Introducción al Big Data

Ajuntament de Barcelona

Barcelona Activa

Índice

1 INTRODUCCIÓN AL *BIG DATA*

Durante el presente módulo daremos nuestros primeros pasos por el mundo del *big data,* introduciremos conceptos clave para entender qué es, cómo funciona, dónde se despliega, por qué tiene sentido tanto para un cliente o una clienta como para una empresa y cuándo es necesario respecto a los sistemas tradicionales.

Empezaremos entendiendo la importancia del dato, ya que es el precursor de cualquier sistema de adquisición, análisis y explotación. Hasta ahora, la *business intelligence* (BI) trataba de ofrecer soluciones a los problemas corporativos, optimizar los procesos internos y de esta forma ganar ventaja sobre su competencia. Debido a la propia evolución de los sistemas informáticos, el crecimiento del volumen de información que se debe procesar es exponencial. Estos sistemas tradicionales (BI), que hacen uso de sistemas de almacenamiento y herramientas que, por sus características, no están preparadas para gestionar tanta cantidad de datos, han quedado obsoletos para las demandas y necesidades de cualquier organización. Por ese motivo, han nacido otras estrategias como el *big data*, que centra sus esfuerzos en cubrir las necesidades del ahora y del mañana, pues unas de sus principales características son su escalabilidad y tolerancia a fallos.

Por lo tanto, un sistema *big data* debe cubrir las etapas principales de los procesos habituales de BI, es decir, debe abarcar todo el flujo del dato, desde que se genera un registro, hasta que es almacenado en nuestro sistema, limpiado y normalizado para su posterior exploración y representado en un informe o cuadro de mando. Durante este primer módulo, veremos las diferencias entre los sistemas actuales y los tradicionales, repasando las bases de datos y técnicas de análisis de información que utiliza cada uno de ellos.

Estructura del módulo 1:

- ¿Qué es el *big data*?
  Presentaremos y entendemos qué es el *big data*, explicando el flujo del dato y algunos casos de uso prácticos que podemos ver en nuestro día a día.
- De dónde provienen los datos y para qué sirven
  Veremos en detalle los tipos de datos que se recogen en el *big data*, los diferentes orígenes y la forma en la que se utilizan para añadir valor a las organizaciones o empresas.
- Las 4 V del *big data*
  Entenderemos qué son las 4 V del *big data* y por qué son necesarias.
- Conceptos básicos relacionados
  Abordaremos conceptos básicos que tienen relación con el *big data*: bases de datos y minería de datos.
- Otros conceptos relacionados
  Se describirán conceptos relacionados con la analítica de datos y la inteligencia de negocio y cómo se conciben dentro de un sistema de big data.
- Resumen de ideas clave
  Repasaremos los conceptos vistos en el primer módulo del curso.

- Ejercicio

Realizaremos un caso práctico sobre datos reales de empresas, para poder aplicar los conceptos estudiados.

1.1 ¿QUÉ ES EL *BIG DATA*?

¡Hola!

En este vídeo abordaremos qué es el *big data*, intentaremos entender cómo funciona y mostraremos por qué está tan de moda hoy en día. Es una nueva forma de ver el mundo, ¡y nadie está fuera de su alcance!

Lo primero que nos puede venir a la cabeza al leer o escuchar *big data* puede ser algo como "datos grandes" o al menos algo relacionado con manejar muchísima información, con una supercomputadora detrás manejando muchos unos y ceros sobre una pantalla negra (tal como recordaremos en *Matrix*), ¡y no iríamos mal encaminados! A fin de cuentas, *big data* es una forma de manejar muchos datos, todos distintos entre sí y a una gran velocidad.

Pero, ¿cómo funciona esto del *big data*? Podemos distinguir cuatro etapas principales:

1. Adquisición de datos: el *big data* está formado por muchos conjuntos de datos, los cuales provienen, entre otros, de bases de datos, ficheros o servicios en la nube (por ejemplo: una red social, la bolsa o la aplicación de meteorología). Hay que tener en cuenta que se pueden llegar a recoger terabytes, o incluso petabytes de información. Los procedimientos tradicionales de ETL (*Extract, Transform and Load, procesos de gestión del dato de business intelligence*) no son capaces de gestionar múltiples orígenes de datos distintos ni procesar tal cantidad de información de manera eficaz, mientras que los procesos *big data* sí están preparados para llevar a cabo estas tareas de forma rápida y veraz.
2. Transformación: todas estas fuentes de información tienen estructuras de datos diferentes, ya que algunas son ficheros de texto plano sin ninguna jerarquía y otras son datos que siguen un esquema fijo. En cualquier caso, un proceso de *big data* es capaz de procesar toda esta información, revisar su integridad y calidad (por ejemplo, validación de esquemas, normalización y de duplicación) y transformarla para poder ser interpretada de una manera sencilla (agregación y filtrado de datos).
3. Almacenamiento: toda la información adquirida se almacena en bases de datos distribuidas, relacionales o no relacionales, o en sistemas de ficheros clusterizados (que veremos más adelante). Estas soluciones pueden residir en un conjunto de servidores locales *on premise* o, cada vez más común, en la nube, donde proveedores como Google, Amazon o Microsoft nos proporcionan estos sistemas a golpe de clic y en apenas segundos.
4. Exploración: para finalizar, una vez que se tiene la información almacenada y bien organizada, podemos crear nuestros propios cuadros de mando para analizarla en

tiempo real, o bien aplicar algún modelo de aprendizaje automático e inteligencia artificial.

Todo esto está muy bien, pero ¿cómo puede ayudarme a mí o a una empresa a mejorar el día a día? Fácil, el *big data* está presente cuando utilizamos una plataforma *streaming*, como Netflix o Spotify, para recomendarnos nuevas series o canciones en función de nuestros gustos. Está presente en las redes sociales para mostrarnos temas, mensajes o fotos de interés. También en Google Maps para indicarnos cuál es la ruta más rápida para llegar a nuestro destino, o en la app del banco para gestionar nuestros ahorros y categorizar nuestros gastos.

Otros casos de uso muy comunes suelen estar enfocados en mejorar los procesos de negocio en las empresas. Por ejemplo, las compañías de venta *online* utilizan algoritmos de predicción para optimizar el stock en función de las búsquedas en internet, redes sociales, tendencias, meteorología, etc. En el mundo de la logística, se está consiguiendo optimizar las rutas de reparto, así como el seguimiento de las mercancías, gracias a la información de tráfico generada por nuestros dispositivos GPS o la que cede tráfico desde sus servidores.

También podemos apreciar el impacto del *big data* en la mejora del rendimiento deportivo, donde se pueden determinar patrones y estilos de juego de diferentes deportes, de modo que se pueda mejorar la planificación de los entrenamientos en función de la condición física o la estrategia utilizada en un partido.

Y esto es solo el comienzo de lo que nos puede ofrecer esta tecnología. ¡Nos vemos en el siguiente vídeo!

1.2 ORIGEN Y UTILITDAD DE LOS DATOS

En este artículo hablaremos de los tipos de datos que existen en el universo del *big data* y cuáles son sus principales fuentes. Mostraremos ejemplos para que nos vayamos familiarizando poco a poco con formatos, sintaxis y para que vayamos comprendiendo cómo afecta cada caso de uso al negocio y cómo puede mejorar nuestra calidad de vida.

Tipos de datos

En primer lugar, analizaremos los distintos tipos de datos con los que se trabaja. Como primera aproximación, imaginaremos que trabajamos con muchos ficheros independientes y que cada uno de ellos puede comprender uno de los formatos que se describen a continuación, los cuales se categorizan en función de su estructura.

- Estructurados
    - Este tipo de datos tiene su longitud, formato y tamaño bien definidos. Se almacenan en tablas, hojas de cálculo o bases de datos. En la tabla que se muestra a continuación, se pueden apreciar las últimas transacciones bancarias de una

persona particular. La información viene organizada por los atributos “Tipo”, “Origen” y “Concepto”, y cada registro tiene una métrica, correspondiente al “Importe” de la operación:

TIPO	ORIGEN	CONCEPTO	IMPORTE
Pago con tarjeta	Cuenta corriente	Gasolinera	55
Transferencia	Cuenta de ahorro	Balance de gastos	500
Pago con tarjeta	Cuenta corriente	Supermercado	30

- Semiestructurados
  - Los registros de un origen semiestructurado no siguen una definición estándar como los estructurados, es decir, no tienen asignado un formato común, pero sí presentan una organización basada en metadatos. Gracias a ella, se establecen objetos y sus relaciones, que muchas veces están aceptados por convención en formatos conocidos como HTML, JSON o XML. En el ejemplo que se muestra a continuación, se puede observar un documento tipo JSON, en el que se ha creado un objeto “Operación” y sus elementos anidados, los cuales representan transacciones bancarias y sus propiedades:

```
{
  “Operación”: [
  {
    “Tipo”: “Pago tarjeta”,
    “Origen”: “Cuenta corriente”,
    “Concepto”: “Gasolinera”,
    “Importe”: “55€”
  },
  {
    “Tipo”: “Transferencia”,
    “Origen”: “Cuenta ahorro”,
    “Concepto”: “Balance gastos”,
    “Importe”: “500€”
  },
  {
    “Tipo”: “Pago tarjeta”,
    “Origen”: “Cuenta corriente”,
    “Concepto”: “Supermercado”,
    “Importe”: “30€”
  },
}
```

- No estructurados
  - Tal como indica su nombre, los conjuntos de datos que comprenden información no estructurada son aquellos que no siguen un formato específico. Suelen asociarse a documentos de texto, imágenes, vídeos o e-mails entre otros. Siguiendo con los ejemplos anteriores, la tabla que describe las transacciones bancarias podría estar almacenada bien en una hoja de cálculo (.xlsx), en formato de texto avanzado (.docx) o

documentos PDF. Su explotación es más compleja que ficheros de texto plano y requiere de *software* específico, muchas veces bajo licencia, o de algoritmos avanzados, como puede ser el procesamiento de imágenes o ficheros de audio.

Tipos de datos por origen

Los sistemas *big data* han llegado para dar respuesta al gran volumen de información disponible que manejamos, de modo que se pueda procesar de forma rápida y crear un valor que ofrezca soluciones a cualquier tipo de escenario. Hasta ahora, las bases de datos tradicionales (Oracle, MySQL, SQL Server, SAP, etc.) han podido dar una solución parcial al procesamiento de la información, pero, ahora, los sistemas modernos, gracias a las nuevas arquitecturas y capacidades de procesamiento, son capaces de adquirir, transformar y analizar los datos de múltiples orígenes en tiempo real. Estas fuentes pueden provenir de la app de mensajería de nuestro *smartphone*, de los sensores que monitorizan la geolocalización, temperatura, posición, etc. de distintos dispositivos, de correos electrónicos o de *logs* de servidores. Todos estos nuevos focos de información forman un nuevo y complejo universo de datos, que, combinados, multiplican su valor, al ser almacenados de forma conjunta en el mismo lugar.

Podríamos considerar algunos de los ejemplos que se describen a continuación como los orígenes más comunes de los sistemas de *big data*.

- Generados por personas. Son aquellas fuentes que generamos de forma o no consciente, bien sea con el uso de las herramientas cotidianas o con diversos registros sanitarios, bancarios, etc.

  - Mensajería instantánea
  - Notas de voz y grabaciones de audio
  - Correos electrónicos
  - Registros electrónicos (médicos, Seguridad Social, bancarios, sistemas privados, etc.)
  - Documentos electrónicos (DNI, pasaporte, carné de conducir, tarjetas de fidelización, etc.)

- Redes sociales y fuentes web. Hoy en día, gran parte de la población publica imágenes, redacta historias o anécdotas, interactúa con otras personas a través de comentarios o reacciones y, por supuesto, utiliza motores de búsqueda para acceder a cualquier tipo de información. Todo esto se almacena en las famosas *cookies*, en el historial de nuestro navegador y, sobre todo, en los servidores de los portales a los que accedemos. Esta información es la más valiosa para las empresas, pues con ella pueden trazar tendencias de mercado, recomendaciones personalizadas de películas, ropa o productos de interés para generar publicidad o registros de actividad. Por lo tanto, encontramos:

  - Cualquier tipo de red social

- Motores de búsqueda (Google, Bing, Yahoo!, etc.)
- Información sobre clics en vínculos y elementos
- RRSS (fuentes de datos de Twitter, publicaciones en Facebook, redes sociales de blogs, prensa...)
- Contenido web (páginas, imágenes, enlaces, etc.)

- Comunicación entre máquinas (*machine-to-machine*, M2M). También conocido por muchos como IoT (*internet of things*). Se trata de fuentes físicas y automatizadas que utilizan sistemas de radiofrecuencia, protocolos como Bluetooh, ZigBee, WiFi, RDIF o GPS para monitorizar una cierta actividad.

  - Tarjetas de acceso y dispositivos de seguimiento RFID
  - Dispositivos de geolocalización a través de señales GPS
  - Otros sensores (parquímetros, máquinas expendedoras, cajeros, etc.)

- Transacciones. Eventos generados por dispositivos de telecomunicaciones (móviles, señales de radiocomunicación...) o eventos únicos de pagos bancarios o ventas.

  - Registros de comunicaciones (llamadas, mensajería, VoIP, etc.)
  - Registros de facturación (pagos con tarjeta, pago *online*, registro de contador de luz inteligente, etc.)
  - Registros de ventas o pedidos *online* (utilización de comercio electrónico y de otras aplicaciones como Glovo o Wallapop)

- Biométricos. Son aquellos que ayudan a la identificación unívoca de una persona en función de rasgos físicos o de conducta. Empleados principalmente en sistemas de acceso y seguridad.

  - Reconocimiento facial
  - Información genética (ADN)
  - Huellas dactilares
  - Escaneo de retina

1.3 LAS 4V DEL *BIG DATA*

¡Hola de nuevo!

En este vídeo veremos una de las bases del *big data*. Es muy importante tener claro cómo identificar una tecnología *big data* de otra que no lo es. Son conceptos muy básicos pero que se suelen confundir entre ellos. No te preocupes, para eso estamos aquí, ¿no?

Cualquier sistema de *big data* se construye sobre cuatro pilares, las conocidas 4 V del *big data*: volumen, variedad, velocidad y veracidad.

Volumen

Hoy en día, los datos se generan automáticamente por los servidores, las redes de comunicación, las interacciones personales, la monitorización IoT, etc. Esto conlleva un volumen masivo de información que debe ser gestionado por un sistema escalable que sea capaz de almacenar y procesar toda esa cantidad de elementos. Hay que tener en cuenta que se tienen unos 25.000 millones de dispositivos conectados a la red y estamos en un crecimiento exponencial. Por eso, el *big data* no solo debe ser capaz de adquirir estos datos aquí y ahora, sino que debe estar preparado para ir creciendo a medida que así lo hacen sus fuentes.

Pero, ¿en qué se traduce esto? Para que nos hagamos una idea, una arquitectura de *big data* estándar debe poder aceptar flujos de cientos de megabytes o gigabytes al segundo y poder almacenar hasta *petabytes* de información, considerando un *petabyte* (PB) como 1024 *terabytes* (TB). De manera global, se están tratando hasta 40 *zettabytes* de datos, o lo que es lo mismo, 40 millones de *petabytes* o 40.000 millones de *terabytes*, una cifra mareante y que parece la antesala de lo que está por llegar.

Variedad

Aquí podemos apreciar la evolución de la tecnología, pues hace unos años los sistemas tradicionales solo procesaban información relacional almacenada en bases de datos. En cambio, ahora se puede trabajar con todo tipo de ficheros y estructuras. ¿Ejemplos? Pues vídeos, fotos, archivos de sonido, correos electrónicos, ficheros de texto o sistemas de monitorización, entre otros. Esto provoca que haya que realizar una ingeniería compleja para el tratamiento, transformación y homogeneización de cada una de las fuentes, con el fin de que el análisis posterior sea eficaz y sencillo.

Velocidad

Esta característica define la rapidez con la que se procesan los datos con los que tiene que interactuar nuestra herramienta de *big data*. Para ello, nuestro sistema deberá admitir un flujo de datos continuo muy grande y de diversas fuentes en tiempo real. Este es uno de los mayores retos de las empresas y de la ingeniería de datos. El hecho de poder procesar la información en tiempo real permite a las organizaciones entender su información con mayor claridad, poder crear análisis predictivos, generar alarmas del estado de sus servicios y tomar decisiones que aportan soluciones estratégicas claves y muy competitivas.

Veracidad

Cuando hablamos de veracidad nos referimos a la calidad del dato, su disponibilidad, el sesgo, ruido y la posible alteración que haya podido sufrir debido a factores ajenos a nuestro sistema. Es la característica más difícil de controlar y uno de los principales quebraderos de cabeza de los responsables del *big data*. Es necesario saber qué y cómo se almacena, es decir, si la información que viene de origen es útil en su estado natural o si es necesario

aplicarle algún tipo de proceso de limpieza para garantizar que el dato es válido, o bien, completar esta información para poder darle un valor añadido. Este punto es tan importante como los otros y cobra vital importancia a la hora de evaluar y analizar la información para ofrecer las mejores soluciones estratégicas.

En resumen, estos cuatro conceptos son la base sobre la que se construye cualquier arquitectura de *big data* y todos ellos son igual de importantes, pues solo se concibe un sistema de estas características si es capaz de analizar un volumen masivo de datos en tiempo real, de múltiples fuentes simultáneamente y con una calidad mínima reconocida. No siempre se puede alcanzar la mejor opción para los cuatro elementos, pero dependerá de nosotros alcanzar la solución de compromiso que mejor se adapte a los requisitos del proyecto.

¡Hasta pronto!

1.4 CONCEPTOS BÁSICOS RELACIONADOS

¡Hola de nuevo!

Durante los próximos minutos aprenderemos qué tipos de bases de datos comprende el universo del *big data*. También introduciremos otros conceptos como la minería y la arquitectura o modelado de datos. ¿Todo a punto? ¡Empezamos!

Bases de datos

Una base de datos es una herramienta que se encarga de organizar la información. Dispone de utilidades para que se pueda acceder, gestionar y actualizar los datos con facilidad. Pueden estudiarse desde distintas perspectivas, aunque podremos hablar de dos tipos de sistemas principales en base a su mutabilidad:

- Estáticos: son aquellos que comprenden datos inmutables, es decir, no se pueden modificar. Su uso está enfocado principalmente a la consulta de la información sobre eventos pasados, facilitando la generación de informes y la minería de datos. También se conocen como OLAP (*On Line Analytical Processing*). Se organizan en cubos multidimensionales, los cuales precargan la información desde un almacén de datos. Los más conocidos son Cognos (IBM), SAP, Oracle Database OLAP o Microsoft Analysis Services.
- Dinámicos: almacenan registros que sí pueden ser alterados. Son los más utilizadas. También llamadas OLTP (*On Line Transaction Processing*). Basados en bases de datos relacionales tradicionales, las cuales se verán a continuación.

También podemos ver las bases de datos en función de su organización:

- Relacionales. Son las más tradicionales y utilizadas tanto por los sistemas de *business intelligence* como por algunos orígenes de *big data.* Se caracterizan por guardar la información de forma estructurada en tablas, donde previamente se ha indicado qué tipo y longitud tendrán sus campos. Permiten relacionar los elementos de distintas tablas de forma sencilla y rápida. Su lenguaje de consulta es SQL (Structured Query Language).
    - Algunas de estas bases de datos podrían ser las tradicionales Oracle, MySQL, SQL Server o bien otras más modernas alojadas en plataformas en la nube como RDS (Amazon Web Services), Cloud SQL (Google Cloud Platform) o Azure SQL Database.
- No relacionales. También conocidas como NoSQL (Not only SQL). Se caracterizan por no tener un esquema definido para el almacenamiento de la información. Esto permite almacenar cualquier tipo de dato, tanto texto, como archivos de audio o vídeo, correos electrónicos o PDF. A diferencia de las anteriores, están preparadas para trabajar en clúster (conjunto de servidores), de forma que permiten:
    - La escalabilidad del sistema
    - Funcionar en modo distribuido, lo que conlleva a un procesamiento paralelizado más rápido que el tradicional
    - Tolerancia a fallos
    - Reducir costes

Por contra, son algo más complejas que las primeras y, en muchos casos, son menos flexibles a la hora de cruzar elementos de distintos conjuntos de datos. Las bases de datos no relacionales serían MongoDB, DocumentDB, Apache Cassandra, Redis o HBase.

Por último, estudiaremos la visión de bases de datos en función de su contenido. Pueden ser:

- Transaccionales: esta categoría engloba la mayoría de bases de datos tradicionales. Aseguran la integridad del dato, tienen baja latencia y son fiables.
- Documentales: guardan datos semiestructurados, principalmente bajo el estándar JSON o XML. Permiten almacenar registros (documentos) con diferentes campos entre sí, mejorando la flexibilidad de la solución. Muy enfocadas a ser utilizadas como objetos, agilizando el procesamiento de los datos. La base de datos documental más conocida es MongoDB.
- Clave/valor: están diseñadas para consultas rápidas en tiempo real, ya que suelen estar almacenadas en memoria (RAM). Redis, Cassandra o DynamoDB (Amazon Web Services o AWS) son las más conocidas para este propósito.
- Columnares: almacenan datos estructurados en columnas, ideales para analíticas de datos. Ejemplos: HBase y Kudu.
- Gráficas: estructuran su información en nodos y aristas, de modo que pueda ser representada gráficamente de manera sencilla. Neo4J es la más utilizada por la comunidad.

Minería de datos

¿Cómo gestionamos y procesamos toda esa información almacenada para que sea útil y práctica en nuestros sistemas? La minería de datos o *data mining* es el conjunto de técnicas y herramientas que analizan grandes volúmenes de datos. Es muy importante tener claras las diferencias entre *big data* y minerías de datos, podemos entenderlo como un todo.

- El *big data* se encarga de recabar toda la información en las bases de datos comentadas previamente.
- A través del *data mining* se limpian los datos y se explotan, creando modelos analíticos y predictivos para el descubrimiento de patrones y tendencias. Esto hace aumentar la rentabilidad y productividad de la organización.

Bajo estos análisis, la minería de datos serviría, por ejemplo, para detectar fraude, predecir una enfermedad o estimar la congestión de tráfico en una ruta determinada.

¡Nos vemos en el siguiente vídeo!

1.5 OTROS CONCEPTOS RELACIONADOS

Hoy en día, todas las empresas tienen fuentes y bases de datos enormes, pero no todas son capaces de analizar toda la información y dar valor a los datos para mejorar su negocio. Aquellas que no consigan superar el reto están condenadas a tomar malas decisiones, a no tener un conocimiento claro de su posición en el mercado y a no saber predecir qué necesidades deberán cubrir a medio y largo plazo. Por ello, es muy importante que se destinen muchos esfuerzos a realizar tareas de análisis sobre los datos de negocio.

En primer lugar, se debe tener claro **qué tipo de análisis se quiere realizar**, pues, en función de nuestro objetivo, podemos encontrar cuatro perspectivas distintas:

- Análisis prescriptivo: nos ayudará en la toma de decisiones, a realizar acciones concretas sobre cuestiones actuales del negocio, optimizando los recursos de la actual operativa. Son técnicas matemáticas que informan de lo que podría suceder y, de esa forma, sugerir decisiones para mejorar los indicadores de negocio. Este análisis ayudaría a una empresa, por ejemplo, a identificar una oportunidad de mercado y poder realizar una campaña de captación de clientes adecuada.
- Análisis predictivo: nos dará una estimación de qué sucederá. Utiliza técnicas de modelización, aprendizaje automático y minería de datos para analizar datos actuales e históricos y a partir de ellos hacer predicciones de acontecimientos futuros. Por ejemplo, es común que las empresas energéticas o de telecomunicaciones tengan modelos que permitan predecir el uso que tiene cada cliente o clienta de su servicio, de forma que puedan proponerles ofertas que permitan un ahorro en sus facturas.
- Análisis diagnóstico: establecerá el porqué de un hecho. Es el más sencillo, ya que, utilizando las reglas de negocio y técnicas básicas de bases de datos, se pueden

obtener los motivos de ocurrencia de un hecho concreto. Son esenciales para determinar y explicar los resultados de acciones determinadas. Por ejemplo, toda empresa utiliza estos análisis para estudiar por qué se ha producido una disminución o aumento de beneficios en un momento concreto, de manera que le permita tomar acciones futuras para evitar o potenciar esa situación.

- Análisis descriptivo: analizará qué sucedió en una situación pasada. Es un análisis estadístico que evalúa los distintos valores que pueden tomar las variables de nuestro modelo y ayudarnos así a reproducir y trazar el camino a una situación deseada. Los más simples ayudan a cuantificar, por ejemplo, el volumen de ventas de una empresa en función de productos o categorías, mientras que otros más complejos ayudan a determinar la distribución geográfica de una enfermedad en un periodo determinado.

En cualquiera de los casos anteriores se ha de tener en cuenta que los resultados son dinámicos, es decir, situaciones actuales pueden ayudar a entender el porqué de hechos históricos. Esto podría alterar y añadir condicionantes a nuestros modelos para que se vayan ajustando cada vez más en el tiempo, utilizando técnicas de aprendizaje automático.

Una vez decidido qué tipo de investigación queremos realizar sobre los datos, se puede proceder a establecer una serie de pasos que serán de ayuda para almacenar la información de forma ordenada y limpia, lo que ayudará, en gran medida, a su posterior análisis. Por ello, existen las siguientes fases:

- Se determina el objetivo y los KPI de negocio (*Key Performance Indicators*). Estos indicadores permitirán medir el resultado de forma numérica, de manera que se sepa previamente qué métricas va a calcular nuestro análisis. Por ejemplo, se puede saber el comportamiento y preferencias de la clientela en función de cuánto y cuándo compran, o evaluar la competencia, sabiendo quién tiene más ventas y quién retiene más clientela.
- Se adquieren los datos. Traeremos a nuestro sistema las distintas fuentes de las que tienen que alimentarse nuestras bases de datos. Esta información podrá ser estructurada, semiestructurada o no estructurada y cada una de estas fases pasará por un proceso alternativo que normalice su información y la almacene de modo que tanto su escritura como lectura sea rápida.
- Se procesan los datos. Se realizan las primeras tareas para organizar la información almacenada de forma correcta. Se filtra, agrupa y cruza la información para construir un modelo correcto para nuestro negocio.
- Se limpian las tablas, colecciones e índices. Se aplicarán procesos de limpieza que se encarguen de modificar o borrar registros erróneos o corruptos y duplicados.
- Análisis exploratorio de los datos. Estas primeras iteraciones tratan de realizar un análisis descriptivo de los datos, viendo qué atributos son significativos y qué valores toman a lo largo del tiempo, gracias a la creación de histogramas, tendencias y otros gráficos.
- Modelado y algoritmos. Se construyen variables estadísticas (medias, medianas, modas, desviaciones, máximos, mínimos, etc.), regresiones y se aplican algoritmos de aprendizaje automático y predicción.

- Explotación de la información. Una vez que se ha realizado el análisis requerido, se generan informes y cuadros de mando para ofrecer, de forma sencilla e interactiva, el resultado del análisis y los KPI. Para ello, es común que se utilicen herramientas de *reporting* como Power BI, MicroStrategy o QlikView.

Con el correcto seguimiento de las fases anteriores se pueden obtener unos resultados que marquen la diferencia en una compañía, pues es fundamental tanto la toma de decisiones como el debido conocimiento del mercado y la situación actual de la organización. Desafortunadamente, tener buenos datos no garantiza buenos resultados de nuestro análisis, pero unos malos datos sí que nos aseguran conclusiones incorrectas. Por ese motivo, es esencial dedicar recursos a conocer qué y cómo se tiene y a dónde

1.6 IDEAS CLAVE: INTRODUCCIÓN AL *BIG DATA*

Durante este primer módulo del curso,hemos dado nuestros primeros pasos en el mundo del *big data*, hemos aprendido qué es y cómo funciona, ofreciendo distintas perspectivas de cómo es de útil esta nueva tecnología para empresas de perfiles tan distintos como las textiles o plataformas *streaming*.

A su vez, se ha mostrado qué tipos de datos existen en función de su estructura (estructurados, no estructurados o semiestructurados) y origen. De esta forma, podemos hacernos una idea de cuánta información se genera a diario, tanto por personas particulares a través de sus dispositivos móviles, como por otros elementos de uso habitual como un cajero, un pedido *online* o la tarjeta de acceso a la empresa.

Se ha introducido el concepto de los cuatro pilares del *big data*, las 4 V sobre las que está construido cada sistema (volumen, variedad, velocidad y veracidad). Hay que tener claro que una arquitectura *big data* se encarga “únicamente” de procesar mucha información de manera muy rápida, de distintas fuentes simultáneamente y ofreciendo una calidad y consistencia adecuadas.

Se ha ofrecido una visión global sobre las bases de datos utilizadas por estos sistemas, de qué tipos principales existen (estáticas y dinámicas), cómo se clasifican en función de su organización (relacionales y no relacionales) y en función de su contenido (transaccionales, documentales, clave/valor, columnares y gráficas).

Por último, además de introducir qué es la minería de datos y cómo se integra con el *big data*, se han descrito los distintos tipos de análisis de información que se pueden llevar a cabo (prescriptivo, predictivo, diagnóstico y descriptivo), las distintas fases por las que debe pasar un dato en nuestro sistema y por qué es tan importante que un negocio dedique recursos y esfuerzos a realizar estas investigaciones sobre sus datos.

2 INFRAESTRUCTURA PARA EL *BIG DATA*

A lo largo de este segundo módulo del curso podremos profundizar en la evolución que ha tenido el *big data* desde sus orígenes y cómo ha conseguido cubrir una necesidad que venía aumentando en los últimos años. Cada vez generamos más información y necesitamos sistemas escalables que estén preparados, ahora y a largo plazo, para soportar ese crecimiento.

A su vez, estudiaremos qué es un clúster y de qué partes se compone. Es esencial que empecemos a consolidar estos conceptos, pues son la base de todos los diseños y arquitecturas de *big data*. Comprobaremos cómo es un flujo real del dato, es decir, desde que este se genera en su origen, cómo se almacena y transforma en nuestro sistema y finalmente cómo se agrega para ser visualizado en un informe o en una herramienta de *reporting*.

Por último, veremos en detalle cuáles son las formas más comunes de gestionar y almacenar la información en el clúster, de modo que seamos autónomos a la hora de elegir un tipo de base de datos u otro en función de nuestras necesidades. Hoy en día, hay muchas alternativas y tipos de sistemas compatibles, por lo que es importante que, además de tener un conocimiento base sobre las distintas categorías de almacenamiento que existen, podamos encontrar un compromiso entre rendimiento, escalabilidad, disponibilidad y accesibilidad de nuestro diseño.

Esto nos permitirá tener una visión global de por qué se necesita el *big data*, cómo se diseña un sistema de estas características, qué partes tiene y cómo funciona. Es solo el comienzo para entender cómo el *big data* ofrece autonomía, valor, inteligencia y reducción de costes al negocio.

En el módulo 2 abordaremos:

- La tecnología y el *big data*
  Se darán a conocer los comienzos del *big data* y el papel que supone para las tecnologías actuales, que son básicas para comprender las siguientes partes del módulo.
- Características de las plataformas de *big data*
  Se presentarán las principales características de diseño y arquitectura. Se describirá cada una de sus capas, apreciando el flujo de los datos desde que se genera hasta que se explota.
- Infraestructura de bases de datos: clústeres
  De qué elementos forman y en qué fases se puede dividir un clúster.
- Almacenaje y gestión de datos
  Introducción a las diferentes soluciones informáticas para el almacenaje y la gestión de datos y sus principales retos.
- Resumen de ideas clave del módulo 2
  Resumen de los principales conceptos.

2.1 LA TECNOLOGÍA Y EL *BIG DATA*

Para que podamos entender qué es el *big data* y su implicación tecnológica, es necesario hacer un breve repaso de su evolución y lo que ha supuesto para los sistemas actuales, los cuales han sabido solucionar el gran problema del aumento exponencial de la generación de información hasta la fecha.

El término *big data* fue empleado por primera vez por la NASA a finales de la década de 1990. La agencia espacial ya advirtió entonces del gran problema que estaba suponiendo el aumento de los datos para sus sistemas. No fue hasta años más tarde, en 2003, cuando Google publicara su sistema de ficheros GFS (Google FileSystem) y el procesamiento MapReduce, que servirían de punto de partida para la creación de lo que hoy conocemos como *big data* y que comenzaría con el proyecto Hadoop por parte de Yahoo!

Este invento supuso una gran revolución: Google consiguió indexar en su motor de búsqueda la información de cualquier web de Internet gracias a los sistemas de ficheros distribuidos. De este modo, en lugar de procesar mucha información en la misma máquina, se llegó a la conclusión de que era mucho más rápido almacenar y procesar la información en distintos servidores de forma paralela, consiguiendo un resultado mucho más rápido y con unos recursos no demasiado costosos.

En 2006, Yahoo! y la comunidad Open Source tomaron el testigo con el desarrollo del ecosistema Hadoop, el primer gran proyecto encargado de almacenar, procesar y analizar grandes volúmenes de datos. Hadoop es un sistema de código abierto que permite distribuir ficheros de forma sencilla en distintos nodos, para poder ejecutar una programación paralela sobre los mismos. Se constituye principalmente de:

- Un sistema de ficheros distribuidos llamado HDFS (Hadoop FileSystem), permitiendo almacenar ficheros en distintos dispositivos.
- Un *framework* de procesamiento llamado MapReduce, que permite aislar el programador de todas las tareas de programación en paralelo; es decir, es el sistema Hadoop el que se encarga de buscar dónde está cada fichero y cómo debe ser tratado para alcanzar la solución de la forma más rápida.

Desde entonces, los diseños y arquitecturas de *big data* no han parado de evolucionar, adoptando distintas formas en plataformas como Cloudera o Hortonworks, que se encargarían de agrupar herramientas de almacenamiento, procesamiento y análisis de datos para facilitar a las empresas su inclusión en esta nueva tecnología. Tanto fue así que, en 2012, Obama fue el primer candidato a unas elecciones presidenciales en utilizar modelos predictivos con la finalidad de obtener ventaja con un margen suficiente frente a sus rivales y, en 2014 y 2015, el IoT y las Smart Cities conectarían el mundo, ayudando a la monitorización de control de calidad del aire, atención médica, supervisión de tráfico, iluminación, etc.

Muchas personalidades expertas señalan que en 2020 se producirá un aumento estimado del 4300 % en la generación de datos anual, por lo que es esencial tener en cuenta el *big data* y su implicación tecnológica, para poder afrontar el crecimiento del volumen y la cantidad de información.

2.2 CARACTERÍSTICAS DE LAS PLATAFORMAS DEL *BIG DATA*

¡Hola de nuevo!

En este vídeo hablaremos del diseño y la composición de una arquitectura de *big data*. Una arquitectura puede estar formada por cinco componentes o fases principales: adquisición de datos, almacenamiento, procesamiento de datos, visualización y administración.

Veámoslos en detalle:

1. Adquisición de datos

Esta primera fase tratará de conectar los distintos orígenes de datos con nuestro sistema, para así poder almacenar y analizar la información posteriormente. Existen dos métodos principales para la recolección de la información:

- Por un lado, *batch* o por lotes. Es una ejecución periódica de un proceso. Buscará en el origen de datos si existe información reciente desde la última conexión realizada. Es común en fuentes como sistemas de ficheros o bases de datos.
- Por otro lado, *streaming* o en tiempo real. Este tipo de recolección se conecta con el origen y crea un flujo continuo de datos, de modo que la información se genera y almacena al mismo tiempo. Es utilizado, con frecuencia, en sistemas de monitorización, detección de anomalías, análisis web o predicción de tendencias.

2. Almacenamiento

Entrando en la fase de almacenamiento, nos encontramos con que las bases de datos tradicionales no son suficientes para gestionar volúmenes tan grandes de información. Por esta razón, encontramos distintas soluciones para guardar los datos adquiridos. Veámoslas:

- Sistemas de ficheros distribuidos: tratan de almacenar ficheros en diversos servidores. Estos ficheros se particionan y se replican en las distintas máquinas, ofreciendo capacidades de alta disponibilidad y escalabilidad a nuestro sistema. El sistema de ficheros original fue Hadoop, aunque ahora se encuentran muchas alternativas en plataformas en la nube como S3 (AWS), Cloud Storage (Google Cloud) y Azure Storage.
- Bases de datos no relacionales. Permiten almacenar información de forma documental (formato JSON o XML) o en forma de clave o valor. Están capacitadas para manejar

grandes cantidades de información, consumen pocos recursos y su escalamiento es sencillo. Podemos destacar bases de datos como Redis, Cassandra, MongoDB o HBase.

- Bases de datos relacionales. Aunque son poco flexibles, consumen más recursos que las anteriores y tienen capacidades de escalado peor. Son una solución esencial cuando se requieren operaciones transaccionales.

3. Procesamiento de datos

Una vez que tenemos la información correctamente almacenada en nuestro sistema de ficheros o bases de datos, llega el momento de procesar y analizar la información. ¿Cómo llevamos a cabo esta tarea? Esto dependerá del tipo de análisis que queramos llevar a cabo, pero todas las herramientas comparten la característica de procesamiento en paralelo. Entre otros, podemos encontrar los siguientes servicios:

- MapReduce. Es la herramienta de procesamiento original de Hadoop. Se compone de una primera fase para recuperar la información (Map) y otra para ejecutar la operación solicitada (Reduce), utilizando la capacidad de cada una de las máquinas de nuestro sistema.
- Apache Spark. Motor de procesamiento distribuido de código abierto. Es el más utilizado actualmente, proporcionando un rendimiento hasta 100 veces más rápido que MapReduce, al poder trabajar en memoria.
- Apache Storm. Procesa en tiempo real y de forma sencilla grandes cantidades de datos.

4. Visualización

Esta última capa del ciclo del dato permite representar el análisis llevado a cabo por la fase de procesamiento. Para ello, tenemos un gran abanico de posibilidades, entre las que encontramos las siguientes categorías:

- *Notebooks*
- Librerías gráficas de JavaScript
- Herramientas de análisis gráfico, entre las que destacan Kibana y Grafana
- Herramientas propietarias, como Tableau, QlikView, MicroStrategy o Power BI

5. Administración

Por último, encontramos esta capa transversal, que se encarga de gestionar y monitorizar los recursos de nuestro sistema. Esto permite conocer en todo momento el estado de nuestros nodos y los servicios que estamos utilizando en cada una de las capas.

Como podemos observar, una arquitectura *big data* pese a ser compleja, tiene fases bien definidas. Esto nos permitirá poderlas estudiar todas de forma individual con mucha facilidad.

¡Hasta pronto!

2.3 INFRAESTRUCTURAS DE BASES DE DATOS: CLÚSTERES

¡Hola!

En este vídeo explicaremos cómo se distribuye la información a lo largo de los servidores que componen un sistema de *big data*. En primer lugar, necesitamos tener claros tres conceptos básicos:

- Un **clúster** es un grupo de servidores que trabajan de forma conjunta. Cada uno de los servidores proporciona almacenamiento, capacidad de procesamiento y gestión de recursos al sistema.
- Un **nodo** es un único servidor en el clúster. Existen dos tipos: los nodos maestros gestionan la distribución de tareas y los nodos esclavos las ejecutan.
- Un **demonio** es un programa en ejecución en uno de los nodos. Cada uno realiza diferentes funciones en el clúster, como la monitorización de los recursos o la planificación de tareas.

A continuación, explicaremos los tres componentes que forman un clúster: la capa de almacenamiento, la de procesamiento y la de gestión de recursos. Juntos proveen al sistema capacidades de procesamiento distribuido de la información.

Vamos a empezar por **la capa de almacenamiento**. Para la explicación, nos basaremos en cómo funciona HDFS, el sistema de almacenamiento de Hadoop, ya que es el sistema de ficheros distribuidos más usado. HDFS provee de un servicio de almacenamiento redundante para cantidades masivas de datos, sin necesidad de utilizar servidores con recursos de *hardware* excesivos.

Los ficheros se dividen en bloques, por defecto, con un tamaño de 128 MB. Estos se distribuyen en tiempo de carga. Cada bloque se replica en múltiples nodos, permitiendo el acceso a los ficheros, aunque tengamos una caída de alguna de las máquinas. En este caso, el nodo maestro se llama Namenode y almacena la metadata, es decir, sabe dónde está situada la partición de cada fichero. Los nodos esclavos se llaman Datanodes y almacenan las distintas particiones de los ficheros.

Seguimos con **la gestión de recursos.** Para que podamos trabajar con los datos almacenados en clúster de manera paralela, es necesario un servicio que se encargue de gestionar los recursos del procesamiento distribuido de la información. Para ello, tenemos soluciones como YARN (*Yet Another Resource Negotiator*), que es la capa de procesamiento de Hadoop, que está compuesta por un gestor de recursos y un planificador de tareas.

En este caso, el nodo maestro se denomina ResourceManager (RM). Se encarga de organizar los recursos de manera global a lo largo del clúster. Los nodos esclavos se llaman NodeManager (NM). Su función principal es reservar recursos para cada una de las aplicaciones solicitadas por el ResourceManager.

Por último, hablaremos del **procesamiento**. En este caso, tenemos *frameworks* de procesamiento distribuidos como Spark, Storm o Flink, y todos funcionan de modo similar. En el caso de Spark, tenemos un *entry point* denominado "Contexto de Spark" (*Spark context*), que utilizará el gestor de recursos elegido (YARN, por ejemplo) para distribuir la información a lo largo del clúster. Spark trabaja en memoria, lo que permite realizar las operaciones de forma mucho más rápida que otros servicios que trabajen en disco. Además, permite recuperarse en caso de caída de alguno de los nodos.

Una vez analizadas las tres partes de un clúster, veamos un ejemplo práctico:

Tenemos una empresa cuyo sistema intenta gestionar el almacenamiento de los albaranes de compra de sus pedidos. Quiere saber, en tiempo real, el número total de pedidos y los beneficios brutos diarios. Para ello, cuando solicitamos a Hadoop que inserte cada albarán en HDFS, lo que hace es dividir los ficheros en varios bloques y guardarlos en diferentes nodos del clúster, consiguiendo que la información esté distribuida. Por otro lado, el *framework* de procesamiento utilizará cada bloque de forma paralela para realizar los cálculos deseados, obteniendo así un funcionamiento mucho más rápido y eficiente que cualquier sistema tradicional.

Con esto ya podemos ver con claridad las bases de un clúster de big data, de qué partes está compuesto y cómo se gestiona. ¡Hasta el próximo vídeo!

2.4 ALMACENAJE Y GESTIÓN DE DATOS

En este artículo, explicaremos qué formas tenemos de almacenar la información en un clúster de *big data*, la importancia y el reto que supone realizar una buena elección para que tanto el funcionamiento como el rendimiento sean los esperados.

En los últimos 15 años, han ido emergiendo muchos tipos de bases de datos y sistemas de almacenamiento, todos ellos válidos en función de necesidades concretas de nuestro sistema. Por ello, es necesario que sepamos diferenciar cada una de las alternativas que se presentan a continuación:

Sistemas de ficheros distribuidos

- HDFS. Es la elección principal cuando necesitamos almacenar y procesar ficheros en un clúster de forma distribuida. Puede desplegarse tanto on premise como en plataformas de cloud y uno de sus principales atractivos es el conjunto de herramientas que se han construido sobre este sistema de ficheros. Sobre la misma arquitectura, se puede establecer la capa de transformación con operaciones MapReduce, utilizando alguna herramienta de data warehouse como Hive, Impala o HBase o un framework de procesamiento como Apache Spark.
- Pese a que la elección más utilizada puede ser HDFS, en la actualidad, están cobrando mucha importancia los sistemas de almacenamiento en la nube (cloud), los cuales

proporcionan servicios PaaS (Platform as a Service) que, de forma rápida y con una configuración mínima, nos permiten empezar a realizar tareas de adquisición de datos para nuestra aplicación. Cada proveedor dispone de su sistema de ficheros y ha creado un marketplace de servicios compatibles, de modo que podamos trabajar los ficheros almacenados con las herramientas que nos habilitan cada uno de ellos. Como principales alternativas, podemos encontrar el servicio S3 en AWS, Cloud Storage en Google Cloud y Azure Storage.

Bases de datos relacionales

Son útiles a la hora de realizar operaciones transaccionales sobre la información almacenada. Entre las principales alternativas destacan:

- Hive: es una herramienta de *data warehouse* construida sobre HDFS, que proporciona una interfaz al usuario o usuaria para poder lanzar consultas y análisis sobre un esquema relacional. Implementa operaciones de MapReduce sobre el clúster y su objetivo es el de trabajar con grandes *datasets*. Pese a penalizar un poco en el tiempo de ejecución y recursos reservados, es capaz de realizar operaciones muy costosas sobre grandes volúmenes de información.
- Impala: es una herramienta muy similar a Hive, que también funciona sobre HDFS. En este caso, permite a los usuarios y usuarias ejecutar consultas SQL con una latencia muy baja, gracias a la gestión de una metadata interna y de un procesamiento paralelo. Se utiliza para realizar tareas descriptivas y analíticas sobre la información de HDFS.
- Es interesante conocer qué alternativas nos ofrecen las plataformas en la nube. En este caso, destacamos RDS en AWS, Cloud SQL en Google Cloud y Azure SQL Database.

Bases de datos no relacionales

Son la mejor elección, cuando la información almacenada no sigue un esquema fijo y se quieren potenciar características de escalabilidad, replicación y tolerancia a fallos en nuestro sistema. Entre las principales opciones encontramos:

- HBase. Trabaja sobre el sistema de ficheros HDFS. Es una base de datos que prioriza la compresión de la información, es rápida al operar en memoria y con una estructura de clave/valor. Es adecuada para hallar un buen compromiso entre operaciones de lectura y escritura sobre grandes conjuntos de datos con un throughput (cantidad de eventos que se procesan por unidad de tiempo) y latencia reducidos.
- MongoDB. Es una base de datos no relacional independiente, es decir que, a diferencia de Hbase, no funciona sobre ningún sistema de almacenamiento propio como HDFS. Es una elección ideal cuando queremos almacenar información documental no estructurada, fácilmente indexable y con la posibilidad de replicación de los datos, o cuando queremos realizar operaciones de balanceo de carga, almacenamiento de ficheros y realizar transformaciones y agregaciones de los datos almacenados fácilmente.

- ElasticSearch. Es un motor de búsqueda indexado muy potente a la hora de realizar búsquedas escalables de texto, almacenamiento de series temporales o la adquisición de *logs* o diversas colecciones de texto. ElasticSearch es una herramienta embebida dentro de un stack llamado ELK, que proporciona, entre otros, una herramienta de adquisición de datos (Logstash), una capa de almacenamiento (ElasticSearch) y una de visualización (Kibana).
- Neo4j. Es una base de datos de grafos, que destaca por su utilidad a la hora de encontrar relaciones entre las entidades de nuestros datos y extraer, de una forma sencilla y gráfica, el máximo valor a la información almacenada.

Todas estas alternativas nos ayudarán a diseñar la arquitectura de *big data* más adecuada para almacenar y procesar nuestra información.

Recordemos: es esencial saber de antemano cómo vienen estructurados los datos, qué queremos hacer con ellos (transformaciones y análisis) y cómo lo haremos (de forma distribuida, escalable, indexada, mediante claves, etc.). Una vez realizado este pequeño análisis previo, ya es solo cuestión de desplegar la solución que más se adecúe a nuestras necesidades.

2.5 IDEAS CLAVE: INFRAESTRUCTURA PARA EL *BIG DATA*

Veamos qué conceptos hemos aprendido en este segundo módulo.

Hemos repasado la **evolución histórica del *big data*** y cómo ha ido integrándose cada vez más en el día a día. Google puede presumir de ser la compañía en tener la red distribuida más amplia del mundo, además de ser la pionera en introducir los conceptos de arquitecturas distribuidas. Esto supuso un antes y un después que ayuda a las compañías actualmente a almacenar y procesar los datos de una manera rápida, escalable y barata.

Se debe tener en cuenta cómo está estructurada una **arquitectura *big data***. Para ello, es interesante que tengamos claro cuál es el flujo natural del dato. En primer lugar, el origen de datos generará los registros necesarios, bien en formato de texto, bien en base de datos o un sistema de colas, para después ser adquirido y almacenado en nuestro repositorio de datos. Posteriormente, se procesará la información, limpiando registros vacíos, nulos o duplicados, validando formatos y aplicando transformaciones y modelos que se encarguen de ofrecer valor al negocio. El resultado de todo este proceso tratará de dar respuesta a los indicadores de rendimiento o KPI planteados, así como ofrecer una visión del estado actual de la información, tendencias y predicciones.

Vemos que en todo momento están presentes las 4 V del *big data*: volumen, velocidad, variedad y veracidad.

Para poder llevar a cabo todo el **proceso de almacenamiento, transformación y visualización de la información** es necesario darle los medios sobre los que operar. Hemos visto cómo

máquinas individuales ya no pueden soportar el volumen de datos que se genera actualmente, por lo que es necesario trabajar con clúster de servidores. Un clúster no es más que un conjunto de máquinas que trabajan en conjunto para ofrecer una capacidad de reposición y procesamiento mayor, a través de la distribución de la información y trabajos paralelos. Este tipo de arquitecturas están pensadas para ser escalables horizontalmente, es decir, podemos añadir más máquinas a nuestro clúster de forma casi automática. También proporcionan características de alta disponibilidad, pues están preparadas para seguir funcionando, aunque alguno de los nodos del clúster quede fuera de funcionamiento temporalmente.

Por último, hemos repasado qué **posibilidades de almacenamiento de la información** existen en función de su categoría, estructura y funcionalidad. Elegir el sistema de almacenaje es una de las decisiones más importantes se tiene que realizar sobre el diseño de un sistema de *big data*, por lo que es fundamental saber las diferencias entre un sistema de ficheros distribuidos y bases de datos relacionales y no relacionales.

3 FRAMEWORKS

A lo largo de este módulo veremos uno de los tipos de servicios más importantes en *big data*: los *frameworks*. Aprenderemos qué son, cómo están formados, en qué lenguaje se programan, qué funcionalidades desarrollan y bajo qué circunstancias explotan su potencial.

Es importante que nos familiaricemos con este tipo de conceptos y que pongamos el foco en cómo funcionan y cuántos tipos de *frameworks* hay, más que aprender la especialización de uno en concreto. Esto se debe a que hay infinidad de alternativas *open-source* disponibles para llevar a cabo nuestra aplicación y todas están en continua evolución, por lo que es probable que las soluciones ofrecidas hoy sean reemplazadas mañana por otras de más modernas y con mejor rendimiento. Por esta razón, a lo largo del módulo se presentará de forma clara y concisa las particularidades de los *frameworks* en función de tu tipología, con el fin de que podamos entender cuál es su funcionamiento, independientemente de su nombre.

Se hará hincapié en los principales *frameworks* de almacenamiento y procesamiento distribuidos que existen ahora mismo en el mercado: Hadoop, Spark y Storm. Con ellos, seremos capaces de cumplir las bases de las 4 V del *big data* fácilmente, pues cada uno ofrece un conjunto de herramientas muy accesible que nos abstraen de arquitectura y funciones de bajo nivel, pudiendo centrarnos en el núcleo de nuestra aplicación y dar valor a la información de nuestro sistema.

Estructura del módulo:

- ¿Qué es un *framework*?
  Explicación introductoria de qué son y cómo funcionan. Conoceremos qué nos aportan a nuestra aplicación, de qué partes están compuestos y cómo se clasifican.
- Principales *frameworks* utilizados
  Comparación completa entre los *frameworks* más utilizados: Hadoop, Storm y Spark. Aprenderemos a qué escenarios se adapta mejor cada uno de ellos, cuáles son sus fuertes y sus características principales.
- Ejemplos de utilización de *framework*
  Presentación de un caso práctico real de la utilización de Hadoop y Spark. Se mostrará una arquitectura *big data*, de modo que podamos tener claro cómo se digieran los datos de diversas fuentes de datos, cómo se limpian y transforman y cómo se analizan a través de los distintos *frameworks* de procesamiento.
- Ejercicio para repasar lo aprendido en el módulo
  En este ejercicio, se deberá decidir qué *framework* es el más adecuado para cada uno de los casos de uso que se muestran. Se pondrán en práctica los conocimientos adquiridos en el módulo.
- Resumen de ideas clave del módulo
  Se realizará un pequeño resumen de las características principales, la arquitectura y los objetivos de los *frameworks* que se han estudiado en el curso.

3.1 ¿QUÉ ES UN *FRAMEWORK*?

¡Hola de nuevo! En este vídeo explicaremos qué es un *framework*, por qué tenemos que utilizarlo y algunos ejemplos que nos serán útiles de cara a plantear nuestros futuros desarrollos. ¡Empezamos!

¿Qué es un *framework*? Un *framework* no es más que un marco de trabajo, es decir, un conjunto de herramientas, convenciones, estándares y buenas prácticas que nos harán la vida más fácil a la hora de crear nuestra aplicación. En general, proporcionan funcionalidades complejas que evitan que dediquemos mucho tiempo a implementar tareas repetitivas o de bajo nivel. De esta forma, podemos focalizar nuestros esfuerzos en aportar valor a la aplicación.

Por lo tanto, podemos decir que un *framework* nos ayudará a cumplir los siguientes objetivos:

- Construcción del diseño de la estructura básica para el *software* desarrollado. Ofrece una serie de funciones, clases y objetos que serán utilizados como un patrón de diseño por el programador o programadora.
- Funcionalidades básicas. El *framework* es el que ofrece todas las funcionalidades de bajo nivel. De esta manera, no tenemos que reinventar la rueda y crear de nuevo todas las singularidades fundamentales. Así evitaremos crear una y otra vez conexiones con una base de datos, la paginación de nuestro sitio web o escribir funciones de procesamiento de texto.

- Reutilización. Una de las características principales de un *framework* es facilitar la creación de bloques de código que permitan ser utilizados en distintos puntos de nuestra aplicación de manera común. Se debe evitar, en la medida de lo posible, escribir la misma funcionalidad más de una vez a lo largo de nuestro programa.
- Aumento de productividad. Nuestra principal dedicación será la de crear propiamente la aplicación. No tendremos que preocuparnos ni de la arquitectura ni de cómo interactúan y se distribuyen los datos en nuestro servidor o clúster. Esto permite, a su vez, que podamos migrar entre *frameworks* fácilmente.
- Favorecer el trabajo en equipo. El hecho de tener una estructura común sobre la que trabajar permite que el código implementado sea más legible. Esto dará la oportunidad a otras personas de entender el funcionamiento del desarrollo, de manera que puedan extender su funcionalidad o corregir algún defecto de su operativa.
- Buenas prácticas. Un *framework* ofrece unos estándares sobre los que tendremos que desplegar nuestro desarrollo. En la mayoría de casos, un buen seguimiento de las buenas prácticas marcadas por la comunidad permite mejorar considerablemente el rendimiento de nuestra aplicación, así como su reutilización e interpretación.

En cuanto a la composición y arquitectura de un *framework*, podemos decir que está compuesto de tres capas distintas, que son:

- Infraestructura. Define tareas de comunicación de red, computación y almacenamiento. Asegura, de manera transparente, a la persona usuaria que datos de diferentes formatos puedan ser almacenados y transferidos de forma eficiente, segura y escalable a nuestro sistema. Comparte objetivo principal con las V del *big data*. Esta capa provee propiedades para gestionar cantidades masivas de información de forma escalable junto al crecimiento de la organización. Intenta optimizar, en la medida de lo posible, el IOPS (operaciones de entrada y salida por segundo), para asegurar que la tasa de transferencia y procesamiento sea la adecuada a nuestras necesidades.
- Plataforma. Es la colección de funciones que facilitan procesar los datos con un buen rendimiento. La plataforma incluye capacidades para integrar, gestionar y aplicar tareas de procesamiento de la información. En entornos de *big data*, esto significa que la plataforma necesita facilitar y gestionar soluciones para el procesamiento y almacenamiento distribuido.
- Procesamiento. Esta capa se encarga de ofrecer funcionalidades de acceso a la información. Las tareas ejecutadas operan sobre los *datasets* almacenados de forma distribuida. Aquí es donde residen las labores de limpieza, transformación, análisis de los datos y ejecución de los algoritmos. Este procesamiento ofrecerá los resultados requeridos y que aportan el valor al negocio.

Por último, clasificaremos los *frameworks* de procesamiento en base a su tipología:

- *Frameworks* de procesamiento batch. Son aquellos que realizan tareas planificadas para el tratamiento de la información. Aquí destacan las operaciones MapReduce de Apache Hadoop.

- *Frameworks* de procesamiento streaming. Marcos de trabajo que gestionan operaciones del dato con flujos en tiempo real, de manera que la información es procesada en cuanto entra en nuestro sistema. Los principales frameworks de este tipo son Apache Storm y Apache Samza.
- *Frameworks* híbridos. Estos últimos destacan por funcionar tanto en batch como en streaming. Apache Spark y Apache Flink son los más apropiados para este tipo de escenarios.

Estos serían los puntos básicos que tendremos en cuenta a la hora de caracterizar y elegir un *framework* de *big data*. Los seguiremos estudiando en los próximos apartados del curso. ¡Hasta pronto!

3.2 PRINCIPALES *FRAMEWORKS* UTILIZADOS

En este artículo se describirán los tres principales *frameworks* de procesamiento de información que existen actualmente: Hadoop, Apache Spark y Apache Storm.

Hadoop

Hadoop es un *framework open source* de procesamiento y almacenamiento distribuido, utilizado en aplicaciones de *big data* y ejecutado sobre sistemas clusterizados. En la mayoría de casos es el centro de ecosistemas encargados de ofrecer analítica avanzada y predictiva, *data mining* y aplicaciones de *machine learning*.

Está formado principalmente por cuatro componentes:

- HDFS (*Hadoop Distributed File System*): sistema de ficheros que gestiona el almacenamiento y el acceso distribuido a la información sobre varios nodos de un clúster.
- YARN (*Yet Another Resource Manager*): es el gestor de recursos de un clúster de Hadoop, responsable de asignar los recursos del sistema a las distintas aplicaciones y tareas ejecutadas.
- MapReduce: es el *framework* de procesamiento usado en aplicaciones *batch* para mover grandes volúmenes de información en sistemas Hadoop.
- Herramientas Hadoop: es el conjunto de utilidades y librerías que proporcionan las capacidades necesarias para dar soporte e interconectar todos los servicios del ecosistema Hadoop.

El funcionamiento de Hadoop se basa en dos componentes principales: el primero es el **sistema de ficheros** (HDFS), que se encarga de dividir los datos en diferentes nodos, replicarlos para ofrecer alta disponibilidad a la aplicación y gestionar la información y el estado del clúster; el segundo componente, **MapReduce**, procesa los datos en cada uno de los nodos paralelamente y calcula el resultado de cada tarea.

Hadoop es importante porque:

- Puede almacenar y procesar grandes cantidades de datos estructurados y no estructurados rápidamente.
- El procesamiento está protegido ante caídas del sistema. De esta forma, si un nodo queda fuera de servicio, la tarea es redirigida automáticamente a otros nodos disponibles para que la computación distribuida no falle.
- Los datos no necesitan ser preprocesados una vez son almacenados. Las organizaciones pueden almacenar cuanta información deseen, incluyendo datos no estructurados (texto, vídeo, imágenes, etc.) y decidir luego qué hacer con ellos.
- Es escalable horizontalmente. En caso de necesidad, se pueden añadir más nodos al clúster para almacenar o procesar más información.

Apache Spark

Spark es un motor de procesamiento distribuido de propósito general. Destaca por su versatilidad, pues los cuatro módulos por los que está compuesto permiten su compatibilidad con muchos escenarios de analítica avanzada, tanto en *batch*, como en *streaming*, para aplicar algoritmos y modelos de predicción o representación de grafos. Está optimizado para trabajar en memoria y puede alcanzar una velocidad de procesamiento hasta 100 veces mayor que con MapReduce, pudiendo manipular *petabytes* de datos al mismo tiempo. Soporta los lenguajes de programación Java, Scala, Python y R.

Los casos de usos más típicos de Apache Spark son:

- Procesamiento *streaming*: procesamiento de *logs*, datos de sensores y, en general, cualquier *stream* de datos como el *clickstreaming* (proveniente de fuentes web), redes sociales, monitorización de sistemas, transacciones financieras, etc. Los datos provienen de forma continua en un *stream* y se almacenan y procesan en el mismo instante en el que entran en el sistema. Este tipo de procesamiento es muy útil para análisis de sentimiento, tratamiento de telemetría en medios de transporte y logística o sistemas de recomendación en aplicación de música y *streaming* de vídeo.
- *Machine learning*: la capacidad de Spark para trabajar con los datos en memoria y ejecutar consultas de forma recursiva y escalable, convierte este *framework* en una excelente opción para ejecutar algoritmos de aprendizaje automático. De esta forma, se pueden ofrecer respuestas de tendencias de mercado o comportamiento, predicción de eventos o detección de fraude.
- Analítica interactiva: proporciona mucha flexibilidad a la hora de ejecutar consultas a base de datos de forma rápida, sin necesidad de que estén preestablecidas por el sistema de forma estática. Spark ayuda a refrescar la información de un cuadro de mando dinámico o a llevar a cabo tareas de *data discovery* para dar respuestas a preguntas de negocio.
- Integración del dato: Spark es una pieza fundamental a la hora de realizar tareas de consistencia de la información, reduciendo considerablemente los costes y tiempo de procesamiento de los procesos corporativos. Es muy utilizado a la hora de crear

procesos ETL para extraer distintos orígenes de datos y completar *jobs* de limpieza, normalización y carga de resultados en el sistema de destino.

Apache Storm

Es un sistema de computación en tiempo real, *open source*, tolerante a fallos y distribuido. A diferencia de Apache Spark, está muy enfocado a procesamiento de *streams* y eventos en tiempo real. Incluye su propio gestor de recursos, mientras que Spark necesita la utilización de YARN o Mesos para la orquestación de tareas.

Dado que comparte muchas singularidades con Spark, es importante tener claras sus diferencias y a qué está enfocado cada *framework* para poder elegir uno u otro correctamente en función de nuestras necesidades:

- Procesamiento *streaming*: Storm ofrece mejor rendimiento al utilizar una metodología *micro-batch*, es decir, comprueba el flujo de entrada con más frecuencia que Spark.
- Lenguajes de programación: Storm soporta más lenguajes que Spark.
- Latencia: Storm proporciona mejor latencia con menos restricciones.
- Coste de desarrollo: Storm no soporta que el mismo código sea utilizado para procesamiento *batch* y de tiempo real, mientras que Spark sí que ofrece esa posibilidad. Es muy importante en entornos multiprocesamiento.
- *Throughput*: Spark puede procesar hasta 100 k de registros por segundo, hasta 10 veces más que Storm.

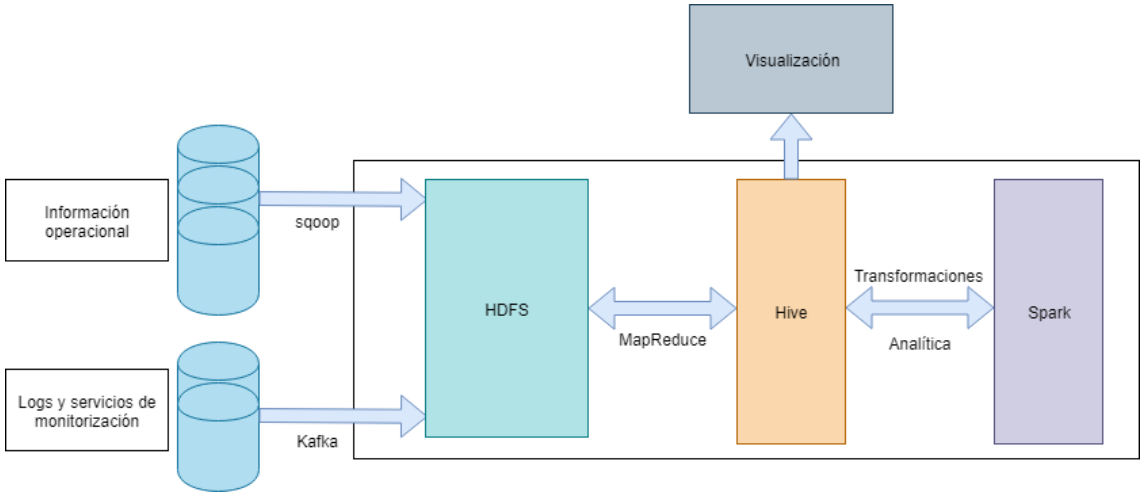
Como se puede apreciar en el análisis del artículo, todos los *frameworks* ofrecen un potencial enorme y todos tienen sus particularidades, por lo que es muy importante saber qué tipo de almacenamiento y procesamiento precisa nuestra organización para poder elegir el motor de procesamiento que más se ajuste a nuestras necesidades.

3.3 EJEMPLO DE UTILIZACIÓN DE *FRAMEWORKS*

En este artículo veremos un caso práctico de la utilización de Hadoop. Para ello, representaremos un *data warehouse* corporativo (EDW, *enterprise data warehouse*), es decir, un repositorio unificado para todos los datos que recogen los diversos procesos de la empresa. En la siguiente imagen se puede apreciar el esquema lógico de la arquitectura, que está dividido en varias capas:

- Orígenes de datos: se tienen fuentes de datos de carácter operacional (facturas, albaranes de compra, inventariado de productos, etc.) y otros que provienen de los sistemas informáticos (*logs* de los servidores o monitorización de sensores). En cuanto a la tipología, vamos a tener fuentes correspondientes a bases de datos, ficheros de texto o colas de Kafka (servicio big data que provee la arquitectura de un sistema de colas para el procesamiento de streams en tiempo real).

- **Almacenamiento:** se dispondrá de un clúster distribuido de HDFS, que se encargará de almacenar todos los ficheros de los orígenes de datos.
- **Procesamiento:** esta capa se encargará de ejecutar operaciones de MapReduce para limpiar, transformar y analizar la información almacenada.
- **Visualización:** se utilizarán diversas herramientas de visualización (MicroStrategy, QlikView o Power BI) que permiten representar cuadros de mando y exportar informes en formato de texto u hojas de cálculo.



Dentro del flujo del dato, destacaremos las siguientes etapas:

Etapa de adquisición de datos

Para que podamos traernos la información más reciente de los orígenes de los datos, se utilizarán tres servicios:

- Sqoop: es una interfaz que permite transferir datos de manera automática entre bases de datos relacionales y HDFS. En este caso, podremos obtener toda la información operacional de forma incremental (la más reciente) a nuestro sistema de ficheros de Hadoop.
- Flume: servicio distribuido encargado de recolectar, agregar y mover grandes cantidades de datos de tipo *log*. Tiene una arquitectura muy sencilla, utiliza pocos recursos, es robusto y está enfocado a flujos de datos en *streaming*. Nos servirá de mucha ayuda a la hora de traernos a HDFS los orígenes de datos de tipo *log*.
- Kafka: es un servicio de procesamiento en *streaming*. A través de la gestión de colas es capaz de proporcionar flujos de datos en tiempo real con un *throughput* alto y baja latencia. Será la base para obtener la información procedente de monitorización de sensores e insertarla en nuestro sistema de ficheros distribuidos.

Etapa de almacenamiento

Como núcleo del *data warehouse*, HDFS es la piedra angular que provee a todos los servicios de la persistencia necesaria para registrar la información corporativa, estado de los servicios en ejecución y resultados de los análisis descriptivos y de predicción.

La información vendrá organizada en las siguientes capas:

- Landing zone: almacena los ficheros nuevos y datos crudos, es decir, sin ningún tipo de tratamiento.
- Cleansing zone: contiene la información una vez que se le han aplicado procesos de limpieza, validación y eliminación de duplicados.
- Transformed zone: registra el modelo de datos tratado, que contiene la estructura final de cada una de las tablas de detalle y agregadas, que servirán de origen a los procesos analíticos y a los *frameworks* de procesamiento. Los resultados analíticos también pueden ser almacenados en esta capa.

Etapa de procesamiento

Para llevar a cabo tanto la traslación y transformación de información entre las capas de nuestro *data warehouse*, como la posterior analítica, se van a utilizar estas dos herramientas:

- Tareas MapReduce: utilizaremos Hive como interfaz SQL para llevarnos los datos de la capa de *landing zone* a la de *cleansing zone*. Hive es un servicio construido sobre HDFS que realiza operaciones de MapReduce para llevar a cabo las consultas programadas. Además, ofrece un modelo relacional de base de datos, que puede ser utilizado por un *framework* de procesamiento de datos como Spark o Storm, o por herramientas de visualización de datos para su posterior explotación.
- *Jobs* de Spark: el clúster de Spark ejecutará todas las tareas de transformación del dato, completando el modelo relacional con nuevas tablas, agregando y cruzando la información de los distintos orígenes de los datos. Por otro lado, también se encargará, a través de su módulo de *machine learning*, de realizar modelos predictivos y analíticos que muestren los resultados buscados.

Etapa de visualización

Por último y no menos importante, se necesita representar toda la información adquirida en nuestro sistema y los resultados de los análisis en varios cuadros de mando. Diferenciaremos dos tipos:

- Dashboards en tiempo real: muy útiles a la hora de mostrar el estado actual de los logs de servidores y la monitorización de sensores. Se podrá apreciar a simple vista, si alguno de los valores supervisados supera un umbral preestablecido, de modo que se puedan prevenir averías o posibles sobrecargas en alguno de los nodos del clúster. También son esenciales a la hora de controlar la gestión y dimensionamiento de recursos asignados a nuestras máquinas.

- Dashboards bajo demanda: estos cuadros de mando se actualizarán de forma periódica o manual, cuando se requiera saber el estado de las KPI de nuestro negocio. Estas visualizaciones representarán el resultado de los algoritmos y análisis llevados a cabo por nuestro motor de procesamiento, ofreciendo información de tendencias, predicciones y análisis de la información actual.

Para implementar estas funcionalidades, se dispone de multitud de herramientas. Entre las más conocidas, se pueden encontrar MicroStrategy, QlikView o Power BI.

A través de todas estas capas se completa el flujo del dato. Recordemos que el dato debe ser adquirido y almacenado en nuestro sistema de ficheros, debe ser limpiado y analizado por nuestro *framework* de procesamiento y, finalmente, debe ser representado por nuestra herramienta de *reporting* preferida.

3.4 IDEAS CLAVE: *FRAMEWORKS*

Una vez finalizado el módulo, es importante que repasemos los puntos más importantes.

Hemos definido un ***framework*** como un conjunto de herramientas y estándares que ayudan a la ingeniería de datos a llevar a cabo tareas mucho más rápido y de forma óptima. Implementa metodologías y buenas prácticas para evitar que se cometan errores a la hora de crear bucles recursivos, accesos recurrentes a la información o, en general, un manejo con un mal rendimiento de las estructuras de datos distribuidas.

Uno de los aspectos más importantes de un *framework* es la infraestructura que provee a nuestro sistema. Con una configuración adecuada, el *framework* se encarga de dividir, distribuir y replicar los paquetes en el clúster, despreocupándonos de toda la capa de bajo nivel del sistema. Estos automatismos ahorran muchos costes a las empresas, pues la instalación de un ***framework*** es rápida, el uso es sencillo y el rendimiento es óptimo.

Se ha visto como Hadoop es uno de los *frameworks* más completos, ya que dispone de cuatro capas. La de almacenamiento está basada en HDFS. El gestor de recursos, clave para la orquestación de tareas entre maestros y esclavos, correspondiente a YARN. La fase de procesamiento se asocia con operaciones de MapReduce. Y, por último, se tienen todas las herramientas de apoyo al desarrollo de las aplicaciones y a la interconexión de servicios del ecosistema de Hadoop.

Además de Hadoop, hemos estudiado otros *frameworks* de procesamiento como:

- Spark (procesamiento streaming e híbrido): además de tener un componente muy fuerte en el procesamiento streaming, es mucho más completo que Storm, al disponer de herramientas de analítica avanzada, aplicación de modelos de machine learning y representación de grafos.
- Storm (principalmente para procesamiento en tiempo real): Apache Storm es un framework exclusivo de tratamiento de streams, parcela en la que supera a Spark.

Finalmente, hemos introducido un ejemplo donde se representó una arquitectura completa de un sistema de *big data* clásico, con las capas de adquisición, almacenamiento, procesamiento y visualización de la información.

Antes de cerrar el módulo, se ha de destacar una particularidad de los *frameworks*, su modularidad. Puede que el día de mañana sean otros los que sustituyan a los que se han estudiado en este módulo, pero todos suelen estar preparados para ofrecer una compatibilidad suficiente para que la migración entre tecnologías no sea demasiado costosa.

Es por ello que el conocimiento más importante a tener en cuenta es el de qué necesidades debemos cubrir (sistemas distribuidos, alta disponibilidad, replicación, procesamiento *batch*, tiempo real, analítica avanzada, etc.) y qué tipos de *frameworks* existen en el mercado que puedan adaptarse a nuestros requerimientos.

4 VISUALIZACIÓN DE DATOS: PROGRAMAS Y TECNOLOGÍAS

En este módulo daremos a conocer algunas herramientas de *business intelligence* (BI) que han sido integradas dentro del ecosistema ***big data***. Estas herramientas son útiles al negocio en la toma de decisiones y en la generación de cuadros de mando, que ayudarán a comprender el análisis realizado y la información de negocio utilizada en el plan estratégico de la empresa. Repasaremos servicios como Tableau, QlikView, Power BI, Kibana y Grafana.

A continuación, presentaremos una serie de casos de uso reales de un diseño *big data*. Son escenarios que se aplican en nuestro día a día y que nos ayudan a automatizar tareas, recomendarnos contenidos personalizados, a detectar fraudes o a optimizar los precios. Esto nos ayudará a comprender la importancia de esta tecnología y el impacto que está teniendo en nuestra sociedad.

Por último, revisaremos cuáles son las tendencias de las herramientas de *big data* en los próximos años. Responderemos a preguntas como "¿cuál será la evolución de las plataformas *cloud*?", "¿qué tipo de procesamiento tiene un mayor recorrido?", "¿sobre qué analítica se van a focalizar los esfuerzos?" o "¿qué fase del flujo del dato va a ser la más reforzada?".

Finalizaremos el módulo con un breve test sobre los contenidos que se han visto en el curso.

Estructura del módulo:

- Visualización de datos
  Importancia y papel que juega en *big data* la visualización de datos.

- Metodologías y programas
  Diferentes metodologías: sus usos y ventajas. Los principales programas, las metodologías que utiliza cada uno y sus principales características: Tableau, QlikView, Power BI, Kibana y Grafana.
- Casos de uso de *big data*
  Ejemplos reales del uso de *big data*.
- Tendencias
  Tendencias en el uso de *big data* en los próximos años.
- Resumen de ideas clave del módulo
  Resumen de ideas clave del módulo.
- Test sobre los contenidos del curso
  10 preguntas tipo test sobre los conceptos trabajados en el curso.

4.1 VISUALIZACIÓN DE DATOS

¡Hola!

En esta sesión analizaremos la importancia que tiene la visualización de datos en el mundo del *big data*. Cualquier compañía que se preocupe por la calidad del dato y por la viabilidad de su negocio tiene que recurrir a este tipo de herramientas.

¿Por qué?

Muy sencillo: para las empresas es muy importante saber en todo momento tanto el estado financiero actual de la organización, como la tendencia de mercado en la próxima temporada. Necesitan métricas e ideas disruptivas para mejorar su negocio. Y todo ello con el uso de herramientas muy sencillas y visuales, accesibles a empresas de pequeño y gran tamaño, gracias a sus posibilidades de escalabilidad.

Con estas soluciones de *software*, las empresas ahora pueden dar sentido a complejos conjuntos de datos de *big data* sin demasiados quebraderos de cabeza. Estas soluciones de *business intelligence* pueden recoger, analizar y convertir los datos en informes y cuadros de mando comprensibles para la persona que administra los sistemas, un o una ingeniera del dato o un ejecutivo o ejecutiva. El objetivo no es otro que dar valor a la información almacenada en nuestros sistemas y convertirlo en beneficios para la empresa. ¿Cómo podemos conseguirlo? Dependerá del caso de uso en el que nos encontremos. Por ejemplo:

- Administración de sistemas. Podemos ahorrar coste de máquinas en la compañía si presentamos un informe con el dimensionamiento y uso de los servidores corporativos. Si se demuestra que la capacidad está sobredimensionada, se puede reacondicionar la arquitectura para que se ajuste más a nuestras necesidades y así evitar pagar por recursos que no se utilizan.
- Licenciamiento. Las empresas invierten mucho dinero en el licenciamiento de *software* de ofimática, procesamiento algorítmico, herramientas de *big data*, etc. Haciendo uso

de servicios de visualización de datos es muy fácil apreciar cuál es el uso que se está dando a las licencias adquiridas y si realmente se están amortizando o si es necesaria la adquisición de un mayor número de ellas.

- Gestión de recursos. Podemos tener el control en todo momento y, a simple vista, de la administración del *stock* de productos, de la gestión de la cartera de clientela y plantilla, del registro de proyectos, etc. Los distintos cuadros de mando pueden representar tanto información histórica, como en tiempo real, así como análisis predictivos del estado futuro de los recursos.
- Tendencias de mercado. Gracias al análisis previo que han realizado nuestros modelos analíticos de *big data*, somos capaces de reproducir cuál será el estado del mercado en función de nuestra experiencia pasada y las necesidades actuales de la clientela. Este es uno de los objetivos de cualquier empresa, pues poder anticiparse a la competencia puede suponer un éxito en lo económico, en visibilidad y en captación de talento.

Entonces, ¿por qué utilizar herramientas de *business intelligence* en nuestro negocio? Pues, tal como hemos visto en los ejemplos, los beneficios de este tipo de herramientas superan con creces las inversiones que conllevan. Pueden ayudar a las empresas a obtener información de gran valor que ayude al crecimiento corporativo, a resolver inquietudes de negocio, a recopilar datos de marketing más rápidamente, a proporcionar una vista en tiempo real de la organización y a permitir la anticipación de resultados futuros utilizando análisis predictivos.

Cada vez son más las empresas que utilizan estos servicios para su crecimiento y por ello el mercado de este tipo de soluciones está en plena expansión. De hecho, el mercado global del *software* de BI (*business intelligence*) ofrecerá un crecimiento anual del 7,1 % hasta 2025. Se espera que las ganancias alcancen los 26 billones de dólares en el año 2021, por los 16,5 billones actuales. Esta expansión está relacionada con la evolución tecnológica del *big* data, que se está desarrollando en la última década y que seguirá en progresión en los próximos años.

Las nuevas tendencias de *software* de BI han proporcionado nuevas capacidades a las organizaciones. El descubrimiento de datos, que solía ser el territorio de los expertos y expertas en análisis avanzado, ahora se hace más fácil con estas plataformas. Esto se logra a través del análisis visual, lo que permite a las personas responsables en la toma de decisiones acceder y actuar de inmediato sobre los datos. Quizás una de las tendencias más importantes en las soluciones de BI es su provisión de soporte móvil, su compatibilidad multiplataforma y su implementación en la nube, lo que permite a las personas usuarias acceder y analizar información desde cualquier dispositivo.

Creemos que sobran los motivos para que las empresas se adhieran a esta tecnología: es el presente y el futuro para el crecimiento de cualquier organización.

¡Hasta luego!

4.2 METODOLOGÍAS Y PROGRAMAS

En este artículo ofreceremos una visión amplia sobre las técnicas de visualización de datos, los factores que afectan a la elección de una gráfica y un análisis de las principales herramientas del mercado.

¿Qué determina la elección de una visualización de datos?

Una visualización es la herramienta para dar sentido al dato. Para presentar la información y sus correlaciones de la forma más sencilla, los y las analistas usan varias técnicas: gráficos, diagramas, mapas, etc. Elegir la mejor técnica y su disposición es el camino para hacer que el dato sea accesible a cualquier perfil. Y también al contrario, una mala representación de la información puede conllevar a no explotar correctamente el dato o a hacerlo irrelevante. Para ello, se presentan cinco factores que influencian a la hora de seleccionar una visualización:

- Público. Es muy importante ajustarse a la audiencia objetivo. Si se busca ofrecer un dato agregado a un cliente o clienta final, quizás, con la visualización sencilla, es más que suficiente. Si, por el contrario, la representación va enfocada a alguien que trabaja en ingeniería del dato, es posible que haya que ofrecer visualizaciones y diagramas con más detalle.
- Contenido. El tipo de dato determina la técnica elegida. Por ejemplo, si se quiere representar una serie temporal, lo ideal es utilizar un gráfico de línea, pero, si queremos comparar los elementos de un atributo concreto, es conveniente utilizar gráficos de barras.
- Contexto. Se deberían utilizar distintas aproximaciones en función del contexto. Según el elemento que se esté estudiando, será importante jugar con la combinación de colores, contrastes y sombras correctos. Esta característica es necesaria para poder mostrar distintas gráficas superpuestas o que compartan la misma representación, permitiendo dibujar más de una métrica de un atributo con claridad.
- Dinamismo. Cada tipo de dato y su nivel de agregación influyen a la hora de elegir la visualización. Por ejemplo, una serie temporal que se actualiza en tiempo real y tiene un detalle de segundo no tendrá el mismo aspecto que una gráfica histórica con información mensual.
- Propósito. La forma en la que están implementadas las distintas visualizaciones también provoca un impacto en el mensaje y objetivo del cuadro de mandos ofrecido. De esta manera, la persona usuaria que necesite saber los KPI de su negocio utilizará un cuadro de mandos sencillo, que debe representar, a simple vista y de forma sencilla, unas métricas representativas del estado de los objetivos de la compañía. Por otro lado, para poder crear un análisis complejo de la información, puede interesar añadir distintos tipos de representaciones con filtros y elementos de control.

Tipos de representación

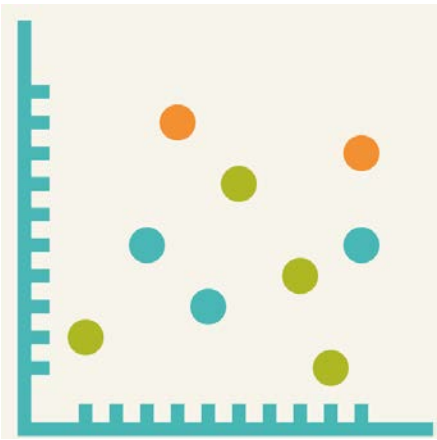
En función de estos cinco factores, se elige entre distintos tipos de visualizaciones posibles. Estas son las más comunes (en el siguiente recurso se puede profundizar sobre la visualización de datos http://atenciociutadana.gencat.cat/web/.content/manuals/guia-visualitzacio-dades.pdf) :

Gráficos

La forma más sencilla de mostrar un conjunto de datos es a través de un gráfico. Pueden variar entre líneas y barras, que muestran una relación entre elementos a lo largo del tiempo y una tarta, que representa los elementos de un atributo de forma proporcional.

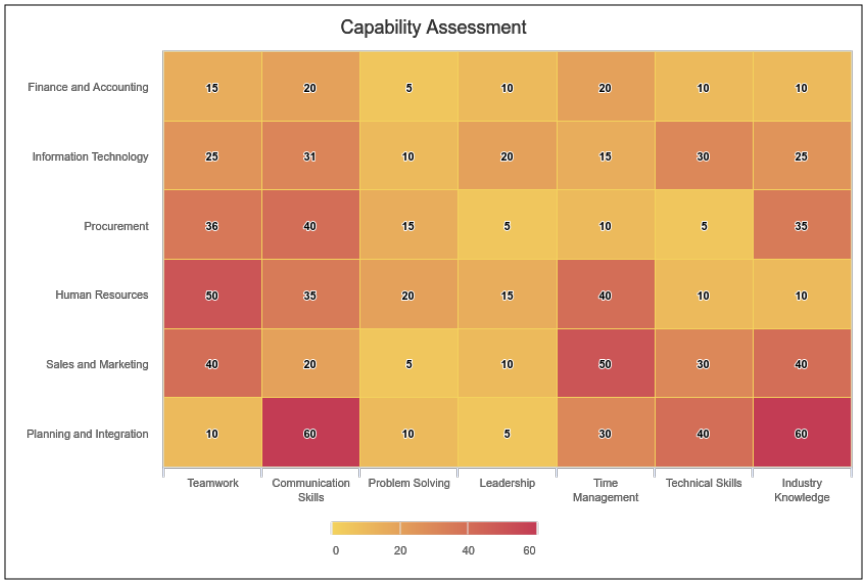
Dispersiones

Permiten distribuir dos o más conjuntos de datos a través de un espacio bidimensional o tridimensional. Muestran la correlación entre los conjuntos de datos y la dispersión de sus variables. Las dispersiones más comunes suelen tener la forma de gráfico de burbujas o gráfico de puntos (también conocidas como parcela XY o simplemente dispersión).



Mapas de calor

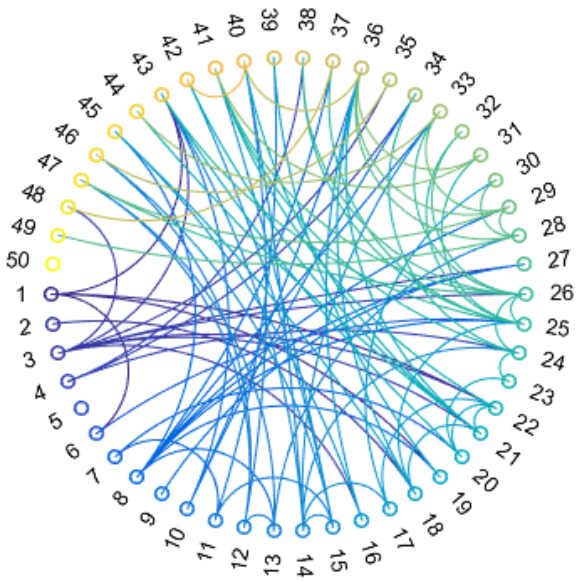
Permiten el posicionamiento de elementos sobre objetos relevantes o áreas, utilizado en la creación de sinópticos, mapas geográficos, planos, el layout de una página web, etc.



Diagramas y matrices

Los diagramas suelen utilizarse para demostrar relaciones complejas entre los datos. Hay varios tipos, entre los más usados: gráficos de herencia, multidimensionales o de árbol.

Las matrices son una visualización típica del *big data*, que permiten mostrar la correlación entre múltiples data sets que evolucionan en tiempo real.



Herramientas de visualización

Tableau (https://www.tableau.com)

Es una de las líderes en este campo, ya que puede ser usada tanto por analistas como por el usuario o usuaria final. Destaca por su sencilla interfaz y multitud de librerías de visualizaciones interactivas. Incluye un amplio catálogo de integración con otras plataformas como bases de datos SQL, Hadoop o Amazon Web Services. Su uso puede estar enfocado a pequeñas visualizaciones ocasionales o a análisis exhaustivos de los datos. Puede manejar tanto streaming de *big data*, como información estática.

QlikView (https://www.qlik.com)

Es la mayor competencia de Tableau. Aunque, en general, ambas ofrecen unas características bastante similares, como principal diferencia cabría señalar que Qlik destaca por su rendimiento, aunque sus cuadros de mando son principalmente estáticos. Todavía no ofrece la posibilidad de crear visualizaciones en tiempo real.

Power BI (https://powerbi.microsoft.com)

Esta herramienta está más enfocada a ofrecer visualizaciones más complejas y analítica avanzada. Es excepcional, gracias a su facilidad de uso e interfaz de drag and drop (coger y soltar). Es muy intuitiva y dispone de muchas capacidades de integración con otras plataformas, en especial con todos los servicios de Microsoft.

Permite crear informes con visualizaciones de diferentes orígenes de datos a la vez y también es compatible con fuentes en streaming.

Kibana (https://www.elastic.co/products/kibana)

Pertenece al stack de ElasticSearch y solo puede trabajar con los datos de esa base de datos. Por contra, puede ser la mejor herramienta de visualización de logs del mercado. Ofrece muchas posibilidades de analítica avanzada, representación de grafos, generación de alarmas de forma automática y utilización de modelos de machine learning dentro de la misma herramienta y de forma interactiva.

Grafana (https://grafana.com)

Es una de las herramientas de visualización de *big data* e IoT más populares, por ser open source y ofrecer un rendimiento muy bueno a la hora de explotar información en tiempo real. Se integra con más de treinta fuentes distintas, incluyendo AWS y ElasticSearch.

Genera tableros dinámicos, permitiendo representar al mismo tiempo información de distintos orígenes de datos y con múltiples métricas diferentes. También facilita la creación de alertas y notificaciones en función de reglas predefinidas.

4.3 CASOS DE USO DE *BIG DATA*

Ahora que ya se tiene una visión global y completa de cómo funciona y de por qué piezas está formado un diseño *big data*, podemos representar una serie de casos de uso reales que se aplican hoy en día en nuestro entorno más cercano:

Visión 360º de la clientela

Actualmente, cualquier empresa proveedora de servicios debe tener la información más detallada posible acerca de su clientela. La información se origina en múltiples fuentes, desde datos proporcionados por el mismo cliente o clienta, pasando por el uso y la geolocalización que realiza la persona usuaria desde su dispositivo móvil, navegación web, conversaciones telefónicas e intercambio de correos electrónicos con la compañía, hasta el cruce con la tendencia de mercado para agruparlo junto a otros usuarios y usuarias afines, así como el contenido publicado en redes sociales.

Toda esta información permite a las compañías emplear modelos analíticos y de aprendizaje automático para poder personalizar la interacción con la clientela. De este modo, la compañía puede sugerir una nueva estrategia de ahorro al cliente o clienta, productos que le pueden interesar, descuentos que puede aplicar a su factura o identificar aquellas personas con más riesgo de causar baja de la empresa.

Prevención de fraude

Los sistemas de prevención y detección de fraude con *big data* son muy utilizados por los bancos para alertar a sus clientes y clientas de que sus tarjetas están siendo utilizadas de forma fraudulenta, pudiendo bloquearlas de manera automática si se trata de un ataque claro. Esto es sencillo de detectar: la alarma se activa cuando se llevan a cabo varios pagos en distintas ubicaciones en un corto periodo de tiempo o cuando se ha realizado una compra en una ubicación muy lejana al domicilio de la persona, sin indicios de que haya realizado ningún viaje en la fecha indicada.

En los últimos años, estos sistemas se han vuelto más sofisticados y han incluido muchas mejoras para prevenir este tipo de incidentes. Por ejemplo, se puede asociar un factor de riesgo a cada una de las compras realizadas a través de la tarjeta bancaria, de modo que se puedan detectar posibles compras fraudulentas que se salgan del patrón de uso del cliente o clienta. Además de ello, los modelos de predicción van aprendiendo a medida que suceden casos nuevos, con el fin de prevenir diferentes escenarios de fraude. De esta manera, por ejemplo, se pueden ubicar códigos postales o zonas donde el índice de criminalidad es mayor.

Optimización de precios

Los negocios B2C (business-to-consumer o negocio a consumidor) o B2B (business-to-business o negocio a negocio) usan tecnologías y analíticas *big data* para optimizar el precio de sus productos. Para cualquier compañía, el objetivo es establecer un precio a sus productos o servicios de forma que puedan maximizar los beneficios conseguidos. Si el precio es demasiado alto, venderán menos y generarán peores beneficios. En caso contrario, si el precio es demasiado bajo, el negocio no será rentable.

En función del histórico de precios, transacciones y el resto de condiciones del mercado, estas compañías son capaces ahora de establecer un precio automático en función de diversas estrategias. Las soluciones *big data* pueden, además, segmentar a la clientela y ofrecer distintas alternativas al servicio ofrecido teniendo en cuenta las necesidades de cada persona, posición geográfica, grupos de edad, estatus social, etcétera.

Motores de recomendación

Es uno de los casos de uso más populares, aplicado por todas las plataformas de reproducción de contenido streaming. Utilizando el historial de películas, series o canciones reproducidas con anterioridad, el sistema es capaz de recomendar contenido afín. Estos algoritmos también son utilizados por páginas de venta online, para recomendar productos similares o del gusto de la persona usuaria, o por motores de búsqueda para ofrecer la publicidad más adecuada a cada perfil.

4.4 TENDENCIAS

¡Hola! Ahora que ya conocemos el estado actual del ecosistema *big data* y sus usos más habituales, es hora de repasar lo que debería ofrecer esta tecnología en el futuro más cercano. ¡Vamos!

Cloud y *big data*

Cada vez son más las empresas que eligen la cloud para poder almacenar, transformar y analizar su información con bajo coste y de forma rápida. Esto no quiere decir que todas las compañías utilicen la nube como sistema principal. Las nubes híbridas son y parecen ser la solución ideal en muchos escenarios. Los procesos batch, ciertas automatizaciones e información sensible seguirán ejecutándose en entornos on premise, mientras que gran parte del procesamiento en tiempo real y la evaluación de modelos de aprendizaje automático se procesará en la nube.

Los beneficios del uso de la computación en la nube sobre arquitecturas *big data* son incuestionables. Las compañías son capaces de establecer una arquitectura escalable y distribuida rápidamente, reduciendo costes en adquisición de máquinas y en el mantenimiento de un data center. Por el contrario, el éxito se encuentra en el equilibrio. No

siempre interesará la inversión de llevarnos toda la información y procesos a la nube. Hay que recordar que, en muchos casos, cada proveedor –véase Microsoft Azure, Amazon Web Services o Google Cloud– impone sus servicios propietarios, condicionando a que todas las aplicaciones y desarrollos que se desplieguen a partir de entonces se lleven a cabo exclusivamente sobre su plataforma.

La importancia de la analítica en tiempo real

Las fuentes de información que proveen a los sistemas de flujos streaming y a su analítica serán cada vez más demandadas por las organizaciones. Estos datos suelen ofrecer información de origen web, redes sociales, dispositivos móviles, servidores o información de la red corporativa. Las analíticas streaming son el plato fuerte de las plataformas cloud. Permiten capturar la información en tiempo real, ayudando a la compañía a que la toma de decisiones sea rápida y precisa.

Los casos de uso más interesantes se encuentran en el mantenimiento predictivo, ciberseguridad, optimización de operaciones, detección de fraude o aplicación de reglas sobre el mercado bursátil. La evolución natural de la tecnología provocará que cada vez sea más sencillo y eficiente implantar modelos predictivos y analítica streaming en las compañías, por lo que en los próximos años será muy común ver este tipo de operativas en nuestro entorno más cercano.

Crecimiento del machine learning

Los próximos años ayudarán a que la convergencia entre la analítica tradicional y los algoritmos de aprendizaje automático sea mucho más integrada. Veremos más organizaciones utilizando machine learning para mejorar las actividades de negocio.

Hasta ahora, las compañías tenían equipos distintos de data science para evaluar los modelos y equipos de ingeniería del dato para adquirir y transformar la información. Estos perfiles profesionales se están convirtiendo en conjuntos mixtos que serán capaces de llevar a cabo el flujo completo del dato con más naturalidad.

De hecho, existen dos tendencias que están acelerando el uso de modelos de machine learning. La primera es que la barrera de entrada cada vez es menor, cualquier persona con conocimientos de programación básicos puede ejecutar un modelo en unos sencillos pasos. La segunda es que cada vez son más las herramientas automáticas que permiten productivizar este tipo de modelos. Actualmente, un data science que crea un buen algoritmo de aprendizaje automático debe esperar a que el ingeniero o ingeniera del dato establezca el flujo necesario para llevarlo a producción. En los próximos años, los frameworks de automatización permitirán a los data science realizar estas operaciones de forma autónoma.

Narrativa de datos y visualización

A medida que los data warehouses se consolidan de forma más ordenada y optimizada en la cloud, se va a producir una mejora inevitable en la calidad de los resultados ofrecidos por la analítica *big data*. Esto se va a traducir en que cada vez va a ser más fácil narrar historias relevantes y precisas a través de cuadros de mando avanzados. Esto, junto a que los modelos de aprendizaje automático aportarán una visión de clasificación, clusterización y predicción mucho más precisa, seremos capaces de ofrecer mucho más valor al dato de la compañía, generando mayores beneficios con menos esfuerzo.

Estas serían las tendencias a tener en cuenta en los próximos años. Somos conscientes de que la evolución de estas tecnologías es exponencial y debemos estar preparados para adaptarnos a las próximas innovaciones que nos van a ofrecer los sistemas *big data*. ¡Las posibilidades para mejorar nuestro día a día son infinitas!

4.5 IDEAS CLAVE: VISUALIZACIÓN DE DATOS: PROGRAMAS Y METODOLOGÍAS

Una vez finalizado el módulo, pasaremos a repasar los puntos más importantes que debemos tener en cuenta.

Visualización de datos

Estas herramientas nos permiten ofrecer una capa de visualización a los datos almacenados en nuestro *data warehouse*. Se caracterizan por su facilidad de uso, por su integración con cualquier tipo de base de datos o sistema de ficheros distribuidos, por su capacidad de representación de procesamiento *batch* o tiempo real y por la gran variedad de recursos de que dispone a la hora de generar cuadros de mando complejos.

Son fundamentales para cualquier negocio, pues ayudan en la toma de decisiones, en proporcionar el estado actual de la información corporativa, en la detección de posibles amenazas o incidencias y en la anticipación de resultados. Además, estos servicios están evolucionando rápidamente para poder ofrecer soluciones analíticas integradas con la herramienta, de modo que podamos mostrar alternativas de *data discovery* y predicciones automáticas en tiempo de visualización.

Es importante recordar los cinco factores que se han de tener en cuenta a la hora de diseñar una visualización de datos: a qué público está dirigida, el tipo de contenido que se quiere ofrecer, su contexto, el dinamismo de los gráficos y el propósito o mensaje que se quiere presentar.

Por otro lado, entre las herramientas de visualización más importantes del mercado encontramos: Tableau, QlikView, Power BI, Kibana y Grafana. Todas tienen sus puntos fuertes y deberemos elegirlas cuidadosamente en función de nuestro presupuesto y necesidades.

Casos de uso

De entre todos los escenarios más típicos dentro del ecosistema *big data*, destacamos cuatro: visión 360º de la clientela, prevención de fraude, optimización de precios y motores de recomendación. Todos ellos se basan en arquitecturas *big data* distribuidas y hacen uso de los *frameworks* que se han visto en el curso.

Es fundamental tener en cuenta la repercusión que están teniendo este tipo de tecnologías en nuestro día a día. Gracias al uso de dispositivos electrónicos, plataformas en la nube, redes sociales, adquisición de productos en comercios, utilización de herramientas financieras, etcétera, generamos una cantidad de datos que son utilizados por empresas grandes o proveedoras de servicios en sus analíticas de datos.

Tendencias

Para finalizar, hemos visto cuál será la evolución natural de *big data* en los próximos años. Se espera que todas las herramientas que se han mostrado a lo largo del curso sigan su desarrollo hacia un mejor rendimiento y una mejor economía de recursos. Alguno de los puntos más importantes del *roadmap* sería el diseño de arquitecturas en nubes híbridas, la focalización en procesamiento de flujos *streaming*, la mejora y crecimiento de los algoritmos de aprendizaje automático y el progreso en la visualización de datos.

Descubre todo lo que Barcelona Activa puede hacer por ti

Acompañamiento durante todo el proceso de búsqueda de empleo

barcelonactiva.cat/treball

Apoyo en la puesta en marcha de tu idea de negocio

barcelonactiva.cat/emprenedoria

Servicios a las empresas e iniciativas socioempresariales

barcelonactiva.cat/empreses

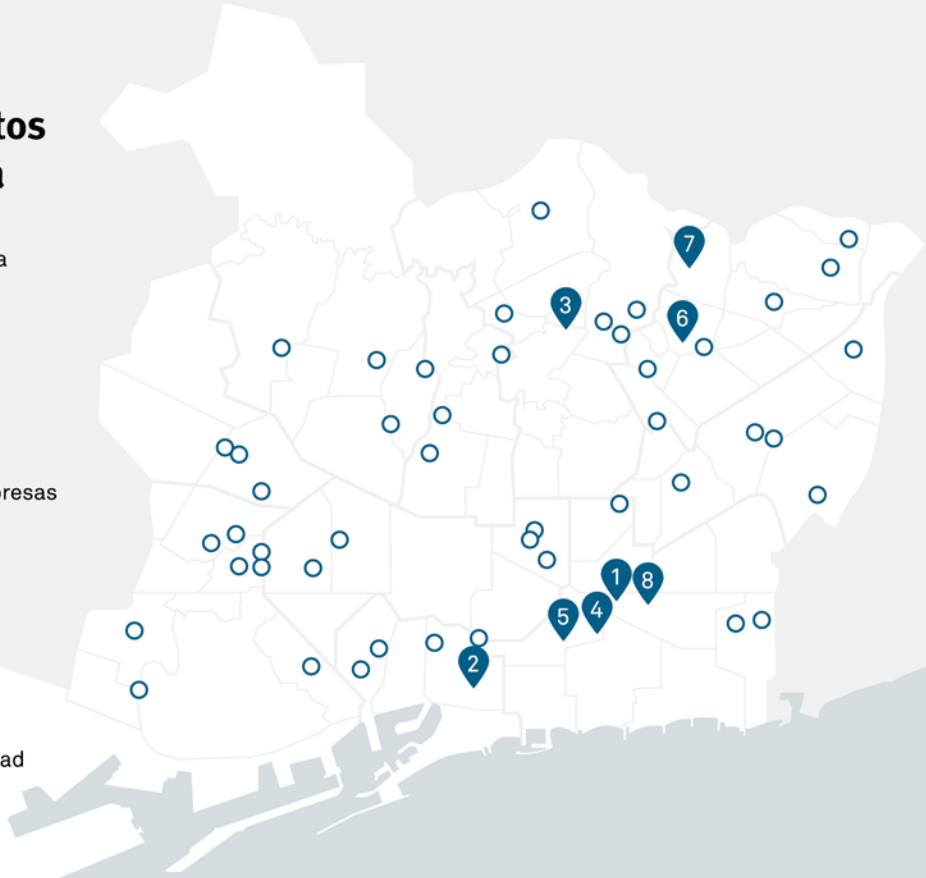
Formación tecnológica y gratuita para la ciudadanía

barcelonactiva.cat/cibernarium

Red de equipamientos de Barcelona Activa

1. Sede Central Barcelona Activa
   Porta 22
   Centro para la Iniciativa Emprendedora Glòries
   Incubadora Glòries
2. Convent de Sant Agustí
3. Ca n'Andalet
4. Oficina de Atención a las Empresas
   Cibernàrium
   Incubadora MediaTIC
5. Incubadora Almogàvers
6. Parque Tecnológico
7. Nou Barris Activa
8. innoBA

○ Puntos de atención en la ciudad



Barcelona Activa

Ajuntament de Barcelona

Última actualización 2020

Cofinanciado por:

UNIÓ EUROPEA
Fons Europeu de Desenvolupament Regional

Síguenos en las redes sociales:

barcelonactiva.cat/cibernarium
barcelonactiva
barcelonactiva
company/barcelona-activa