1. Introducción

Un dato se puede definir como una representación simbólica o numérica de un hecho o evento. Es una unidad básica de información que se utiliza para describir, medir o representar algo en un entorno determinado.

Los datos pueden ser de diferentes tipos (por ejemplo, números, texto, sonido, imágenes o vídeos) y pueden obtenerse a partir de diversas fuentes, como mediciones, registros o transacciones. Estos por sí solos carecen de significado, pero cuando se procesan, estructuran y analizan dentro de un contexto pueden transformarse en información útil.

Tomando como ejemplo una vivienda actual, sus distintos elementos y dispositivos generan una gran cantidad de datos, tales como consumos de agua, electricidad y gas, así como el funcionamiento de las instalaciones de aire acondicionado y calefacción y de determinados electrodomésticos (frigorífico, lavadora, TV, etc.) o de sistemas de seguridad y alarmas. Una adecuada organización de esta información puede facilitar la toma de decisiones para optimizar el consumo y reducir el gasto mensual.

La importancia de los datos radica en que son la materia prima para el análisis, la toma de decisiones y la generación de conocimiento. A medida que la tecnología y la digitalización avanzan, la cantidad de datos disponibles ha crecido exponencialmente, lo que ha llevado al surgimiento del término *big data* (macrodatos) y a nuevos roles de trabajo, como *científico de datos*, *ingeniero de datos* o *analista de datos*. El análisis de grandes volúmenes de datos permite descubrir patrones, tendencias y correlaciones que pueden aprovecharse en diferentes campos, como la ciencia, la investigación, los negocios, la medicina y muchas otras áreas.

2. Aspectos clave del Big Data

Las 5 V del Big Data son un conjunto de características clave que describen los retos y oportunidades del manejo de grandes volúmenes de datos. Estas son:

1. **Volumen**
   Se refiere a la cantidad masiva de datos que se generan constantemente desde múltiples fuentes, como redes

sociales, dispositivos IoT, transacciones, sensores, etc. El Big Data implica manejar terabytes o incluso petabytes de datos.

2. **Velocidad**
   Describe la rapidez con la que se generan, procesan y analizan los datos. Hoy en día, muchas aplicaciones requieren procesamiento en tiempo real o casi en tiempo real (como los sistemas de recomendación o el monitoreo financiero).
3. **Variedad**
   Indica la diversidad de tipos y formatos de datos. Los datos pueden ser estructurados (bases de datos), semi-estructurados (JSON, XML) o no estructurados (imágenes, videos, texto libre).
4. **Veracidad**
   Hace referencia a la calidad y fiabilidad de los datos. No todos los datos son precisos o completos, por lo que es fundamental validar y limpiar los datos antes de analizarlos.
5. **Valor**
   Se refiere a la capacidad de extraer información útil y relevante de los datos. No sirve de mucho tener grandes volúmenes de datos si no se pueden convertir en conocimiento que aporte valor al negocio o a la toma de decisiones.

3. Tipos de datos, estructura y representación

Para obtener, categorizar y analizar debidamente los datos en un contexto determinado, es fundamental conocer los diferentes tipos de datos que pueden existir así como su estructura. De esta forma, se puede trabajar en un modelo que permita explorar e interpretar estos datos de una forma óptima.

Los tipos de datos pueden clasificarse según su estructura en tres categorías principales: estructurados, semiestructurados y no estructurados. A continuación, se define cada categoría:

- **Datos estructurados**. Los datos estructurados están organizados en una estructura predefinida y siguen un formato consistente. Suelen almacenarse en bases de datos relacionales y se representan mediante tablas con filas y columnas (similar a una hoja de cálculo de Excel). Cada columna tiene un tipo de datos específico y se espera que los valores se ajusten a esa estructura. Los datos estructurados se pueden consultar, analizar y procesar fácilmente utilizando consultas y algoritmos diseñados para ese formato.

Por ejemplo, una tabla de clientes suscritos a un servicio con columnas como ID de cliente, nombre, apellidos, correo electrónico, teléfono, fecha de alta e importe de suscripción.

Tabla 1. Clientes suscritos a un servicio

id_client	firstname	lastname	email	phone	subscription_date	amount
00001	Luisa	Martin	lmartin@correo.com	343555222	5/2/2023	30
00002	Alberto	Díaz	adiaz@correo.com	442123444	6/4/2023	50

En este ejemplo, cada fila de la tabla representa un cliente individual. Los datos están organizados en columnas (ID de cliente, nombre, apellidos, etc.). Cada columna tiene un tipo de dato específico, como cadenas de caracteres (nombre, apellidos y correo electrónico), fechas (fecha de compra) y valor numérico (importe suscripción). Estos datos son fáciles de organizar, clasificar y analizar utilizando bases de datos relacionales, herramientas de generación de informes, etc.

- **Datos semiestructurados**. Los datos semiestructurados tienen cierta estructura, pero no siguen un esquema estricto como los datos estructurados. Pueden estar organizados en formatos como JSON (JavaScript Object Notation, en castellano, 'notación de objetos JavaScript') o (eXtensible Markup Language, en castellano, 'lenguaje de marcado extensible'), donde se utiliza una jerarquía y etiquetas para representar la información. Aunque hay cierta estructura, los valores pueden ser opcionales o pueden variar entre diferentes elementos o documentos.

Siguiendo el ejemplo anterior, se muestra un archivo JSON que contiene los datos de un cliente suscrito:

```
{
        “clientes”: [
                {
                “Id_cliente”: “00002”,
                “nombre”: “Alberto”,
                “apellidos”: “Díaz”,
                “email”: “adiaz@correo.com”,
                “fecha_alta”: "6/4/2023”,
                 “importe_subscripcion”: “50”
                }
        ]
}
```

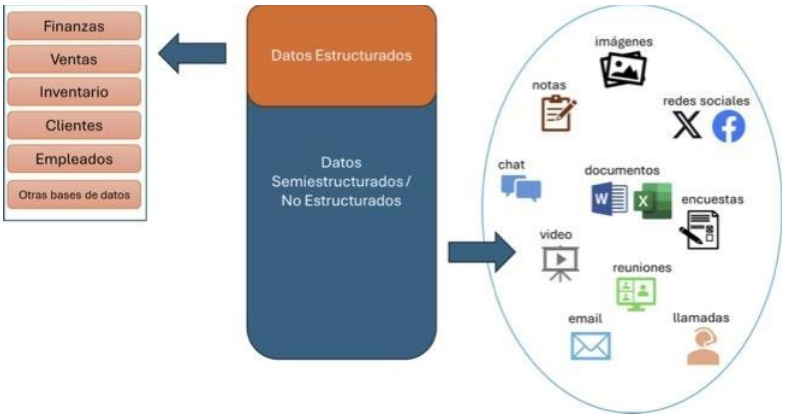
En este ejemplo, se utiliza una estructura de datos llamada *clientes* que contiene una colección de objetos individuales representando a cada cliente. Cada objeto tiene propiedades como ID de cliente, nombre, apellidos, teléfono, etc. Según el cliente, pueden almacenarse más o menos propiedades.

El formato JSON, aunque no tiene una estructura rígida como una tabla de base de datos, permite representar datos semiestructurados de manera legible y fácilmente manipulable, puesto que los valores van acompañados de su correspondiente tipo de atributo.

- **Datos no estructurados**. Los datos no estructurados no tienen una estructura predefinida y no siguen un formato específico. Pueden ser datos en forma de texto libre, imágenes, vídeos, grabaciones de voz, correos electrónicos o publicaciones de redes sociales, entre otros. Estos datos no están organizados en una forma tabular o jerárquica y no se puede realizar una clasificación fácilmente. Requieren técnicas de procesamiento especializadas, como el procesamiento del lenguaje natural o el análisis de imágenes, para extraer información significativa.

Se estima que alrededor del 20 % de los datos que genera y manipula una empresa son estructurados, mientras que el 80 % de los datos restantes son no estructurados.

Figura 3. Ejemplo de relación de datos estructurados y no estructurados generados por una organización empresarial



Adicionalmente, existen otros tipos de clasificación que pueden darse según el contexto y la disciplina en la que se esté trabajando. A continuación, se indican algunos ejemplos:

- **Datos sensibles y no sensibles**. Los datos sensibles son los que pueden requerir protección especial debido a su naturaleza confidencial o su potencial para causar daño a una persona u organización si se divulgan o se utilizan de manera inapropiada. Ejemplos de datos sensibles incluyen información médica, datos financieros u orientación sexual, entre otros.
- **Datos personales y no personales**. Los datos personales se refieren a la información que identifica o puede identificar a una persona específica, como el nombre, la dirección, el número de identificación, etc. Los datos no personales son los que no están directamente relacionados con una persona identificable (como, por ejemplo, la fecha de alta de un cliente).
- **Datos históricos y en tiempo real**. Los datos históricos son los que se han recopilado y almacenado en un momento anterior, y proporcionan información sobre eventos pasados. Los datos en tiempo real son los que se generan y transmiten instantáneamente a medida que ocurren, y permiten la toma de decisiones en tiempo real.
- **Datos transaccionales y no transaccionales**. Los datos transaccionales (también conocidos como datos operativos) son los referentes a transacciones y operaciones comerciales, como compras, ventas, reservas o suscripciones, entre otros. Los datos no transaccionales son los que representan información que no está directamente relacionada con las transacciones a tiempo real, pero que proporcionan información de referencia o analítica, como datos descriptivos de un cliente o ventas totales del último periodo fiscal.

4. Ciclo de vida del dato: del dato al conocimiento

Un determinado dato puede pasar por diferentes etapas desde su creación o captura hasta su obsolescencia. Aunque las etapas específicas pueden variar según el contexto, generalmente se definen las siguientes fases en el ciclo de vida del dato:

1. **Creación**. En esta etapa, los datos se generan o recopilan por primera vez. Pueden provenir de diversas fuentes, como transacciones, sensores, encuestas, formularios, registros, etc.
2. **Captura y almacenamiento**. Los datos se capturan y se almacenan en sistemas adecuados, como bases de datos relacionales, almacenes de datos, *data lakes* u otras plataformas de almacenamiento. Aquí se realiza el proceso de guardar los datos en un formato y estructura apropiados.
3. **Organización y procesamiento**. En esta fase, los datos se organizan, se limpian, se transforman y se procesan para su uso posterior. Se aplican técnicas de limpieza de datos, normalización, agregación y otras operaciones para asegurar su calidad y coherencia.
4. **Análisis y explotación**. En esta etapa, los datos se ponen a disposición de los usuarios y sistemas que los necesiten a través de informes, cuadros de mando, interfaces de programación de aplicaciones (API) y exportaciones de datos, entre otros métodos. Una vez disponibles, se utilizan diferentes técnicas para extraer información y conocimientos relevantes o descubrir patrones, tendencias y relaciones, mediante el uso de consultas directas, minería de datos, aprendizaje automático, estadísticas y visualización de datos, entre otros.
5. **Retención y archivado**. En esta fase, los datos se retienen y archivan de acuerdo con las políticas de retención de la organización y los requisitos legales. Algunos datos pueden almacenarse a largo plazo para cumplir con regulaciones o para futuros análisis históricos.
6. **Eliminación**. Los datos que ya no son necesarios o relevantes se eliminan de forma segura según las políticas de retención y privacidad de la organización. Esto implica la destrucción o el borrado seguro de los datos para proteger la privacidad y cumplir con las regulaciones.

5. Sistemas de almacenamiento y análisis de datos

Un sistema de almacenamiento de datos es una infraestructura tecnológica diseñada para gestionar y almacenar datos de manera eficiente y organizada. Son sistemas generalmente optimizados para manejar grandes volúmenes de datos. Estos sistemas son esenciales para las organizaciones que necesitan gestionar datos en una variedad de tipos y escalas.

Base de datos SQL y NoSQL

A la hora de almacenar datos, los sistemas de almacenamiento digital pueden clasificarse en dos categorías principales: **SQL** (relacionales) y **NoSQL** (no relacionales).

El término *SQL* es un acrónimo de *Structured Query Language*, el cual es un lenguaje de programación que permite consultar, manipular y cambiar datos en una base de datos relacional.

El término *NoSQL*, que significa *Not Only SQL* o *No Solo SQL*, hace referencia a bases de datos que proporcionan un mecanismo de almacenamiento y consulta diferente a las tablas y relaciones utilizadas en las bases de datos relacionales. Los sistemas NoSQL surgieron para abordar problemas de escalabilidad, flexibilidad y rendimiento que las bases de datos relacionales tradicionales no podían resolver de manera eficiente.

Las diferencias entre las bases de datos SQL y NoSQL radican en su estructura, modelo de datos, escalabilidad y casos de uso. A continuación, se realiza una comparación de las principales diferencias entre ambos tipos de bases de datos:

- **Modelo de datos**. Las bases de datos SQL utilizan un modelo de datos tabular basado en tablas con filas y columnas, estando sus tablas relacionadas entre sí. En comparación con las SQL, las bases de datos NoSQL utilizan diversos modelos de datos, como documentos, gráficos, clave-valor y columnas. No se adhieren a un esquema fijo y pueden adaptarse a estructuras más flexibles.
- **Estructura de datos**. Las bases de datos SQL requieren de un esquema predefinido y rígido que debe respetarse a la hora de introducir y manipular datos. Dependiendo de la complejidad de la base de datos, los cambios en el esquema pueden ser complicados de abordar. Sin embargo, las bases de datos NoSQL no requieren de un esquema fijo, lo que les permite agregar, modificar o eliminar campos fácilmente, sin interrumpir la funcionalidad.
- **Escalabilidad**. Las bases de datos SQL escalan verticalmente, normalmente en un mismo servidor, y pueden requerir un aumento

de hardware para añadir más capacidad de almacenamiento o memoria. Esto puede llegar a ser bastante costoso cuando se manejan grandes volúmenes de datos. En contraposición, las bases de datos NoSQL pueden escalar horizontalmente, de modo que se pueden añadir más servidores para distribuir los datos de forma más eficiente, lo que las hace más apropiadas para infraestructuras basadas en la nube.

- **Consultas**. Las bases de datos SQL están optimizadas para consultas estructuradas y complejas, utilizando el lenguaje SQL. En cambio, las bases de datos NoSQL utilizan una variedad de lenguajes y formatos para interactuar con los datos, por ejemplo lenguajes de consulta basados en JSON, JavaScript o líneas de comandos, y pueden incluir también el propio SQL o variaciones de este. Esta flexibilidad facilita el manejo de datos no estructurados.
- **Casos de uso**. Las bases de datos SQL son adecuadas para aplicaciones que necesitan integridad de datos y relaciones complejas, como sistemas de gestión de relaciones con el cliente (CRM) o sistemas de gestión de contenidos (CMS). Las bases de datos NoSQL son ideales para aplicaciones que gestionan grandes volúmenes de datos no estructurados, como redes sociales, aplicaciones móviles y aplicaciones de internet de las cosas (IoT).

A grandes rasgos, las principales diferencias entre bases de datos SQL y NoSQL pueden sintetizarse tal como muestra en la tabla 2.

Tabla 2. Tabla resumen de las diferencias entre sistemas SQL y NoSQL

	Bases de datos SQL	Bases de datos NoSQL
Modelo de datos	Utiliza un único modelo de datos, el tabular.	Utiliza diversidad de modelos de datos.
Estructura de datos	Con un esquema predefinido.	Sin un esquema predefinido.
Escalabilidad	Vertical.	Horizontal.
Consultas	Lenguaje SQL.	Variedad de lenguajes.
Casos de uso	Aplicaciones que necesitan integridad de datos y relaciones complejas.	Aplicaciones que gestionan grandes volúmenes de datos no estructurados.

En el momento de preparar este módulo, algunos ejemplos de bases de datos SQL y NOSQL del mercado eran los siguientes:

Figura 6. Ejemplos de bases de datos SQL y No SQL existentes en el mercado

6. Almacenes de datos (*data warehouses*)

Un data warehouse (almacén de datos, en español) es una infraestructura de almacenamiento de datos utilizada para almacenar, organizar y analizar grandes volúmenes de datos estructurados, en ocasiones también datos semiestructurados y no estructurados.

Estos datos pueden provenir de de diversas fuentes dentro de una organización, como bases de datos operativas, sistemas de gestión de relaciones con los clientes (CRM), sistemas de gestión de recursos empresariales (ERP) y otras fuentes.

El objetivo principal de un data warehouse es proporcionar un entorno centralizado para la toma de decisiones empresariales al permitir a las organizaciones consolidar datos de diferentes fuentes y analizarlos para obtener perspectivas significativas y oportunidades de negocio.

El concepto de 'data warehouse' está estrechamente ligado a la toma de decisiones basada en datos y al aprovechamiento de información corporativa desde su origen.
Un data warehouse típicamente se complementa con herramientas y tecnologías para la extracción, transformación y carga (ETL) de los datos, así como plataformas para el análisis y la visualización de datos. Utiliza técnicas como la indexación, la partición y la optimización de consultas para garantizar que las consultas analíticas sean rápidas y eficientes, incluso cuando se trabaja con grandes volúmenes de datos.

¿Qué diferencias hay entre un data warehouse y data lake?

Los data lake y los data warehouse se utilizan de forma generalizada para el almacenaje de big data, pero, aunque ambos son almacenes de datos, estos no son términos intercambiables. Un data lake o "lago de datos" es un gran conjunto de datos en bruto, que todavía no tiene una finalidad definida. En cambio, un data warehouse o "almacén de datos" es un depósito de datos que ya están estructurados y filtrados y han sido procesados para un propósito concreto.

Es importante realizar la distinción, ya que los data lake y los data warehosue atienden a diferentes propósitos, por lo que requieren un enfoque diferente para ser optimizados adecuadamente.

¿Cómo funciona un data warehouse?

Un data warehouse suele ser el almacén de datos central de una organización. Después de extraer datos de sus fuentes originales e integrarlos en el data warehouse, estos son procesados, transformados y organizados en vistas y tablas de dimensiones o hechos. El método comúnmente utilizado para este propósito es el proceso **ETL** (Extract, Transform and Load) o, más recientemente, **ELT** (Extract, Load, and Transform).

A continuación, detallamos cómo funciona un data warehouse:

1. **Extracción (Extract)**: Los datos se extraen de diversas fuentes dentro de una organización, como bases de datos operativas, sistemas CRM, ERP y otros sistemas de información. Estos datos pueden ser de diferentes tipos, como datos transaccionales, datos de clientes, datos de ventas, etc.
2. **Transformación (Transform)**: Una vez que los datos se han extraído, pasan por un proceso de transformación. Durante esta etapa, los datos se limpian y se transforman en un formato estándar. Esto implica corregir errores, eliminar duplicados y convertir los datos a un formato consistente. Además, se pueden aplicar ciertas reglas de negocio y cálculos para crear datos agregados y derivados que sean útiles para el análisis.
3. **Carga (Load)**: Los datos transformados se cargan en el data warehouse, donde se almacenan en estructuras optimizadas para el análisis. Los datos suelen organizarse en tablas y columnas para facilitar las consultas. Este proceso puede implicar la creación de índices para mejorar la velocidad de las consultas y la partición de datos para una gestión más eficiente.
4. **Análisis (Analysis)**: Una vez que los datos están en el data warehouse, los usuarios pueden realizar análisis complejos y generar informes. Esto se hace utilizando herramientas de análisis y visualización que acceden a los datos del data warehouse y permiten a los usuarios hacer preguntas complejas y descubrir patrones y tendencias en los datos.
5. **Presentación (Presentation)**: Los resultados del análisis se presentan a los usuarios en forma de informes, dashboards y visualizaciones interactivas. Estos informes ayudan a las organizaciones a tomar decisiones informadas basadas en los datos analizados.

¿Qué es ETL (Extraer, Transformar y Cargar)?

En un proceso ETL, los datos se extraen de sus fuentes de origen y se someten a transformaciones para prepararlos antes de cargarlos en los sistemas de destino.

En un escenario ETL tradicional, los datos no estructurados se extraen y se cargan en un área de preparación o de staging, donde se someten a un proceso de transformación. Durante esta etapa, los datos se organizan, se limpian y se transforman en datos estructurados. Este proceso de transformación garantiza que los datos, ahora estructurados, sean compatibles con el sistema de almacenamiento de datos de destino, generalmente un data warehouse.

¿Qué es ELT (Extraer, Cargar y Transformar)?

En un enfoque de procesamiento de datos conocido como ELT (Extraer, Cargar, Transformar), se extraen datos no estructurados de un sistema fuente y se cargan directamente en un data lake para su posterior transformación. A diferencia del enfoque tradicional ETL (Extraer, Transformar, Cargar), los datos están disponibles de inmediato para los sistemas de inteligencia empresarial sin requerir una preparación previa. Esto permite a los analistas y científicos de datos realizar transformaciones ad-hoc según sea necesario.

El enfoque ELT es especialmente útil para llevar a cabo transformaciones básicas en los datos, como validación o eliminación de duplicados. Estos procesos se actualizan en tiempo real y se aplican a grandes volúmenes de datos en su estado original.

7. El proceso de análisis de datos

Los analistas y usuarios de negocios que trabajan con datos suelen atravesar varias etapas o fases, desde que definen sus objetivos de análisis hasta que finalmente toman decisiones basadas en la visualización y el análisis de los datos.
A continuación, se explican las diferentes fases:

1) Definición de objetivos. El primer paso consiste en establecer el objetivo del análisis que se necesita realizar. El analista deberá responder a preguntas como las siguientes:

    - ¿Qué datos y cálculos necesito visualizar?
    - ¿Cuál es el mejor formato de visualización que me ayude a analizar estos datos?
    - ¿Qué patrón necesito identificar en los datos visualizados?

2) Selección de datos. En esta fase se identifica los datos que son relevantes para el análisis. Por ejemplo, en una base de datos

relacional, esto implica elegir las tablas y columnas de la base de datos que se necesitan para lograr el objetivo del análisis.

3) Extracción, transformación y carga de datos (ETL). Este proceso implica extraer los datos necesarios de diversas fuentes, aplicar transformaciones para prepararlos para el análisis y luego cargarlos en una base de datos destinada a dicho análisis, como un almacén de datos. Las transformaciones pueden abarcar desde la limpieza de datos hasta la combinación de diversas fuentes, el cálculo de nuevas métricas y el ajuste de la estructura para que sea adecuada para el análisis. Normalmente, este proceso está automatizado y se ejecuta de manera recurrente, lo que permite que los analistas cuenten con la información necesaria al realizar el análisis. Sin embargo, en casos de datos recién adquiridos, es esencial llevar a cabo este proceso antes de iniciar el análisis.

4) Análisis de datos. En esta fase se definen las técnicas de análisis adecuadas pudiendo crear modelos si es necesario. Estos modelos pueden ser predictivos, clasificatorios, de agrupamiento y de regresión, entre otros, según el problema en cuestión.

5) Visualización de datos. Esta fase implica la representación gráfica de los datos analizados tanto para comprender de forma intuitiva los datos obtenidos como para detectar información más compleja, como relaciones, tendencias y patrones, que no son fácilmente deducibles a partir de la visualización de los datos en formato tabular. Estas visualizaciones, como gráficos, mapas, diagramas y tablas, permiten una comprensión más rápida y profunda de la información. Esta fase es esencial cuando deben presentarse los resultados de manera clara y convincente a las partes interesadas.

6) Interpretación y toma de decisiones. Una vez visualizados, los resultados obtenidos del análisis se interpretan para responder a las preguntas planteadas y alcanzar los objetivos del análisis. Esto implica extraer conclusiones

significativas y tomar decisiones informadas basadas en la información derivada. Estas decisiones pueden ser estratégicas, operativas o tácticas, según el contexto.

7) Retroalimentación y mejora. Después de tomar decisiones basadas en el análisis, es importante evaluar la efectividad de esas decisiones y retroalimentar los resultados al proceso de análisis para mejoras futuras.