurona combina valores, aplica una función y transmite información a la siguiente capa. Esta estructura permite modelar relaciones no lineales, es decir, relaciones complejas que no siguen una correspondencia simple y directa entre entrada y salida.

La idea básica puede resumirse así. Los datos entran en la red y van atravesando distintas capas. En cada una de ellas, el sistema transforma la información y va extrayendo patrones cada vez más elaborados. Al final, la red produce una salida, que puede ser una **clasificación**, una **probabilidad**, una **predicción** o cualquier otro resultado útil para la tarea planteada. Durante el entrenamiento, la red ajusta sus parámetros internos para reducir errores. Ese ajuste suele realizarse mediante **retropropagación**, un procedimiento muy utilizado para modificar los pesos de la red y mejorar su rendimiento.

Para fijar mejor esta arquitectura, conviene ordenar sus componentes principales:

Componente	Función principal
Capa de entrada	Recibe los datos iniciales con los que trabaja el modelo.
Capas ocultas	Transforman la información y permiten descubrir relaciones complejas entre variables.
Neuronas	Calculan combinaciones de entradas y aplican una función de activación.
Capa de salida	Produce la predicción, clasificación o resultado final.
Retropropagación	Ajusta los parámetros internos para reducir errores durante el entrenamiento.

Esta estructura explica por qué el **Deep Learning** suele asociarse a problemas de mayor complejidad. Las redes neuronales profundas están diseñadas para identificar patrones complejos y aprender representaciones útiles sin que una persona tenga que definir manualmente cada rasgo importante. Esta capacidad resulta especialmente valiosa cuando se trabaja con **imágenes, audio, texto, vídeo** o señales difíciles de describir mediante reglas cerradas.

La diferencia entre **red neuronal** y **Deep Learning** puede parecer pequeña, pero conviene dejarla clara. Toda red neuronal pertenece a esta familia de modelos. Sin embargo, cuando la arquitectura incorpora varias capas ocultas y gana profundidad, se habla de forma más específica de **Deep Learning**. Esa profundidad permite captar patrones más abstractos y más complejos, aunque también suele exigir **más datos, más capacidad de cálculo** y un entrenamiento más costoso.



Desde la perspectiva empresarial, el **Deep Learning** destaca cuando la tarea exige procesar información compleja y difícil de modelar con reglas simples o con enfoques menos avanzados. Esto ocurre, por ejemplo, en el reconocimiento de defectos en imágenes, en la transcripción y análisis de voz, en la clasificación de texto a gran escala, en la traducción automática o en la detección de patrones sutiles dentro de grandes volúmenes de datos. En estos contextos, la profundidad de la red ayuda al sistema a aprender representaciones más ricas y útiles.

Ejemplo

Una empresa industrial quiere detectar microdefectos en piezas fabricadas. Con una revisión visual manual, el proceso puede ser lento y variar según la persona que inspecciona. Con reglas fijas, resulta difícil capturar todas las formas posibles de defecto. En cambio, una red neuronal profunda entrenada con miles de imágenes etiquetadas puede aprender patrones visuales muy complejos y clasificar nuevas imágenes con mayor consistencia. En este caso, el valor del Deep Learning no está en “pensar” como una persona, sino en reconocer estructuras complejas dentro de datos visuales.

Sabías que...

Muchas aplicaciones actuales que trabajan con texto, imagen o voz utilizan redes profundas, aunque la persona usuaria no lo vea directamente. Detrás de una herramienta de transcripción, de un sistema de visión artificial o de un clasificador de documentos suele haber modelos que aprenden representaciones internas a través de varias capas.

Actividad 13

Relaciona cada elemento con su función.

- a. Capa de entrada.
 - b. Capas ocultas.
 - c. Capa de salida.
 - d. Retropropagación.
-
1. Ajusta parámetros para reducir errores durante el entrenamiento.
 2. Recibe los datos iniciales del modelo.
 3. Produce la predicción o resultado final.
 4. Transforman la información y captan relaciones complejas.

2.4. Procesamiento del lenguaje natural.

El **Procesamiento del Lenguaje Natural**, conocido habitualmente como **NLP** por sus siglas en inglés, es la rama de la inteligencia artificial orientada a trabajar con **lenguaje humano**, ya sea escrito o hablado. Se trata de uno de los campos más importantes de la IA actual, sobre todo porque una parte enorme de la actividad empresarial se expresa precisamente en forma de lenguaje: correos, contratos, informes, incidencias, reclamaciones, mensajes de soporte, documentación técnica o conversaciones con la clientela.

Entender esta rama exige partir de una idea sencilla: gran parte de la información útil de una empresa no está en tablas numéricas, sino en **textos y comunicaciones**. Trabajar manualmente con ese volumen de lenguaje resulta lento y costoso. Precisamente ahí entra el **NLP**, cuyo objetivo general es permitir que los sistemas **analicen, clasifiquen, interpreten, resuman o generen lenguaje** con utilidad práctica. En otras palabras, intenta convertir el lenguaje en algo que un sistema pueda tratar de forma operativa.

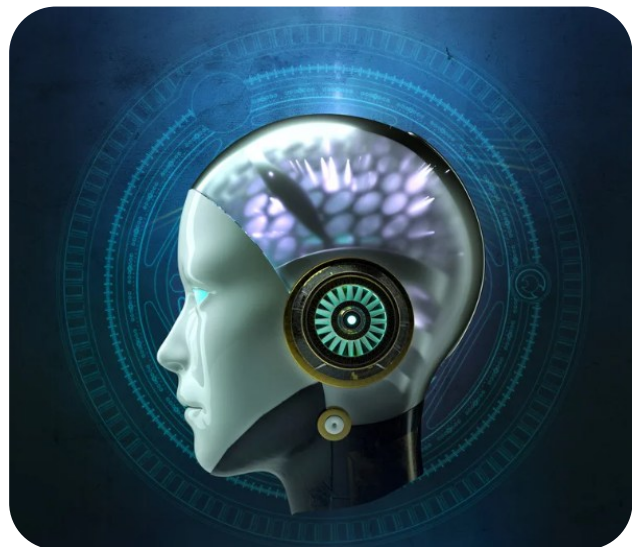
Dentro de este campo, los **modelos de lenguaje** ocupan un lugar muy importante. De forma sencilla, un modelo de lenguaje estima la probabilidad de que aparezca un **token** o una secuencia de tokens dentro de una secuencia más larga. Un token puede ser una palabra, una parte de una palabra o incluso un carácter, según el sistema utilizado. Esta idea resulta útil porque explica la base con la que muchos modelos actuales procesan texto: trabajan con unidades del lenguaje y calculan relaciones probabilísticas y contextuales entre ellas.

Para ordenar mejor lo que hace esta rama, conviene resumir algunas de sus tareas más habituales:

Tarea de NLP	Qué permite hacer
Clasificación de texto	Asignar categorías a documentos, mensajes o incidencias.
Traducción automática	Convertir texto de una lengua a otra.
Resumen automático	Reducir un documento a sus ideas principales.
Extracción de información	Localizar datos, entidades o relaciones relevantes dentro de un texto.
Respuesta a preguntas.	Generar o seleccionar respuestas a partir de una consulta textual.
Generación de texto	Producir texto nuevo coherente con una instrucción o un contexto.

Este conjunto de tareas muestra que el **Procesamiento del Lenguaje Natural** no se limita a “leer palabras” de forma superficial. En realidad, intenta representar el lenguaje de una manera que permita clasificarlo, resumirlo, completarlo, traducirlo o relacionarlo con una intención concreta. Para ello, muchos sistemas trabajan con **tokenización**, es decir, con la división del texto en unidades más pequeñas que después se transforman en representaciones numéricas que el modelo puede procesar.

También conviene señalar que el **NLP** ha cambiado mucho con el tiempo. Durante años se utilizaron enfoques más simples, como modelos basados en frecuencias y secuencias cortas de palabras. Más adelante, fueron ganando peso las redes neuronales y, después, los modelos preentrenados de lenguaje, que



transformaron profundamente el campo. Gracias a ese avance, hoy existen sistemas mucho más capaces para resumir, traducir, responder preguntas o analizar documentos extensos.

Desde una perspectiva empresarial, el valor del **NLP** es muy amplio. Puede utilizarse para clasificar automáticamente correos y consultas, detectar temas recurrentes en reclamaciones, resumir informes extensos, extraer datos clave de contratos, asistir al personal de soporte con respuestas sugeridas o analizar grandes volúmenes de documentación técnica. En todos estos casos, el sistema no sustituye por completo el juicio humano, pero sí puede reducir carga operativa, ordenar mejor la información y acelerar tareas que dependen del lenguaje escrito o hablado.

Ejemplo

Una empresa recibe miles de mensajes al mes en su servicio de atención. Sin ayuda tecnológica, el personal debe leer uno por uno esos textos, clasificarlos y derivarlos al área correspondiente. Con un sistema de NLP, puede automatizarse una parte importante del proceso: identificar el tema principal del mensaje, detectar urgencia, resumir el contenido y sugerir a qué departamento debe enviarse. Aquí la utilidad no reside en una conversación espectacular, sino en tratar lenguaje humano de forma suficientemente estructurada como para apoyar el trabajo diario.

Sabías que...

Muchos sistemas actuales de lenguaje no trabajan directamente con frases completas tal y como las leemos las personas. Primero las dividen en tokens y después convierten esos tokens en representaciones numéricas para poder analizarlos y generar respuestas.

Actividad 14

Completa las frases con las palabras correctas: tokens, resumen, clasificación, lenguaje.

- El NLP es la rama de la IA que trabaja con el _____ humano.
- Un modelo de lenguaje suele procesar unidades llamadas _____.
- Reducir un documento a sus ideas principales es una tarea de _____ automático.
- Asignar una categoría a un correo o a una incidencia es una tarea de _____ de texto.

2.5. Visión por computador.



La **visión por computador** es la rama de la inteligencia artificial que trabaja con **información visual** para que un sistema pueda procesarla y extraer de ella contenido útil. Dicho de una forma sencilla, su objetivo es que una máquina pueda **analizar imágenes o vídeo** y convertir esa información en una salida operativa, como una clasificación, una detección, una localización o una alerta.

Esta definición es importante porque permite entender bien la diferencia entre **tener una imagen**

y **aprovechar lo que aparece en ella**. Una imagen contiene muchísima información visual, pero esa información no sirve de nada por sí sola si el sistema no puede identificar objetos, regiones,

patrones o relaciones espaciales. La visión por computador intenta precisamente transformar esa señal visual en una salida útil para un proceso.

Dentro del mapa general de la inteligencia artificial, la visión por computador suele aparecer muy vinculada al **aprendizaje automático** y, en muchas aplicaciones actuales, al **Deep Learning**. Esto no significa que toda solución de visión por computador dependa obligatoriamente de redes profundas, pero sí explica por qué una parte importante de sus avances recientes se ha apoyado en modelos capaces de aprender patrones visuales complejos a partir de grandes volúmenes de datos. La lógica de esta rama puede entenderse de forma bastante clara. Una persona mira una imagen y puede reconocer si hay una pieza defectuosa, una caja, una estantería desordenada o un producto fuera de lugar. La visión por computador intenta lograr que un sistema haga algo parecido desde un punto de vista operativo: detectar qué aparece en la escena, dónde aparece y, en algunos casos, cómo evoluciona a lo largo del tiempo.

Para ordenar mejor sus tareas principales, conviene tener en cuenta que la visión por computador no se reduce a una sola operación. Entre las tareas más habituales se encuentran las siguientes:

Tarea visual	Qué permite obtener
Clasificación de imagen	Determinar qué contenido principal aparece en una imagen.
Detección de objetos	Localizar uno o varios objetos dentro de una escena.
Seguimiento de objetos	Mantener identificados objetos a lo largo de una secuencia de vídeo o de una cámara en tiempo real.
Segmentación semántica	Asignar categorías a regiones o grupos de píxeles de la imagen.
Segmentación por instancias	Separar objetos concretos, incluso cuando pertenecen a la misma clase.

Esta clasificación resulta útil porque ayuda a ver que la complejidad puede aumentar bastante según la tarea. **Clasificar** una imagen suele consistir en responder qué aparece en ella. **Detectar** objetos obliga, además, a indicar dónde aparecen. **Segmentar** exige ir todavía más lejos, ya que implica trabajar con regiones o incluso con píxeles concretos. Por tanto, no todas las aplicaciones visuales exigen el mismo nivel de dificultad ni la misma cantidad de datos.

En la empresa, esta rama adquiere valor cuando una tarea depende de observar y reconocer visualmente algo que no resulta práctico revisar siempre de forma manual. A partir de las capacidades anteriores, la visión por computador puede aplicarse en contextos como la **inspección visual de calidad**, el **recuento de objetos**, la **detección de defectos**, el **análisis de flujo en espacios físicos** o la **verificación automática de presencia o ausencia de elementos** en una imagen.

Estas aplicaciones no aparecen porque la máquina vea como una persona. Su utilidad está en que puede **convertir patrones visuales en resultados utilizables dentro de un proceso**. Esa traducción desde la tarea visual al proceso de negocio es, precisamente, lo que convierte a la visión por computador en una herramienta operativa dentro de la empresa.

También conviene señalar que la visión por computador se ha desarrollado con mucha fuerza gracias al **Deep Learning**. Esto explica por qué, en muchos manuales actuales, esta rama aparece muy unida a redes neuronales profundas, a modelos entrenados con grandes conjuntos de imágenes y, más recientemente, a arquitecturas más avanzadas para tareas visuales. Aun así, desde el punto de vista



conceptual, sigue siendo útil tratarla como una rama propia, porque el tipo de problema que aborda está definido por el **dato visual** y no únicamente por la técnica concreta que se utilice para procesarlo.

En consecuencia, la visión por computador debe entenderse como la rama de la IA que permite **analizar imágenes y vídeo para obtener resultados operativos**. Su importancia en la empresa crece allí donde una parte del trabajo depende de revisar visualmente objetos, escenas o movimientos, y donde esa revisión puede beneficiarse de mayor velocidad, continuidad o consistencia.

Ejemplo

En una planta industrial, una revisión manual puede detectar piezas mal terminadas, pero hacerlo una por una durante toda la producción exige tiempo, concentración continua y puede variar según la persona que inspecciona. Si se entrena un sistema visual para reconocer determinadas irregularidades en imágenes de las piezas, el proceso puede ganar continuidad y consistencia. En este caso, la IA no sustituye toda la supervisión humana, pero sí aporta una capacidad de cribado visual que ayuda a detectar incidencias con mayor rapidez.

La misma lógica puede trasladarse a otros contextos empresariales. Una tienda puede utilizar cámaras para comprobar si faltan productos en determinadas baldas. Un almacén puede contar paquetes automáticamente. Un sistema logístico puede verificar si una etiqueta está colocada correctamente. En todos estos casos, la base es la misma: extraer información útil a partir de la imagen.

Sabías que...

Dos tareas visuales que a veces se confunden son la detección y la segmentación. Detectar suele indicar dónde está un objeto dentro de una imagen. Segmentar va más allá, porque delimita con mucho más detalle la zona que ocupa.

Actividad 15

Relaciona cada tarea de visión por computador con su función.

- a. Clasificación de imagen.
 - b. Detección de objetos.
 - c. Seguimiento de objetos.
 - d. Segmentación semántica.
-
1. Mantener identificado un elemento a lo largo de una secuencia de vídeo.
 2. Indicar qué contenido principal aparece en una imagen.
 3. Asignar categorías a regiones o grupos de píxeles.
 4. Localizar uno o varios elementos dentro de la escena.

2.6. Sistemas expertos y representación del conocimiento.

Frente a ramas como la visión por computador, muy vinculadas al aprendizaje desde datos, los **sistemas expertos** y la **representación del conocimiento** remiten a una tradición distinta dentro de la inteligencia artificial. Se trata de una línea más próxima al **razonamiento simbólico**, a la explicitación del conocimiento y a la idea de que una máquina puede comportarse de forma útil si dispone de una estructura adecuada para representar **hechos, reglas y relaciones**.

Para entender bien este apartado, conviene separar dos ideas que están relacionadas, pero que no significan exactamente lo mismo. Por un lado, están los **sistemas expertos**, que son sistemas diseñados para trabajar sobre un dominio especializado y ofrecer conclusiones, recomendaciones o diagnósticos apoyándose en conocimiento formalizado. Por otro lado, está la **representación del conocimiento**, que se refiere a la manera en que ese conocimiento se organiza y se expresa para que una máquina pueda utilizarlo.

La relación entre ambos conceptos es muy estrecha. Un sistema experto necesita alguna forma de representación del conocimiento para poder razonar sobre el dominio con el que trabaja. Dicho de otra manera, si el conocimiento no está estructurado de una forma que el sistema pueda manejar, no resulta posible construir un razonamiento operativo.

Los primeros sistemas expertos trabajaban sobre todo con **conjuntos de reglas**. Más adelante, su desarrollo fue incorporando una mejor separación entre la **base de conocimiento** y el **mecanismo de razonamiento**. Esta evolución fue importante porque facilitó el mantenimiento del sistema, la actualización del conocimiento y también la posibilidad de explicar cómo se había alcanzado una conclusión.

A partir de esta idea, puede decirse que un sistema experto suele apoyarse en varias piezas reconocibles. Necesita conocimiento del dominio, necesita una forma de representarlo y necesita un mecanismo capaz de operar con él. Expresado de manera ordenada, suele funcionar sobre esta estructura general:

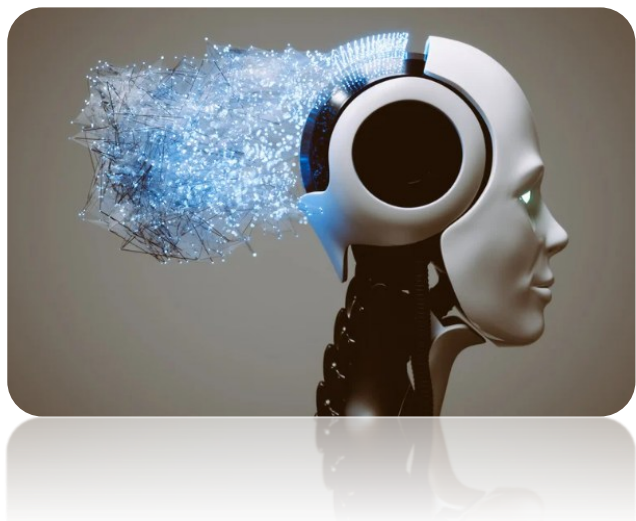
Componente	Función dentro del sistema experto
Conocimiento del dominio	Reúne hechos, reglas, relaciones o criterios especializados sobre un área concreta.
Representación del conocimiento	Organiza ese contenido de forma utilizable por la máquina.
Mecanismo de inferencia	Aplica reglas o procedimientos para llegar a conclusiones, recomendaciones o diagnósticos.
Capacidad de explicación	Permite, en muchos casos, justificar cómo se ha alcanzado una conclusión.

Este esquema ayuda a entender por qué los sistemas expertos tuvieron tanta relevancia histórica. Permitían trasladar parte de la experiencia o del criterio de personas especializadas a un sistema que podía aplicarlo de forma consistente dentro de un dominio bien delimitado. En otras palabras, no pretendían reproducir una inteligencia general, sino ofrecer competencia en problemas concretos.

Por eso han resultado especialmente útiles en ámbitos como los siguientes:

- **Diagnóstico de incidencias.**
- **Clasificación de problemas técnicos.**
- **SopORTE a la toma de decisiones.**
- **Aplicación de reglas de negocio.**
- **Comprobaciones normativas o de cumplimiento.**

La fuerza de este enfoque reside en que el conocimiento se expresa de forma **explícita**. Esto marca una diferencia importante respecto a modelos que aprenden patrones a partir de grandes cantidades de datos sin que las reglas aparezcan escritas de manera directa. En los sistemas expertos, la lógica del dominio puede verse, revisarse y, en muchos casos, explicarse mejor.



Ahora bien, la **representación del conocimiento** no debe entenderse como un simple almacén de información. Su importancia va mucho más allá. La forma en que el conocimiento se representa condiciona qué puede expresar el sistema, cómo razona y qué tipo de inferencias puede realizar. Por eso esta rama ha sido tan importante dentro de la IA: no se limita a guardar conocimiento, sino que determina cómo puede usarse.

A lo largo del desarrollo de esta tradición se han utilizado distintas formas de representación. Entre ellas pueden encontrarse las **reglas**, las **redes semánticas**, las **ontologías**, las **taxonomías**, las **lógicas formales** y otras estructuras destinadas a organizar conceptos, propiedades y relaciones. Esto demuestra que la representación del conocimiento no es una técnica única, sino una familia de maneras de estructurar el mundo para razonar sobre él.

Desde una perspectiva empresarial, esta rama resulta muy útil cuando el problema depende mucho de **criterios explícitos**, **reglas interpretables** y **trazabilidad de la decisión**. Puede pensarse, por ejemplo, en evaluaciones normativas, comprobaciones de cumplimiento, diagnóstico de incidencias siguiendo árboles de decisión formalizados, sistemas de apoyo técnico o herramientas que deben justificar por qué recomiendan una acción concreta.

En este tipo de situaciones, un enfoque puramente basado en datos puede no ser suficiente o no ser el más adecuado, sobre todo si la organización necesita entender el razonamiento del sistema y poder revisarlo. Aquí es donde los sistemas expertos conservan una utilidad clara: permiten trabajar con conocimiento que debe mantenerse visible, ordenado y justificable.

También es importante destacar que los sistemas expertos no han desaparecido por el hecho de que hoy existan modelos más orientados a datos. En realidad, siguen siendo útiles allí donde la organización necesita **explicabilidad**, **control**, **actualización directa de reglas** y **trazabilidad**. Por eso continúan siendo una referencia importante para entender una parte fundamental de la historia y de la práctica de la inteligencia artificial.

En consecuencia, los sistemas expertos y la representación del conocimiento ocupan una posición muy relevante dentro de la IA. Representan una vía en la que la inteligencia artificial se construye a partir de **conocimiento explícito**, **estructuras formales** e **inferencia razonada**. Aunque hoy conviven con enfoques mucho más orientados a datos, siguen siendo esenciales para comprender cómo una máquina puede razonar sobre un dominio especializado, explicar sus conclusiones y trabajar con conocimiento que no siempre resulta fácil reducir a simples correlaciones estadísticas.

Ejemplo

Supóngase una empresa que gestiona incidencias técnicas con un protocolo muy definido. Parte del conocimiento del personal senior puede formalizarse en reglas: si aparece un determinado síntoma junto con ciertas señales de contexto, conviene revisar primero un componente concreto; si se dan otras condiciones, debe seguirse otra secuencia de comprobación. Un sistema experto puede organizar ese conocimiento y ofrecer recomendaciones iniciales consistentes. En este caso, su valor no procede de aprender automáticamente a partir de millones de ejemplos, sino de representar bien el saber del dominio y utilizarlo de forma ordenada.

Sabías que...

Una de las grandes ventajas de los sistemas expertos es que, en muchos casos, permiten explicar por qué han llegado a una recomendación. Esa capacidad resulta muy valiosa en entornos donde la empresa necesita justificar decisiones.

Actividad 16

Completa las frases con las palabras correctas: inferencia, reglas, conocimiento, experto.

- Un sistema _____ trabaja sobre un dominio especializado.
- La representación del _____ organiza la información para que la máquina pueda utilizarla.
- El mecanismo de _____ aplica procedimientos para alcanzar conclusiones.
- Muchos sistemas expertos se apoyan en _____ para orientar sus recomendaciones.

2.7. Robótica inteligente y automatización avanzada.

La **robótica inteligente** puede entenderse como la aplicación de capacidades de **percepción, decisión y actuación** en sistemas físicos que operan en el entorno real. No se trata solo de una máquina que se mueve o repite un gesto mecánico. En realidad, es un sistema en el que se combinan **hardware, sensores, software, control y, en muchos casos, técnicas de inteligencia artificial** para actuar con cierto grado de autonomía. En la terminología utilizada por la IFR a partir de ISO 8373:2021, un robot es un **mecanismo actuado y programado, con cierto grado de autonomía**, capaz de realizar **locomoción, manipulación o posicionamiento**.

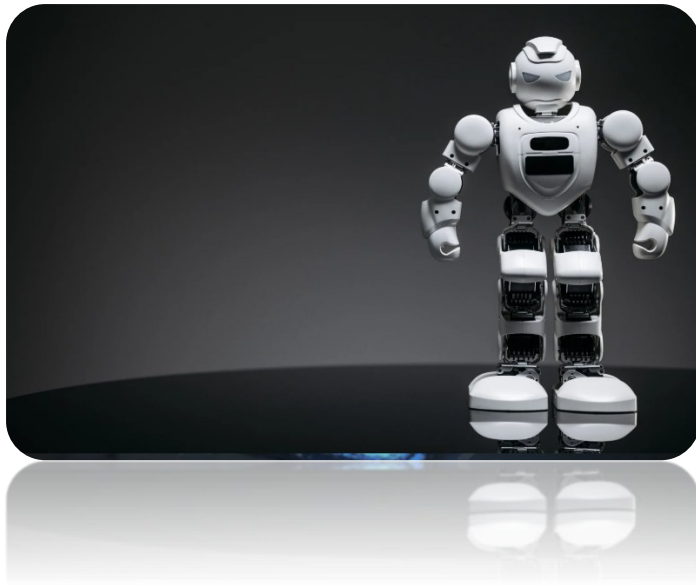
Esta definición ayuda a distinguir un robot de otras máquinas automatizadas. Una máquina puede repetir movimientos de forma automática y, aun así, no encajar plenamente en la idea de robot si no dispone de ese grado de autonomía ni de capacidad para operar con cierta adaptabilidad dentro de su entorno. Por eso, la robótica inteligente no debe verse como una simple prolongación de la automatización clásica, sino como una evolución hacia sistemas más flexibles, sensorizados e integrados.

A partir de esta base, la **automatización avanzada** puede interpretarse como el paso desde sistemas que ejecutan secuencias cerradas a sistemas capaces de **percibir condiciones cambiantes, ajustar su comportamiento y coordinar componentes físicos y digitales**. Dicho de una forma sencilla, cuanto más depende un sistema de sensores, de software de control, de datos del entorno y de capacidad de reacción, más se aleja del automatismo rígido y más se aproxima a la lógica de los sistemas autónomos o semiautónomos. Esta es la zona en la que la automatización avanzada y la robótica inteligente empiezan a encontrarse.

Para ver mejor la diferencia, conviene compararlas de forma ordenada:

Tipo de sistema	Rasgo dominante	Forma de funcionamiento	Ejemplo orientativo
Automatización tradicional	Repite una secuencia o regla fija.	Actúa según parámetros previamente definidos, con poca adaptación al contexto.	Una línea que ejecuta siempre el mismo movimiento cuando se activa una señal concreta.
Automatización avanzada	Integra sensorización, software y capacidad de ajuste.	Reacciona a variaciones del entorno o del proceso con mayor flexibilidad.	Un sistema que redistribuye tareas o modifica su actuación según datos operativos.
Robótica inteligente	Une percepción, decisión y acción física con cierto grado de autonomía.	Interpreta información del entorno y actúa sobre él mediante locomoción, manipulación o posicionamiento.	Un robot móvil o manipulador que colabora en operaciones de almacén o montaje.

Esta comparación permite fijar una idea esencial: **toda robótica participa de la automatización, pero no toda automatización avanzada adopta forma robótica**. Del mismo modo, no todos los robots son inteligentes en el mismo grado ni por los mismos mecanismos. Hay robots más rígidos y previsibles, y también hay sistemas de automatización sin cuerpo robótico que incorporan capacidades analíticas muy sofisticadas. Lo que interesa aquí es el punto en el que **la percepción del entorno, la decisión y la actuación física quedan integradas en un mismo sistema operativo**.



Dentro de este campo también conviene distinguir entre **robot industrial** y **robot de servicio**. La IFR, apoyándose igualmente en ISO 8373, considera robot industrial al manipulador multipropósito, automáticamente controlado y reprogramable, utilizado para tareas de automatización industrial. En cambio, los robots de servicio son aquellos que realizan tareas útiles para personas o equipos fuera de ese marco industrial clásico. Esta distinción es importante porque muestra que la robótica inteligente no se limita a la fábrica: también aparece en logística, mantenimiento, limpieza, salud, inspección,

agricultura o asistencia profesional.

Uno de los desarrollos más representativos de esta evolución es la **robótica colaborativa**. Los llamados **cobots** están pensados para compartir espacio de trabajo con personas y participar en tareas donde la interacción humano-robot forma parte del propio diseño del proceso. Esto ha cambiado bastante la imagen tradicional del robot aislado dentro de una célula cerrada. Además, la IFR informó de que los cobots alcanzaron una cuota del **10,5 %** de los robots industriales instalados en **2023**, lo que refleja su creciente importancia en entornos donde se busca flexibilidad y una entrada más accesible a la automatización avanzada.

Desde un punto de vista funcional, un sistema de robótica inteligente suele combinar varios componentes básicos:

Componente	Papel dentro de la robótica inteligente
Sensores	Captan información del entorno, del estado del robot o del proceso.
Actuadores y mecanismos	Permiten mover, posicionar o manipular elementos físicos.
Software de control	Coordina movimientos, secuencias y respuestas operativas.
Módulos de percepción o IA	Interpretan señales, reconocen patrones o apoyan la toma de decisiones.
Sistemas de comunicación e integración	Conectan el robot con otros equipos, plataformas o redes de producción y datos.

Esta estructura deja ver por qué la robótica inteligente se sitúa en un punto de encuentro entre varias disciplinas. No basta con construir una máquina robusta. Tampoco basta con añadir un modelo de IA. Lo decisivo es integrar **sensado, razonamiento operativo, control, comunicación y seguridad** en un sistema que pueda interactuar con un entorno físico cambiante. Por eso la normalización técnica tiene tanto peso en este campo.

Desde el punto de vista empresarial, la robótica inteligente aporta valor cuando el proceso combina **repetición, exigencia física, necesidad de precisión, continuidad operativa o trazabilidad**. Puede ayudar a estabilizar tareas, sostener ritmos de producción, mejorar la consistencia de ciertas operaciones o reforzar la seguridad en trabajos donde existe fatiga o riesgo físico. Ahora bien, reducirla a una “máquina que sustituye personas” sería una visión demasiado pobre. En muchos casos, su lógica real consiste en **redistribuir tareas**, dejar a la máquina la parte más repetitiva o físicamente exigente, y reservar a las personas funciones de supervisión, ajuste, mantenimiento o resolución de excepciones.

También conviene mantener una expectativa realista. La robótica inteligente exige **diseño del proceso, integración técnica, seguridad, mantenimiento y formación**. Su rendimiento no depende solo del robot o del algoritmo, sino de cómo se implanta dentro de un sistema de trabajo real. Por eso, cuanto más compleja sea la interacción entre personas, máquinas, datos y entorno físico, mayor importancia tendrá una implantación cuidadosa.

Ejemplo

Puede pensarse en un centro logístico donde una parte del trabajo consiste en mover contenedores, localizar ubicaciones y abastecer puestos de preparación de pedidos. Una automatización simple podría basarse en cintas o rutas fijas. En cambio, una solución de robótica inteligente puede incorporar plataformas móviles, sensores de entorno y conexión con el sistema de gestión del almacén para ajustar recorridos, detectar obstáculos y coordinar entregas internas. En este caso, la empresa no obtiene valor solo por “tener robots”, sino porque esos sistemas se integran en la lógica operativa y mejoran el flujo real del trabajo.

Sabías que...

Un robot puede incorporar tecnología robótica y, aun así, no cumplir la definición completa de robot si funciona solo por teleoperación y no tiene ese grado mínimo de autonomía que exigen las definiciones técnicas más utilizadas.

Actividad 17

Relaciona cada concepto con su definición correcta.

- Automatización tradicional.
 - Automatización avanzada.
 - Robótica inteligente.
 - Robot de servicio.
- Sistema que integra percepción, decisión y actuación física con cierto grado de autonomía.
 - Sistema que repite secuencias fijas con poca adaptación al entorno.
 - Robot que realiza tareas útiles para personas o equipos fuera del uso industrial clásico.
 - Sistema que combina sensores, software y capacidad de ajuste ante cambios del proceso.

2.8. IA generativa y modelos fundacionales.

La **IA generativa** es una de las manifestaciones más visibles de la inteligencia artificial actual, pero para entenderla bien conviene relacionarla con una noción más amplia: la de **modelo fundacional**. En términos sencillos, la IA generativa es aquella que **produce contenido nuevo** a partir de una entrada, una instrucción o un contexto. Ese contenido puede adoptar muchas formas: **texto, imagen, audio, vídeo, código o combinaciones de varios formatos**. NIST la describe precisamente como una clase de IA



EDITORIAL TUTOR FORMACIÓN

capaz de generar contenido a partir de datos, observaciones o descripciones proporcionadas en sus interfaces.

Los **modelos fundacionales**, por su parte, se definieron de forma muy influyente en Stanford como modelos **entrenados con datos amplios**, generalmente mediante autosupervisión a gran escala, que pueden **adaptarse a una amplia variedad de tareas posteriores**. Esta definición es muy importante porque explica el cambio de paradigma que se ha producido en la construcción de sistemas de IA: ya no se parte siempre de un modelo diseñado desde cero para una única tarea, sino de una base general que después puede ajustarse, especializarse o integrarse en productos y servicios muy distintos.

La relación entre ambas ideas es estrecha, pero no idéntica. Muchos sistemas de **IA generativa** actuales se apoyan en **modelos fundacionales**, aunque el concepto de modelo fundacional es más amplio que el de sistema generativo concreto. Además, en el marco regulatorio europeo actual aparece otra expresión muy relevante: **modelo de propósito general**. La Comisión Europea utiliza esta categoría para referirse a modelos con una **generalidad significativa** y capacidad para desempeñar una amplia gama de tareas distintas, y ha desarrollado preguntas y respuestas, directrices y un código de prácticas específicos para este tipo de modelos.

Para aclarar mejor esta relación, conviene compararlos de forma sencilla:

Concepto	Qué describe	Papel dentro del ecosistema actual
IA generativa	Sistemas capaces de producir contenido nuevo a partir de una entrada o instrucción.	Se materializa en aplicaciones que generan texto, imagen, audio, vídeo, código u otros contenidos.
Modelo fundacional	Modelo entrenado a gran escala sobre datos amplios y adaptable a muchas tareas posteriores.	Funciona como base técnica sobre la que se construyen aplicaciones, ajustes y servicios diversos.
Modelo de propósito general	Modelo con alta generalidad y capacidad para desempeñar una amplia gama de tareas distintas.	Es una categoría central en el marco europeo actual para los modelos generalistas.

Esta comparación ayuda a evitar una confusión frecuente. La **IA generativa** suele identificarse con la interfaz que redacta, resume, traduce o crea imágenes. El **modelo fundacional** remite, en cambio, a la infraestructura técnica subyacente que ha sido entrenada a gran escala y luego puede reutilizarse en muchas tareas. Y el **modelo de propósito general** introduce una categoría especialmente importante desde el punto de vista regulatorio y funcional. En la práctica, muchas herramientas visibles para la persona usuaria son la capa final de una base técnica mucho más amplia.

El cambio que introducen estos modelos es profundo. Un mismo modelo entrenado sobre datos extensos puede adaptarse para **resumir documentos, responder preguntas, traducir, clasificar texto, generar borradores, producir imágenes o asistir en tareas de programación**. En el plano empresarial, esto supone una ventaja clara de **reutilización, escalabilidad y velocidad de despliegue**. En lugar de desarrollar desde cero una solución distinta para cada tarea, muchas organizaciones pueden apoyarse en una base amplia y adaptable.

Ahora bien, esa misma amplitud introduce también riesgos y exigencias nuevas. La Comisión Europea ha desarrollado reglas específicas para los modelos de propósito general dentro del **AI Act**, y desde el **2 de agosto de 2025** las obligaciones correspondientes empezaron a aplicarse a los proveedores que los ponen en el mercado de la Unión Europea. Además, para los modelos con **riesgo sistémico** existen obligaciones adicionales ligadas a evaluación, mitigación de riesgos, documentación e interacción con la AI Office. Todo ello muestra que estos modelos ya no se consideran una cuestión marginal, sino una pieza estructural del ecosistema actual.

Desde el punto de vista funcional, la IA generativa trabaja con una lógica distinta de la de muchos sistemas clásicos de IA. En lugar de limitarse a clasificar o predecir, **genera una salida nueva** condicionada por una instrucción o por un contexto previo. Esa diferencia es importante porque introduce retos específicos. NIST ha desarrollado un perfil de gestión del riesgo para IA generativa que presta especial atención a cuestiones como **integridad de la información, procedencia del contenido, toxicidad, sesgo, alucinaciones, confidencialidad, privacidad y gobernanza**. Esto



indica que la IA generativa no debe verse solo como una ampliación de capacidades anteriores, sino como un cambio cualitativo con exigencias propias.

En la empresa, sus posibilidades son muy amplias. Puede ayudar a **redactar borradores, resumir documentos extensos, crear respuestas iniciales, producir variaciones de contenido comercial, asistir a personal técnico con documentación, apoyar tareas**

de programación o generar material visual preliminar. Sin embargo, presentar estas capacidades como una automatización completa del trabajo intelectual sería una simplificación excesiva. Su valor real depende mucho de cómo se integre dentro de un flujo de revisión, responsabilidad y uso razonable.

Los modelos fundacionales también introducen una cuestión estructural importante: la **dependencia de una misma base técnica para muchas aplicaciones posteriores.** Esta reutilización aporta rapidez y escalabilidad, pero también puede concentrar riesgos. Si muchas soluciones dependen del mismo modelo base, ciertos errores, sesgos, vulnerabilidades o decisiones del proveedor pueden propagarse ampliamente. Por eso han ganado tanta importancia la **transparencia, la documentación y la rendición de cuentas** en torno a estos modelos.

También conviene subrayar una idea importante: **IA generativa y modelo fundacional** no equivalen a inteligencia general. Aunque su impacto público haya disparado la impresión de que la IA puede hacerlo todo, el enfoque técnico y regulatorio actual se centra en otra cuestión mucho más concreta: **qué capacidades tienen estos sistemas, en qué contextos se usan, qué riesgos introducen y qué obligaciones deben asumir quienes los desarrollan o despliegan.** Esa mirada resulta mucho más útil para la empresa que las expectativas exageradas.

Para resumir mejor su doble cara, puede resultar útil esta comparación:

Dimensión	Sentido en IA generativa y modelos fundacionales
Potencial operativo	Permiten reutilizar una misma base técnica para muchas tareas y acelerar la producción de contenidos y la asistencia cognitiva.
Escalabilidad	Facilitan desplegar aplicaciones diversas sin entrenar desde cero un modelo nuevo para cada función.
Dependencia	Pueden concentrar gran parte del ecosistema en unas pocas infraestructuras de modelo.
Riesgos	Introducen problemas de integridad de la información, alucinaciones, sesgos, privacidad, copyright, seguridad y trazabilidad.
Exigencia organizativa	Requieren documentación, evaluación, supervisión humana y políticas de uso ajustadas al caso concreto.

En consecuencia, la **IA generativa** y los **modelos fundacionales** ocupan hoy un lugar central dentro del mapa de la inteligencia artificial. La primera representa la capacidad de producir contenido nuevo de forma operativa. Los segundos representan el paradigma técnico que permite reutilizar una base amplia de capacidades en numerosas tareas posteriores. Juntas, ambas nociones ayudan a entender una parte esencial de la IA actual: el paso desde modelos diseñados para funciones muy concretas hacia infraestructuras generales capaces de alimentar una gran variedad de aplicaciones empresariales, con posibilidades muy amplias y con exigencias de control igualmente elevadas.

Ejemplo

Puede imaginarse un departamento comercial que prepara propuestas para distintos tipos de clientes. Un sistema generativo puede crear borradores iniciales a partir de plantillas, datos del cliente y objetivos de la oferta. Eso puede reducir tiempo y acelerar la primera redacción. Aun así, la revisión humana sigue siendo necesaria para comprobar exactitud, tono, coherencia contractual y adaptación al contexto. En este caso, la IA generativa no sustituye automáticamente la función profesional, pero sí puede convertirse en una herramienta de producción asistida.

Sabías que...

Desde el 2 de agosto de 2025, los proveedores que ponen en el mercado europeo modelos de propósito general deben cumplir las obligaciones específicas del AI Act aplicables a esos modelos.

Actividad 18

Relaciona cada concepto con su descripción correcta.

- a. IA generativa.
 - b. Modelo fundacional.
 - c. Modelo de propósito general.
 - d. Riesgo sistémico.
-
1. Categoría regulatoria europea para modelos con gran generalidad y amplio rango de tareas.
 2. Sistema capaz de producir contenido nuevo a partir de una entrada.
 3. Modelo entrenado a gran escala sobre datos amplios y adaptable a muchas tareas posteriores.
 4. Situación en la que un modelo muy avanzado puede requerir obligaciones reforzadas de evaluación y mitigación.

2.9. Algoritmos de clasificación.

Los **algoritmos de clasificación** se utilizan cuando el objetivo del modelo consiste en **asignar una clase o categoría** a cada caso analizado. Dicho de una forma sencilla, el sistema no intenta responder **cuánto vale algo**, sino **a qué grupo pertenece**. Esa diferencia es importante porque permite distinguir la clasificación de otras tareas del aprendizaje automático. Aquí el resultado esperado no es una cantidad continua, sino una **etiqueta**.

Desde una perspectiva empresarial, la clasificación aparece en muchos problemas cotidianos. Puede utilizarse para decidir si un correo es **spam** o no lo es, si una transacción es **sospechosa** o no lo es, si una solicitud debe pasar a **revisión manual** o puede aprobarse de forma automática, o si una incidencia pertenece a una **categoría operativa concreta**. En todos estos casos, el sistema recibe información de entrada y debe decidir entre varias clases posibles.

A veces solo existen dos opciones. En ese caso se habla de **clasificación binaria**. Otras veces hay varias categorías posibles, como ocurre cuando se clasifica un documento, una consulta o una reclamación en distintos grupos. También puede suceder que un mismo caso reciba varias etiquetas al mismo tiempo, por ejemplo, cuando un documento se marca como **urgente, confidencial y jurídico**.

Para entender bien la lógica de estos algoritmos, conviene fijarse en la estructura general del problema. Un modelo de clasificación aprende, a partir de ejemplos previos, qué patrones de las variables de entrada suelen estar asociados a cada clase. Después aplica ese aprendizaje a casos nuevos.

En muchos modelos, sobre todo en problemas binarios, la salida inicial no es directamente una etiqueta, sino una **probabilidad**. Después, esa probabilidad se transforma en clase mediante un **umbral de decisión**. Este detalle es muy importante, porque muestra que la clasificación no

depende solo del modelo entrenado. También influye la forma en que la organización decide convertir la predicción en una acción operativa.

Ejemplo

Puede verse con un ejemplo sencillo. Si un modelo estima un 0,92 de probabilidad de impago y la empresa ha fijado un umbral de 0,80 para activar una revisión, ese caso se tratará como riesgo alto. Si el umbral estuviera en 0,95, el mismo caso no se clasificaría igual. Por tanto, la clasificación no depende solo del algoritmo, sino también del criterio empresarial con el que se interpreta la salida del modelo.

Para ordenar sus formas más habituales, puede resultar útil resumir algunos tipos frecuentes de clasificación:

Modalidad de clasificación	Qué hace	Ejemplo de uso empresarial
Clasificación binaria	Decide entre dos clases posibles.	Detectar si un correo es fraudulento o legítimo.
Clasificación multiclase	Decide entre varias categorías posibles.	Clasificar incidencias de atención al cliente por tema principal.
Clasificación multietiqueta	Asigna varias etiquetas a un mismo caso.	Etiquetar un documento como urgente, jurídico y confidencial al mismo tiempo.
Clasificación probabilística	Estima probabilidades antes de convertirlas en clase.	Valorar la probabilidad de abandono de un cliente antes de activar una campaña de retención.



Esta clasificación ayuda a entender que no existe una única forma de clasificar. Lo que une a todos estos casos es que el modelo debe aprender una **frontera de separación entre clases**, aunque esa frontera puede ser sencilla o muy compleja según la naturaleza de los datos.

En cuanto a los algoritmos concretos, dentro de la clasificación aparecen varias familias muy conocidas. Entre las más habituales se

encuentran las siguientes:

Familia algorítmica	Idea general de funcionamiento	Situaciones donde suele resultar útil
Regresión logística	Estima la probabilidad de pertenecer a una clase y luego la convierte en decisión mediante un umbral.	Problemas binarios interpretables, como riesgo, respuesta o fraude inicial.
Árboles de decisión	Dividen los datos mediante reglas sucesivas hasta llegar a una clase.	Contextos donde interesa interpretar el camino de decisión.

Vecinos más cercanos	Clasifican un caso según los casos más parecidos del conjunto de entrenamiento.	Problemas donde la similitud entre observaciones resulta relevante.
Naive Bayes	Utiliza supuestos probabilísticos para estimar la clase más probable.	Texto, documentos o problemas con variables sencillas y rápidas de procesar.
Máquinas de vectores de soporte	Buscan una frontera de separación entre clases.	Casos donde interesa una separación robusta entre categorías.
Redes neuronales	Aprenden relaciones no lineales y complejas entre variables y clases.	Problemas con gran volumen de datos o patrones difíciles de modelar de forma simple.

Esta tabla no significa que exista un algoritmo universalmente mejor. Más bien muestra que la clasificación puede abordarse con estrategias muy distintas. La elección depende del volumen de datos, de la necesidad de interpretación, del coste de los errores y de la complejidad del problema. También conviene recordar que la clasificación no se limita a “poner una etiqueta”. En la práctica, lo importante es **cómo se evalúa el rendimiento del sistema**. En muchos problemas empresariales, no todos los errores pesan igual. En un detector de fraude puede ser preferible aceptar algunos **falsos positivos** antes que dejar pasar fraudes reales. En cambio, en una herramienta de revisión de candidaturas, un exceso de **falsos negativos** podría hacer que se pierdan perfiles valiosos. Por eso, el valor de un modelo de clasificación siempre debe analizarse en relación con el problema organizativo en el que se utiliza.

Ejemplo

Supóngase una empresa aseguradora que recibe miles de partes de siniestro al mes. El objetivo no es calcular todavía el coste exacto de cada expediente, sino decidir qué partes deben pasar a revisión prioritaria por posible irregularidad. El problema, por tanto, es de clasificación: revisión prioritaria frente a tramitación ordinaria. Para resolverlo, el sistema puede aprender a partir de partes anteriores, variables del asegurado, historial, tipo de siniestro o incoherencias documentales. El valor del algoritmo no está en producir una cifra exacta, sino en asignar una categoría operativa que permita ordenar mejor el flujo de trabajo.

Sabías que...

Un mismo modelo de clasificación puede cambiar mucho su comportamiento si se modifica el umbral de decisión. A veces, la diferencia entre un sistema más permisivo y otro más estricto no está en el algoritmo, sino en el criterio con el que la empresa convierte probabilidades en acciones.

Actividad 19

Relaciona cada modalidad de clasificación con su ejemplo.

- a. Clasificación binaria.
- b. Clasificación multiclase.
- c. Clasificación multietiqueta.
- d. Clasificación probabilística.

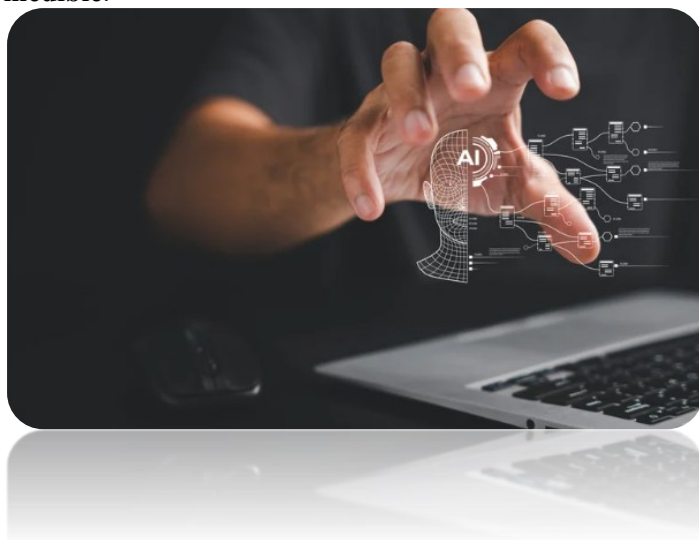
1. Estimar la probabilidad de abandono de un cliente antes de decidir si activar una campaña.
2. Decidir si un correo es legítimo o fraudulento.
3. Clasificar una reclamación en facturación, logística, producto o atención comercial.
4. Etiquetar un documento como urgente, confidencial y jurídico al mismo tiempo.

2.10. Algoritmos de regresión.

Los **algoritmos de regresión** se emplean cuando el objetivo del modelo es producir una **predicción numérica**. A diferencia de la clasificación, aquí no se intenta decidir a qué grupo pertenece un caso, sino **estimar una cantidad continua**. Esa cantidad puede ser un precio, un tiempo, una demanda, un coste, un consumo o cualquier otra magnitud medible.

En la empresa, la regresión aparece con mucha frecuencia, aunque no siempre se nombre así. Cuando una organización quiere estimar **ventas futuras, tiempo de entrega, consumo energético, demanda esperada, coste probable de una operación, valor de un inmueble o duración de una incidencia**, en realidad está planteando un problema de regresión.

Esta idea conviene dejarla muy clara: no toda salida numérica implica regresión. Hay casos en los que un sistema devuelve un número, pero ese número funciona como una **etiqueta** y no como una magnitud continua. Por ejemplo, un código postal está formado por números, pero no expresa una cantidad medible. En ese caso no estaríamos ante regresión, sino ante clasificación. Para hablar propiamente de regresión, el número debe representar una **cantidad con sentido continuo o medible**.



También es importante no confundir el nombre de algunos algoritmos con la naturaleza del problema. La **regresión lineal** encaja claramente en la predicción de valores continuos. En cambio, la **regresión logística**, pese a llevar la palabra regresión en el nombre, suele emplearse en problemas de clasificación porque genera una probabilidad que después se convierte en clase.

Para ordenar mejor las situaciones donde se utilizan estos algoritmos, puede resultar útil presentar algunos ejemplos típicos:

Problema de regresión	Qué se intenta estimar	Ejemplo empresarial
Predicción de ventas	El volumen futuro de ventas en unidades o valor monetario.	Estimar la facturación semanal de una tienda.
Predicción de tiempo	La duración esperada de un proceso o servicio.	Prever cuánto tardará en resolverse una incidencia técnica.
Estimación de coste	El importe probable asociado a una operación o evento.	Estimar el coste de una reparación o de un siniestro.
Predicción de demanda	La cantidad futura de producto o servicio requerida.	Anticipar cuántos pedidos recibirá un canal online.
Valoración de activos	El valor monetario esperado de un bien.	Estimar el precio de venta de un inmueble o de un vehículo.

Esta ordenación muestra que la regresión está muy ligada a la **planificación**, a la **gestión de recursos** y a la **previsión operativa**. Allí donde la empresa necesita una cifra para tomar decisiones,

asignar capacidad o valorar escenarios, la regresión suele ser una de las familias de algoritmos más útiles.

Desde el punto de vista de las técnicas, la regresión puede abordarse con distintos enfoques. Entre los más habituales se encuentran los siguientes:

Familia algorítmica	Idea general	Cuándo suele tener sentido
Regresión lineal	Busca una relación lineal entre variables de entrada y variable objetivo.	Cuando interesa una base simple, interpretable y útil como punto de partida.
Regresiones regularizadas	Ajustan modelos lineales controlando complejidad y peso de variables.	Cuando hay muchas variables o riesgo de sobreajuste.
Vecinos más cercanos para regresión	Estiman el valor de un caso según valores de casos parecidos.	Cuando la similitud entre observaciones aporta información útil.
Árboles y ensamblados	Dividen el espacio de variables o combinan varios modelos para captar relaciones más complejas.	Cuando la relación entre variables no es lineal y el problema resulta más irregular.
Redes neuronales	Aprenden funciones no lineales complejas a partir de suficientes datos.	Cuando el fenómeno es complejo y existe volumen de datos adecuado.

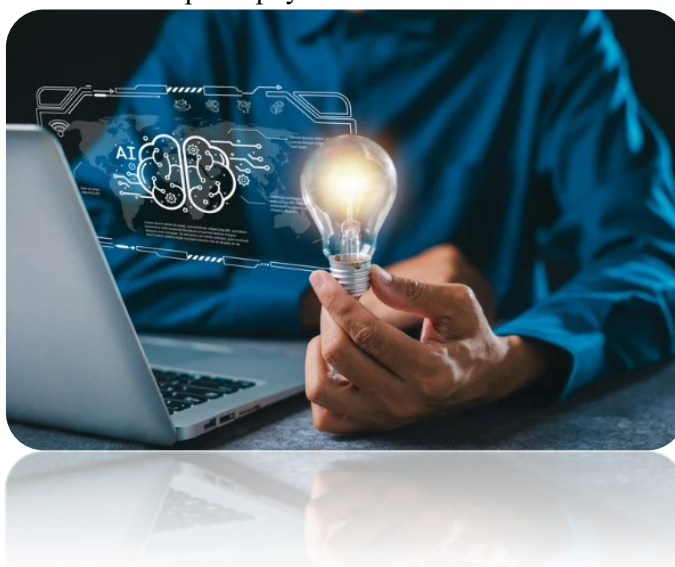
Esta diversidad técnica es importante porque no todas las variables empresariales se comportan de manera lineal ni estable. Algunas responden razonablemente bien a un modelo sencillo. Otras dependen de interacciones complejas entre tiempo, contexto, temporada, cliente, canal o ubicación. Por eso la regresión debe entenderse como una **familia de problemas y de algoritmos**, y no como una única fórmula fija.

En términos operativos, la calidad de un modelo de regresión se juzga por la **distancia entre la predicción y el valor real**. A diferencia de la clasificación, aquí no se trata de acertar o fallar una etiqueta, sino de medir cuánto se aparta el sistema de la cantidad verdadera. Por eso, en regresión adquieren especial importancia medidas del error medio.

Esto cambia también la forma de interpretar el valor del modelo. En una clasificación, el foco está en si el sistema separa bien unas clases de otras. En una regresión, lo importante es saber si la cifra estimada resulta lo bastante cercana a la realidad como para apoyar una decisión útil.

Ejemplo

Supóngase una empresa de distribución que quiere prever el tiempo estimado de entrega de cada pedido para mejorar su promesa comercial y ajustar recursos. La variable objetivo es un número continuo: horas o días de entrega. El modelo puede aprender a partir de variables como origen, destino, franja horaria, volumen del pedido, histórico de tráfico, incidencias previas, saturación del almacén o meteorología. El resultado no será una categoría, sino una **cifra estimada**.



Esa cifra podrá servir para informar a la clientela, reorganizar rutas o activar alertas si el tiempo esperado supera cierto umbral.

En la práctica, muchas decisiones empresariales que parecen simples dependen de buenas predicciones numéricas. Saber si una tienda venderá 200 o 350 unidades la semana que viene, o si una incidencia tardará 2 horas o 18 en resolverse, cambia por completo la planificación. Por eso la regresión tiene tanto peso en previsión, valoración y gestión operativa.

Sabías que...

Un modelo de regresión puede ser técnicamente correcto y, aun así, resultar poco útil si el error medio que comete es demasiado alto para el contexto de negocio. No basta con que prediga “más o menos bien”. Tiene que hacerlo con la precisión suficiente para que la empresa pueda tomar decisiones razonables.

Actividad 20

Completa las frases con las palabras correctas: **continua, cantidad, clase, regresión**.

- Los algoritmos de _____ se utilizan cuando el objetivo es estimar un valor numérico.
- En este tipo de problema, el sistema intenta predecir una _____ y no una etiqueta.
- La variable objetivo debe tener sentido _____ o medible.
- Si el objetivo es decidir a qué grupo pertenece un caso, entonces hablamos de una _____ y no de regresión.

2.11. Algoritmos de agrupamiento.

Los **algoritmos de agrupamiento**, también conocidos como **clustering**, suelen situarse dentro del **aprendizaje no supervisado**. Su rasgo principal es muy claro: trabajan con **datos no etiquetados**. Esto significa que, a diferencia de la clasificación o de la regresión, el sistema **no recibe una respuesta correcta previa**. No sabe de antemano qué grupo corresponde a cada caso. Su tarea consiste en **descubrir estructuras, similitudes o agrupaciones** dentro de los datos.

Dicho de una forma sencilla, el agrupamiento intenta responder a una pregunta básica: **qué observaciones se parecen lo suficiente entre sí como para formar grupos relativamente coherentes**. Es importante subrayar que esos grupos **no vienen definidos desde el principio**. El algoritmo no aprende una etiqueta conocida, sino que propone agrupaciones a partir de la **similitud**, la **distancia** o la **densidad** de los datos.

Por eso el clustering resulta tan útil en tareas de **análisis exploratorio, segmentación y descubrimiento de patrones**. En la empresa puede aplicarse, por ejemplo, a la segmentación de clientela, a la agrupación de productos según comportamiento de compra, a la identificación de perfiles de uso de una plataforma, al análisis de tipologías de incidencias o a la detección de patrones de comportamiento en operaciones.

En todos estos casos, la organización no pide al modelo que decida si un caso pertenece a una clase ya conocida. Lo que busca es que el sistema **revele grupos útiles dentro del conjunto de datos**. Para ordenar mejor su lógica, puede resumirse así:

Aspecto del clustering	Significado práctico
No hay etiquetas previas	El algoritmo no recibe una respuesta correcta durante el entrenamiento.
Busca estructura en los datos	Intenta descubrir similitudes, proximidades o densidades que permitan formar grupos.
El resultado son clusters o grupos internos	Cada observación queda asociada a un grupo descubierto por el algoritmo.
Se usa mucho con fines exploratorios o de segmentación.	Ayuda a comprender mejor la composición del conjunto de datos.

Esta estructura hace que el agrupamiento tenga un papel distinto al de la clasificación. En clasificación, las clases ya están definidas por la organización o por el problema. En clustering, los grupos **emergen del análisis del propio conjunto de datos**. Esa diferencia es muy importante, porque también cambia la forma de interpretar el resultado.

Un **cluster** no es, por sí mismo, una verdad absoluta. Es una agrupación útil según un criterio de similitud determinado. Por eso la interpretación de los grupos exige siempre una **lectura contextual** y un buen conocimiento del negocio. El algoritmo puede formar grupos, pero corresponde a la empresa decidir si esos grupos tienen realmente sentido operativo.

En cuanto a las familias algorítmicas, existen varios enfoques de agrupamiento. Cada uno parte de una idea distinta sobre qué significa formar grupos. Algunos se basan en **centroides**, otros en **jerarquías** y otros en **densidades**.

Con fines didácticos, pueden resumirse así:

Familia de agrupamiento	Idea de fondo	Rasgo distintivo
K-means	Forma grupos alrededor de centroides representativos.	Resulta útil cuando se busca una partición clara en un número prefijado de grupos.
Agrupamiento jerárquico	Construye una estructura de grupos en forma de árbol mediante fusiones o divisiones sucesivas.	Ayuda a ver relaciones jerárquicas entre observaciones o grupos.
DBSCAN	Considera los grupos como zonas de alta densidad separadas por zonas de baja densidad.	Puede encontrar grupos de formas irregulares y distinguir ruido o puntos aislados.
Mezclas gaussianas	Modelan los datos como combinación de distribuciones probabilísticas.	Ofrecen una visión más flexible y probabilística de la pertenencia a grupos.



Esta comparación es muy útil porque muestra que **no existe una sola idea de grupo**. En algunos problemas interesa dividir claramente el conjunto en varios segmentos manejables. En otros, lo importante es detectar estructuras más irregulares o identificar observaciones aisladas que no encajan bien en ningún grupo. Desde un punto de vista empresarial, esta diversidad importa mucho. Si una empresa quiere segmentar clientes para fines comerciales, puede resultarle suficiente una partición relativamente clara y estable, donde un algoritmo como **K-means** sea útil. En cambio, si lo que busca es descubrir

comportamientos anómalos, estructuras más desordenadas o grupos poco compactos, puede convenir más un enfoque basado en densidad.

Por tanto, el algoritmo adecuado depende del tipo de estructura que se espera encontrar y del uso que se quiera dar al resultado.

Ahora bien, precisamente porque trabaja sin etiquetas previas, el agrupamiento exige prudencia. El hecho de que un algoritmo devuelva grupos no significa automáticamente que esos grupos tengan un valor estratégico claro. La interpretación humana y el conocimiento del negocio son decisivos para evitar conclusiones forzadas o artificiales.

Por eso, el clustering suele ser especialmente valioso como **herramienta exploratoria** y de apoyo al análisis, más que como respuesta definitiva separada del contexto.

En síntesis, los algoritmos de agrupamiento permiten **descubrir estructura en datos no etiquetados** y resultan especialmente útiles cuando la organización quiere segmentar, explorar o comprender patrones sin partir de una clasificación previa. Su valor reside en que ayudan a ver los datos con una lógica distinta: no preguntan qué clase corresponde a cada caso, sino **qué grupos emergen de su parecido o proximidad**.

Ejemplo

Supóngase una cadena de comercio electrónico que quiere comprender mejor a su clientela. No dispone de etiquetas previas que indiquen “tipo de cliente”, pero sí tiene datos sobre frecuencia de compra, ticket medio, categorías adquiridas, uso de promociones, devoluciones y canales de contacto. Un algoritmo de agrupamiento puede revelar que existe un grupo muy sensible al precio, otro de alto valor y baja frecuencia, otro muy fiel a pocas categorías y otro con comportamiento muy irregular. Ninguno de esos grupos estaba etiquetado antes, pero el clustering permite descubrirlos. Después, la empresa puede interpretar esos segmentos y decidir si le resultan útiles para diseñar campañas, ajustar surtido o mejorar el servicio.

Sabías que...

En clustering, elegir el número de grupos no siempre es una decisión automática. En muchos casos, la empresa debe valorar si una segmentación con tres, cuatro o cinco grupos resulta más útil para su análisis y para sus decisiones.

Actividad 21

Relaciona cada familia de agrupamiento con su rasgo principal.

- a. K-means.
 - b. Agrupamiento jerárquico.
 - c. DBSCAN.
 - d. Mezclas gaussianas.
1. Ofrece una visión probabilística de la pertenencia a los grupos.
 2. Forma grupos alrededor de centroides.
 3. Construye relaciones entre grupos en forma de árbol.
 4. Detecta zonas de alta densidad y puede separar ruido.

2.12. Algoritmos de recomendación.

Los **algoritmos de recomendación** forman una familia especialmente importante en entornos digitales y en modelos de negocio centrados en la **personalización**. Su función principal consiste en **seleccionar, para cada persona o situación, un conjunto reducido de elementos relevantes** dentro de un catálogo mucho más amplio.

EDITORIAL TUTOR FORMACIÓN

Esta idea es muy importante, porque muestra que recomendar no consiste simplemente en ordenar todos los productos o contenidos posibles. El verdadero reto está en decidir **qué pocos elementos tienen más sentido para una persona concreta** dentro de una colección enorme de opciones. Dicho de una manera sencilla, un sistema de recomendación intenta responder a preguntas como estas:

¿Qué curso puede interesar más a esta alumna?

¿Qué producto conviene mostrar primero a este cliente?

¿Qué documento puede resultar más útil para esta persona en este momento?

Por eso, los algoritmos de recomendación son especialmente valiosos cuando la empresa trabaja con **muchos elementos disponibles** y necesita convertir esa abundancia en una **selección relevante y manejable**.

En términos generales, un sistema de recomendación suele organizarse en varias fases. De forma simplificada, puede resumirse así:

Fase	Función principal	Sentido operativo
Generación de candidatos	Reduce un catálogo enorme a un subconjunto de opciones plausibles.	Hace viable el sistema a gran escala.
Scoring o puntuación	Asigna una relevancia estimada a cada candidato.	Permite ordenar las opciones según probabilidad de interés.
Re-ranking o reajuste final	Ajusta la lista final incorporando factores adicionales.	Mejora diversidad, novedad, frescura o restricciones del negocio.

Esta estructura deja ver que recomendar no equivale simplemente a calcular parecidos. En la práctica, suele ser un proceso en varias etapas. Primero hay que reducir el universo de opciones. Después hay que valorar cuáles parecen más relevantes. Y, por último, muchas veces hay que reajustar la lista final para evitar repeticiones, introducir diversidad o respetar determinados criterios estratégicos.

Por tanto, la recomendación no se limita a una sola operación matemática. Es una tarea compleja en la que entran en juego **escala, relevancia, contexto y experiencia de uso**.

En cuanto a los enfoques principales, suelen distinguirse tres grandes líneas:

Enfoque de recomendación	Cómo funciona	Punto fuerte principal
Filtrado basado en contenido	Recomienda elementos parecidos a los que ya gustan, utilizando características del contenido y del historial de la persona usuaria.	Personalización directa a partir de rasgos del contenido.
Filtrado colaborativo	Recomienda a partir de similitudes entre personas y elementos, aprendiendo patrones conjuntos de preferencia.	Puede descubrir afinidades no evidentes y generar recomendaciones más sorprendentes.
Enfoques híbridos	Combinan señales de contenido, comportamiento y contexto.	Suelen equilibrar mejor cobertura, personalización y robustez.

El **filtrado basado en contenido** parte de una lógica bastante intuitiva. Si una persona ha mostrado interés por ciertos elementos, el sistema le recomienda otros con características parecidas. Por ejemplo, si una alumna consulta varios cursos de ciberseguridad básica, el sistema puede sugerirle otros contenidos similares dentro de esa misma línea temática.

El **filtrado colaborativo**, en cambio, funciona de otra manera. En lugar de fijarse solo en las características del contenido, utiliza patrones de comportamiento de muchas personas usuarias. Si

dos personas presentan comportamientos similares, el sistema puede recomendar a una contenidos que le gustaron a la otra, aunque no parezcan similares a simple vista. Esta estrategia resulta especialmente interesante porque puede descubrir afinidades ocultas o preferencias no tan evidentes. En muchos sistemas reales, la solución más útil termina siendo **híbrida**, ya que combina información del contenido, señales de comportamiento y datos del contexto. Esto permite construir recomendaciones más equilibradas y adaptadas a situaciones reales de uso.



Un elemento muy importante en los recomendadores modernos es la representación mediante **embeddings**, es decir, representaciones numéricas compactas que permiten situar personas usuarias y elementos en un espacio común de similitud. Dicho de un modo sencillo, el sistema aprende a representar clientes, productos, cursos o documentos de forma que los elementos más cercanos en ese espacio tengan también una relación más fuerte en términos de interés o afinidad.

Esta idea es importante porque muestra que los sistemas de recomendación actuales no se basan solo en reglas sencillas, sino también en representaciones matemáticas capaces de condensar señales complejas de preferencia.

Desde la perspectiva empresarial, los recomendadores resultan útiles mucho más allá de las plataformas de vídeo o de música. También pueden aplicarse en:

- **Comercio electrónico.**
- **Formación digital.**
- **Banca y seguros.**
- **Catálogos documentales.**
- **Plataformas de contenidos corporativos.**
- **Sugerencia de acciones comerciales.**

Siempre que exista un conjunto amplio de opciones y una necesidad de seleccionar qué puede resultar más relevante para una persona concreta o para un contexto de uso determinado, la lógica de recomendación puede aportar valor.

También conviene subrayar que los sistemas de recomendación no buscan solo relevancia máxima. En muchos casos deben equilibrar varios objetivos a la vez. A veces interesa mostrar elementos nuevos, evitar repetir siempre lo mismo, respetar reglas comerciales o introducir cierta diversidad. Por eso, la mejor recomendación no siempre es la de máxima probabilidad bruta, sino la que encaja mejor en la experiencia general de la persona usuaria y en los objetivos del negocio.

Por último, los recomendadores también requieren métricas específicas. En este tipo de sistemas no basta con saber si el modelo acierta “en general”. Muchas veces lo importante es si los elementos más útiles aparecen en los **primeros puestos**, porque ahí es donde la persona usuaria presta realmente atención. Esto cambia la forma de evaluar su rendimiento y explica por qué la recomendación tiene una lógica propia dentro del aprendizaje automático.

En definitiva, los algoritmos de recomendación constituyen una familia central en los entornos donde la empresa necesita **seleccionar, personalizar y priorizar** elementos para cada persona o situación. Su valor no consiste solo en sugerir productos, sino en **organizar la abundancia de opciones y transformarla en decisiones relevantes a escala**. Por eso se han convertido en una pieza clave de la inteligencia artificial aplicada a catálogos, contenidos, ofertas y experiencias digitales personalizadas.

Ejemplo

Supóngase una empresa de formación online con cientos de cursos, itinerarios y recursos de apoyo. Mostrar el catálogo completo a cada estudiante tiene poco sentido. Un recomendador puede generar primero candidatos a partir del historial del alumno, de su sector profesional o del contenido que ya ha consultado. Después puede puntuar esos candidatos según probabilidad de interés o finalización. Finalmente, puede reordenar la lista para introducir variedad, novedad o prioridades estratégicas de la plataforma. El resultado no es una simple lista alfabética, sino una selección personalizada y operativa.

Sabías que...

Una buena recomendación no siempre es la más obvia. A veces los sistemas más útiles son precisamente los que combinan relevancia con cierta capacidad de descubrimiento, para que la persona usuaria encuentre opciones que probablemente no habría buscado por sí sola.

Actividad 22

Completa las frases con las palabras correctas: candidatos, contenido, colaborativo, re-ranking.

- En la primera fase, el sistema reduce el catálogo a un conjunto de _____.
- El filtrado basado en _____ recomienda elementos parecidos a los que ya interesan a la persona usuaria.
- El filtrado _____ aprovecha patrones compartidos entre personas y elementos.
- La fase de _____ ajusta la lista final incorporando factores como diversidad o frescura.

2.13. Algoritmos de búsqueda, optimización y heurísticas.

Los **algoritmos de búsqueda y optimización** se utilizan cuando la cuestión principal no es **a qué clase pertenece un caso** ni **qué valor numérico tomará una variable**, sino **qué solución conviene elegir entre muchas alternativas posibles**. En este tipo de problemas, lo importante es recorrer un espacio de opciones, comprobar cuáles son válidas y, cuando sea posible, encontrar la mejor según un criterio determinado. Herramientas como OR-Tools se orientan precisamente a este tipo de situaciones y trabajan con áreas como optimización lineal, programación entera, programación por restricciones, flujos, asignación, empaquetado, planificación y rutas.

En términos generales, un **problema de búsqueda** aparece cuando existe un conjunto amplio de estados, caminos o configuraciones posibles y se necesita localizar una solución válida. Un **problema de optimización** aparece cuando, además de encontrar una solución factible, interesa encontrar la mejor según un criterio objetivo, por ejemplo minimizar tiempo, distancia o coste, o maximizar beneficio, cobertura o eficiencia. En algunos casos, además, lo primero no es optimizar, sino simplemente conseguir una combinación que cumpla todas las restricciones. La programación por restricciones trabaja muy bien en ese terreno: parte de un conjunto enorme de candidatos y busca soluciones compatibles con las condiciones del problema.

Esta diferencia entre **factibilidad** y **optimización** es muy importante. No todos los problemas empresariales exigen desde el inicio una solución óptima en sentido estricto. A veces la necesidad principal consiste en encontrar una planificación válida, una asignación compatible o una secuencia posible. Después, si el contexto lo permite, esa solución puede mejorarse. Dicho de otro modo, en muchos entornos reales primero hay que conseguir que el sistema funcione y después ya se intenta que funcione mejor.

EDITORIAL TUTOR FORMACIÓN

Para ordenar mejor este campo, conviene distinguir tres ideas relacionadas, pero no idénticas:

Idea central	Qué persigue	Ejemplo orientativo
Búsqueda	Encontrar una solución dentro de un espacio de posibilidades.	Hallar una secuencia válida de tareas que cumpla restricciones.
Optimización	Encontrar la mejor solución según un criterio objetivo.	Asignar rutas minimizando distancia o tiempo total.
Heurística	Llegar a soluciones buenas de forma práctica sin explorar exhaustivamente todas las posibilidades.	Generar una solución inicial rápida para reparto o planificación.

Esta distinción ayuda a entender por qué estos algoritmos tienen tanto valor en la empresa. Cuando el número de combinaciones posibles crece muy deprisa, explorar todas las alternativas deja de ser inviable en la práctica. En esos escenarios, se necesitan estrategias que permitan avanzar con rapidez hacia soluciones razonables sin revisar absolutamente todo el espacio de búsqueda.



Aquí entran en juego las **heurísticas**. Una heurística es una estrategia práctica que ayuda a encontrar soluciones buenas sin necesidad de recorrer exhaustivamente todas las opciones ni de demostrar siempre que se ha alcanzado el óptimo global. En problemas reales, esta lógica resulta muy útil. En OR-Tools, por ejemplo, el proceso puede comenzar con una **estrategia de primera solución** y continuar después con técnicas de **búsqueda local** o con **metaheurísticas** que intentan mejorar esa primera propuesta.

Entre las estrategias más conocidas se encuentran las siguientes:

Estrategia	Función principal	Sentido práctico
Solución inicial	Encuentra una primera alternativa factible.	Permite arrancar el proceso sin esperar a una búsqueda exhaustiva.
Descenso voraz	Acepta mejoras locales mientras reduzcan el coste.	Resulta útil cuando interesa una mejora rápida y directa.
Guided Local Search	Introduce mecanismos para escapar de mínimos locales.	Suele ser muy eficaz en problemas de rutas.
Simulated Annealing	Acepta ciertos empeoramientos temporales para salir de atascos locales.	Ayuda a explorar regiones distintas del espacio de soluciones.

Tabu Search	Evita volver repetidamente a soluciones parecidas ya exploradas.	Favorece una exploración más amplia del problema.
--------------------	--	---

Esta tabla no significa que todas las heurísticas sirvan igual para cualquier caso. Cada una responde a una lógica distinta de exploración y mejora. Lo importante es entender que las heurísticas no son un recurso menor, sino una herramienta muy valiosa cuando el problema es demasiado grande o demasiado costoso como para resolverse de forma exacta en tiempos razonables. OR-Tools distingue precisamente entre estrategias de primera solución y metaheurísticas de mejora como **GREEDY_DESCENT**, **GUIDED_LOCAL_SEARCH**, **SIMULATED_ANNEALING**, **TABU_SEARCH** y **GENERIC_TABU_SEARCH**.

Según un punto de vista empresarial, estos algoritmos resultan muy útiles en problemas como la **planificación de turnos**, la **asignación de tareas**, el **reparto**, la **secuenciación de órdenes de producción**, la **ubicación de recursos**, el **empaquetado** o la **configuración de rutas logísticas**. Son problemas en los que la organización no necesita que el sistema prediga una etiqueta o un valor, sino que construya una decisión operativa compatible con restricciones reales.

También conviene subrayar que no existe una sola técnica universal para todos estos problemas. En unos casos encajará mejor la **programación lineal**. En otros, la **programación entera**. En otros, la **programación por restricciones** o los enfoques específicos de **rutas y búsqueda local**. La elección depende de cómo esté formulado el problema, del tipo de restricciones y del criterio que se quiera optimizar.

Los algoritmos de **búsqueda, optimización y heurísticas** forman una familia fundamental cuando la empresa necesita **planificar, asignar, secuenciar o decidir entre muchas combinaciones posibles**. Su valor no está en predecir una etiqueta o una cantidad, sino en encontrar soluciones viables y, cuando se puede, soluciones especialmente buenas dentro de espacios complejos. Por eso son esenciales en áreas donde el tiempo, el coste, la distancia o la compatibilidad entre restricciones marcan una diferencia real en la operación diaria.

Ejemplo

Supóngase una empresa de reparto que tiene varias furgonetas, múltiples direcciones de entrega, restricciones horarias y objetivos de eficiencia. El problema no consiste en clasificar pedidos ni en segmentar clientes, sino en decidir qué ruta seguirá cada vehículo y en qué orden visitará cada punto. En este caso, la empresa necesita una solución viable y, si es posible, una solución especialmente buena en términos de tiempo, distancia o equilibrio de carga. Aquí la búsqueda y la optimización pasan a ser totalmente operativas.

Otro ejemplo muy claro aparece en la planificación de personal. Cuando una organización debe construir turnos, respetar descansos, cubrir necesidades mínimas y cuadrar jornadas, el problema no se resuelve aprendiendo a partir de ejemplos etiquetados, sino explorando combinaciones compatibles con múltiples reglas. En este contexto, la programación por restricciones encaja especialmente bien porque trabaja precisamente sobre variables, restricciones y soluciones factibles.

Sabías que...

En muchos problemas reales, conseguir una primera solución válida ya es un avance importante. La mejora posterior puede llegar después mediante estrategias de búsqueda local o metaheurísticas.

Actividad 23

Relaciona cada concepto con su definición correcta.

- a. Búsqueda.
 - b. Optimización.
 - c. Heurística.
 - d. Programación por restricciones.
1. Encontrar una solución compatible dentro de un gran conjunto de candidatos.
 2. Encontrar la mejor solución según un criterio objetivo.
 3. Estrategia práctica para llegar a soluciones buenas sin explorar todo el espacio.
 4. Localizar una solución dentro de un espacio amplio de posibilidades.

2.14. Criterios para elegir una técnica de IA en función del problema empresarial.

Elegir una técnica de IA adecuada no consiste en empezar por el algoritmo más llamativo ni por la herramienta más reciente. La elección correcta depende, ante todo, de **cómo se formula el problema**. La propia formación de Google sobre **Problem Framing** insiste en que primero hay que traducir un problema del mundo real a un problema tratable con aprendizaje automático, definiendo el objetivo ideal, el papel del modelo y los criterios de éxito. Esta idea es fundamental, porque deja claro que el punto de partida no es la tecnología, sino la formulación correcta de la necesidad empresarial.

Esta lógica encaja muy bien con la guía de scikit-learn para **elegir el estimador adecuado**. Ese mapa de decisión no pretende dar una respuesta exacta para todos los casos, pero sí ofrece una orientación muy útil: antes de pensar en modelos concretos, conviene aclarar si se quiere **predecir una categoría, predecir una cantidad, trabajar con datos etiquetados o no etiquetados** y si se dispone de un volumen mínimo razonable de muestras. Incluso comienza con una comprobación tan básica como contar con **más de 50 muestras** antes de entrar en ciertas decisiones del flujo.

A partir de esa idea, el primer criterio debe ser siempre la **naturaleza de la pregunta de negocio**. No es lo mismo querer clasificar incidencias, estimar ventas, descubrir segmentos de clientela, recomendar productos o planificar rutas. Cada uno de esos problemas remite a una familia técnica distinta. Por eso, el criterio correcto no es preguntarse qué modelo está de moda, sino qué tipo de salida necesita realmente la organización.

Con ese punto de partida, puede resumirse la relación entre problema empresarial y familia técnica de la siguiente manera:

Tipo de problema empresarial	Pregunta principal	Familia técnica que suele encajar mejor
Clasificación	¿A qué categoría pertenece este caso?	Algoritmos de clasificación supervisada.
Predicción numérica	¿Qué valor o cantidad se espera?	Algoritmos de regresión.
Segmentación	¿Qué grupos aparecen sin etiquetas previas?	Algoritmos de agrupamiento o clustering.
Personalización	¿Qué elemento conviene sugerir a cada usuario?	Sistemas de recomendación.
Planificación y asignación	¿Qué combinación o secuencia de decisiones es más adecuada?	Búsqueda, optimización y heurísticas.
Comprensión de texto, imagen o audio.	¿Cómo extraer significado o producir salidas a partir de datos no estructurados?	NLP, visión por computador, deep learning u otras técnicas específicas del dato.



Esta tabla obliga a formular bien el problema antes de pensar en la herramienta. Si una empresa afirma que quiere **usar IA para mejorar ventas**, todavía no ha definido de verdad el problema. Puede que necesite una recomendación personalizada, una regresión para prever demanda, una clasificación para detectar riesgo de abandono o una optimización para ajustar promociones y stock. La clave no está en el objetivo general, sino en la precisión con la que se define la tarea concreta.

Un segundo criterio decisivo es el **tipo de dato disponible**. No se trabaja igual con tablas estructuradas, con texto, con imágenes, con señales de sensores o con historiales de interacción. El formato del dato condiciona mucho qué modelos tienen más sentido y qué técnicas pueden aprovechar mejor la información. Esta idea aparece de forma muy clara tanto en la organización de la formación de Google, que separa cursos por dominios y tipos de problema, como en la propia estructura de scikit-learn.

En términos prácticos, esta relación puede resumirse así:

Tipo de dato predominante	Técnicas que suelen tener mejor encaje
Datos tabulares estructurados	Clasificación, regresión, árboles, bosques, boosting y otros modelos supervisados clásicos.
Texto y documentos	NLP, clasificación de texto, embeddings, modelos de lenguaje y sistemas generativos según el caso.
Imagen y vídeo	Visión por computador y, con frecuencia, deep learning.
Historiales de interacción	Recomendación, ranking y modelos de comportamiento.
Restricciones, recursos y combinaciones.	Optimización, programación lineal, entera o por restricciones, además de heurísticas.

Este criterio evita errores muy comunes. No tiene sentido intentar resolver con clustering un problema que en realidad necesita una predicción supervisada. Tampoco conviene forzar un modelo generativo cuando la tarea principal consiste en asignar turnos compatibles o construir rutas bajo restricciones. Elegir bien significa **alinear problema, dato y salida esperada**.

Un tercer criterio fundamental es la **cantidad y calidad de los datos**. La guía de scikit-learn deja claro desde el inicio que el volumen de muestras influye en la decisión sobre qué modelos probar. Esto no significa que exista un número mágico válido para todos los proyectos, pero sí que la disponibilidad de datos útiles es una condición básica de viabilidad. Si hay pocos ejemplos, si están mal etiquetados o si no reflejan bien el proceso real, algunas técnicas perderán fiabilidad o dejarán de ser razonables.

Un cuarto criterio tiene que ver con la **necesidad de interpretación, confianza y gobernanza**. No siempre basta con que un modelo tenga buen rendimiento. En muchos contextos empresariales también hace falta entender por qué decide, cómo se supervisa y qué consecuencias puede tener un error. El marco de gestión del riesgo de NIST insiste precisamente en incorporar consideraciones de **trustworthiness** en el diseño, el desarrollo, el uso y la evaluación de sistemas de IA. Eso obliga a valorar explicabilidad, control, riesgo y adecuación al contexto, y no solo precisión técnica.

EDITORIAL TUTOR FORMACIÓN

Desde esta perspectiva, antes de elegir una técnica conviene plantearse al menos estas preguntas:

Pregunta de decisión	Por qué importa
¿Se necesita una salida categórica, numérica, segmentada, recomendada o planificada?	Define la familia básica de técnicas candidatas.
¿Los datos están etiquetados o no?	Separa con rapidez aprendizaje supervisado y no supervisado.
¿Qué formato tienen los datos?	Orienta hacia modelos tabulares, de texto, imagen, recomendación u optimización.
¿Cuántos datos útiles existen y con qué calidad?	Condiciona viabilidad, estabilidad y rendimiento.
¿Hace falta interpretar la decisión con claridad?	Puede favorecer modelos más trazables o reglas explícitas.
¿Qué coste tiene equivocarse?	Influye en umbrales, métricas y tolerancia al riesgo.
¿La empresa necesita predecir o necesita optimizar una decisión?	Ayuda a diferenciar aprendizaje automático predictivo de optimización y búsqueda.

Esta tabla tiene mucho valor práctico porque desplaza la conversación desde **qué técnica parece más avanzada** hacia **qué necesita de verdad el proceso empresarial**. En muchos casos, el mejor primer paso no es un modelo complejo, sino una técnica suficientemente adecuada, bien definida y compatible con los recursos de la organización. Esa idea está muy presente tanto en el enfoque de **Problem Framing** de Google como en la guía orientativa de scikit-learn.

Un quinto criterio importante es distinguir entre **problemas de predicción** y **problemas de decisión**. Esta diferencia suele pasarse por alto. Un modelo puede predecir muy bien la demanda futura de un producto y, aun así, no resolver directamente cómo debe repartirse el stock entre distintos almacenes. Lo primero es una tarea de regresión. Lo segundo puede ser un problema de optimización. Del mismo modo, un sistema puede clasificar incidencias con bastante acierto, pero la decisión sobre cómo asignar recursos para resolverlas puede exigir técnicas de planificación o de programación por restricciones. Separar predicción y decisión evita muchos proyectos mal planteados.

Otro criterio muy relevante es el **nivel de madurez y recursos de la organización**. Una solución técnicamente sofisticada puede no ser la más conveniente si la empresa no dispone de datos estables, personal preparado o procesos de seguimiento suficientes. Por eso, elegir una técnica también implica valorar si la organización podrá mantenerla, supervisarla y gobernarla con continuidad. La adecuación no es solo técnica. También es operativa y organizativa.

En consecuencia, elegir una técnica de IA adecuada exige un enfoque más disciplinado de lo que parece a primera vista. No se trata de escoger entre siglas o herramientas de forma aislada, sino de alinear **el problema real, el tipo de salida esperada, la naturaleza del dato, el nivel de riesgo aceptable y la capacidad operativa de la empresa** para implantar la solución. Cuando esas dimensiones se analizan juntas, la elección técnica deja de ser una cuestión puramente tecnológica y pasa a convertirse en una decisión de diseño organizativo.

Por tanto, el criterio central para elegir una técnica de IA no es la complejidad aparente del modelo, sino su **adecuación al problema empresarial bien formulado**. Ese ajuste entre tarea, dato, riesgo y contexto es lo que permite decidir con sentido si conviene una clasificación, una regresión, un agrupamiento, una recomendación, una técnica de NLP, un modelo profundo o una estrategia de búsqueda y optimización. Desde ahí, la IA deja de ser un catálogo de herramientas sueltas y pasa a entenderse como un conjunto de técnicas que solo tienen valor cuando encajan de verdad con la necesidad operativa que se pretende resolver.

Ejemplo

Supóngase una empresa de mantenimiento industrial que quiere usar IA para reducir paradas. Si formula mal el problema, puede escoger una sola técnica y esperar demasiado de ella. En realidad, ese objetivo puede descomponerse en varias tareas. Una regresión puede estimar el tiempo esperado hasta una avería. Una clasificación puede decidir si un caso debe tratarse como riesgo alto o bajo. Un algoritmo de agrupamiento puede descubrir perfiles de fallo en máquinas similares. Y una técnica de optimización puede decidir en qué orden asignar equipos de mantenimiento y ventanas de intervención. La técnica correcta depende, por tanto, del subproblema exacto que se quiera resolver en cada fase.

Sabías que...

Un proyecto de IA puede fracasar no porque el algoritmo sea malo, sino porque el problema estaba mal planteado desde el principio. Formular bien la tarea suele ser tan importante como elegir el modelo.

Actividad 24

Completa las frases con las palabras correctas: problema, datos, predicción, optimización.

- a. El primer paso para elegir una técnica de IA adecuada es formular bien el _____ empresarial.
- b. El tipo de _____ disponible condiciona mucho qué técnica puede encajar mejor.
- c. Estimar una cantidad futura suele remitir a un problema de _____.
- d. Asignar recursos o construir la mejor secuencia posible suele remitir a un problema de _____.

3. Uso del Machine/Deep Learning.

Una parte muy importante de la inteligencia artificial actual se basa en sistemas capaces de **aprender a partir de datos**. Este cambio ha modificado profundamente la forma de construir soluciones tecnológicas, ya que en muchos casos deja de ser necesario programar manualmente todas las reglas para cada situación posible. En su lugar, se trabaja con modelos que identifican **patrones, relaciones y regularidades** en los datos para generar predicciones, clasificaciones o decisiones útiles dentro de un proceso determinado.

Comprender esta lógica exige explicar qué significa aprender a partir de datos, qué diferencias existen entre **Machine Learning** y **Deep Learning**, qué tipos de aprendizaje automático pueden utilizarse y cómo se desarrolla un proyecto de este tipo desde la preparación del dato hasta la evaluación del modelo. También resulta necesario abordar los errores más frecuentes, las condiciones que hacen viable su implantación y la forma en que estas técnicas se aplican en problemas empresariales reales, donde el rendimiento del sistema depende tanto del modelo como de la calidad del proceso en el que se integra.

3.1. Qué es aprender a partir de datos.

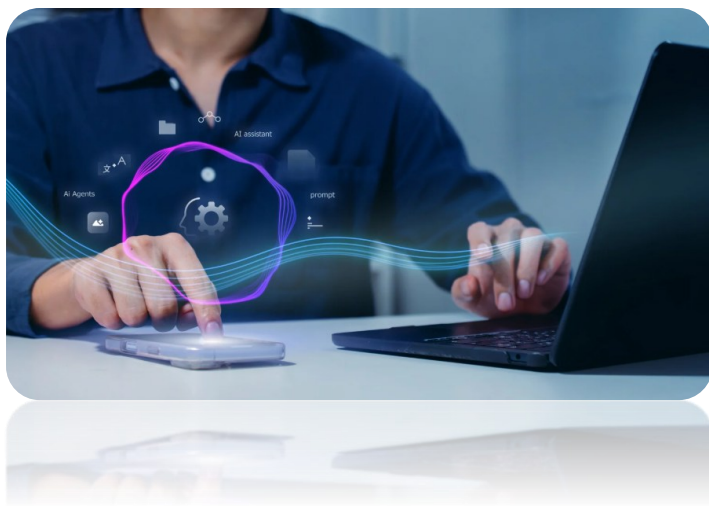
Aprender a partir de datos significa construir un modelo que, después de observar ejemplos, sea capaz de producir una salida útil cuando se enfrenta a casos nuevos. Dicho de una forma sencilla, el sistema no recibe una lista completa de instrucciones cerradas para cada situación, sino que ajusta su comportamiento a partir de ejemplos previos. En el enfoque de Machine Learning de Google, el modelo se entrena con datos para hacer predicciones útiles o generar contenido, y después se aplica a nuevos casos mediante inferencia.

Esta idea puede parecer abstracta al principio, pero en realidad responde a una lógica bastante concreta. Un modelo recibe datos de entrada, busca regularidades durante el entrenamiento y, una vez ajustado, aplica ese aprendizaje a ejemplos nuevos. Lo importante es que el sistema no se limita a memorizar el pasado, sino que intenta captar relaciones que puedan **generalizarse** a situaciones no vistas previamente. Esa capacidad de generalización es una de las bases más importantes del aprendizaje automático.

La diferencia con el software tradicional resulta especialmente clara. En un programa clásico, la persona programadora define las reglas de funcionamiento con bastante detalle. En cambio, cuando se aprende a partir de datos, el sistema recibe ejemplos y ajusta un modelo que después podrá aplicarse a nuevos casos. Esto permite abordar problemas donde el volumen de información o la complejidad de los patrones harían poco realista una programación puramente manual.

Para entender mejor este cambio, conviene observar los elementos básicos que suelen intervenir en cualquier proceso de aprendizaje desde datos:

Elemento básico	Función dentro del aprendizaje a partir de datos.
Datos	Proporcionan los ejemplos a partir de los cuales el sistema detecta regularidades.
Modelo	Es la estructura matemática o computacional que se ajusta durante el entrenamiento.
Entrenamiento	Es el proceso mediante el cual el modelo aprende a relacionar entradas y salidas o a captar patrones.
Inferencia	Es la aplicación del modelo ya entrenado a casos nuevos.
Evaluación	Permite comprobar si el aprendizaje obtenido resulta útil, estable y suficientemente preciso.



A partir de esta estructura se entiende mejor por qué el **dato** ocupa un lugar tan central. Aprender a partir de datos no significa acumular información sin más, sino disponer de ejemplos adecuados para que el modelo capture relaciones relevantes. Si esos datos no representan bien el fenómeno que se quiere modelar, si están mal preparados o si contienen errores importantes, el aprendizaje será deficiente. La calidad del resultado depende en gran medida de la calidad del dato y de cómo se entrena y evalúa el modelo.

También conviene dejar claro que aprender a partir de datos no equivale a **copiar el pasado** de forma mecánica. El valor real del aprendizaje automático reside en su capacidad para generalizar. Un modelo no se considera útil porque repita de memoria los datos de entrenamiento, sino porque funciona razonablemente bien cuando recibe datos no vistos durante esa fase. En la práctica empresarial, esto significa transformar información histórica en una capacidad operativa para estimar, clasificar, detectar o apoyar decisiones sobre casos futuros.

Otro aspecto importante es que no todos los procesos de aprendizaje a partir de datos funcionan igual. Existen distintas formas de aprender, según se disponga o no de etiquetas, según el sistema reciba recompensas o según el objetivo sea generar contenido nuevo. Lo que une a todas estas variantes es que el comportamiento útil del modelo surge del tratamiento de datos o de experiencia previa, y no de una lista completa de reglas escritas una por una.

En consecuencia, aprender a partir de datos puede entenderse como el **núcleo operativo del Machine Learning**. Supone construir modelos que extraen regularidades de ejemplos y que utilizan ese aprendizaje para producir salidas útiles sobre casos nuevos. Esta forma de trabajar ha cambiado profundamente el desarrollo de la IA aplicada porque permite abordar problemas donde la complejidad del patrón, la cantidad de información o la variabilidad de los casos hacen poco realista una programación manual cerrada.

Ejemplo

Supóngase una empresa de comercio electrónico que quiere anticipar qué pedidos tienen más probabilidad de retrasarse. Programar una regla fija para cada combinación de almacén, franja horaria, volumen, histórico de incidencias o saturación logística sería muy difícil. En cambio, si el sistema dispone de suficientes ejemplos previos de entregas y retrasos, puede entrenarse un modelo que aprenda patrones y que después estime el riesgo en pedidos nuevos. En este caso, aprender a partir de datos significa convertir un histórico operativo en una capacidad de predicción útil.

Sabías que...

Un modelo puede parecer muy bueno durante el entrenamiento y, aun así, fallar cuando recibe casos nuevos. Por eso la generalización es una idea mucho más importante que el simple hecho de haber aprendido sobre un conjunto de datos previo.