Material Imprimible

Curso Data Analytics

Módulo 1: Fundamentos de Data Analytics

Contenidos:

- ¿Qué es Data Analytics?
- Tecnologías aplicadas
- Dato versus información
- Tipos de datos
- Bases de datos
- Modelo Relacional. Objetos y concepto
- Lenguaje SQL

¿Qué es Data Analytics?

Data Analytics es un conjunto de técnicas que permiten la recolección, transformación, análisis y visualización de datos, con el objetivo de generar información con valor agregado.

Es una práctica que combina la captura de los datos, la aplicación de modernas tecnologías para su procesamiento, análisis y transformación, y la presentación de resultados como información.

Tecnologías aplicadas:

- Bases de datos

Son herramientas que permiten almacenar datos y procesarlos cuando se los requiera. Es la principal tecnología para el respaldo de los datos que poseen las organizaciones. Pueden usarse para diversos fines, tanto para alojar transacciones como para realizar análisis.

Los principales proveedores de tecnología ofrecen bases de datos, como Microsoft SQL Server, Oracle DataBase, Sap Hana, entre muchos otros.

Hoy en día, con el auge de la computación en la nube, las bases de datos pueden ser accedidas a través de entornos web (como Google Chrome o Firefox), aunque muchas empresas aún mantienen el acceso mediante software de gestión de bases de datos.

- Herramientas de visualización

Es una tecnología que cada vez está más presente en todas las organizaciones. Si bien no ofrecen funcionalidades de almacenamiento, permiten conectarse a múltiples fuentes de datos (incluyendo las mencionadas bases de datos) para utilizar los datos y generar visualizaciones de todo tipo. Las mismas pueden ser desde un simple gráfico a una compleja infografía interactiva. Las herramientas de visualización más famosas son

Microsoft PowerBI, Tableau y Google Data Studio, aunque hay otras tecnologías muy utilizadas.

- Lenguajes de programación

Es el nivel más complejo de análisis de datos, ya que consiste en escribir código en un editor o entorno de desarrollo integrado. Los lenguajes más utilizados son R y Python. El primero tiene un enfoque estadístico y cuenta con numerosas librerías que permiten hacer todo tipo de análisis, desde gráficos hasta testeo de hipótesis y predicciones. Python tiene un desarrollo más amplio, con un claro enfoque en el Analytics avanzado y su aplicación en Data Science, aunque también es muy cómodo su uso para análisis exploratorios.

Datos versus información

Un dato es la unidad mínima de almacenamiento que existe, previo a ser procesado. Justamente el poder ser almacenado lo vuelve un activo importante a la organización, que debe utilizarse en el momento adecuado para mejorar procesos de toma de decisiones.

Información es el resultado de procesar datos almacenados. Puede consistir desde una simple respuesta a un conjunto de gráficos complejos. El uso de la información se transforma en una herramienta central para que una organización agregue valor a su modelo de negocios y/u operación.

Tipos de datos

Los datos no son todos similares. Según su composición, los principales tipos de datos son los siguientes:

Tipo de dato	Descripción	Ejemplo
Texto / Cadena	Contenido alfanumérico. Pueden ser desde una letra o número, una palabra o un texto.	“Hola” “Jose Perez” “Boca10”
Enteros	Números enteros	1 10 100
Decimales	Números con datos decimales	1,25 3,14 100,3476
Fecha Fecha y Hora	Almacenan desde el año hasta los milisegundos. Puede optarse por almacenarse sólo la fecha	“2022-01-01” “2022-01-01 10:30:25.62”
Booleanos	Definen si es Verdadero o Falso el contenido de esa unidad	True False

Bases de datos

Una base de datos es un repositorio donde los datos se almacenan y pueden ser procesados y consumidos por otros aplicativos. Hay dos principales tipos de bases de datos:

- Estructuradas: son las bases de datos tradicionales y más utilizadas. Se componen de objetos que almacenan los datos en estructuras tabulares, típicamente denominadas tablas. Cada una de estas tablas se componen de filas y columnas. El diseño, construcción y uso de estas de bases de datos se respalda en el lenguaje de consultas estructurado (SQL).

Por ejemplo, el almacenamiento de ventas facturas podría ser parte de una base de datos estructurada.

- No estructuradas: son bases de datos que permiten el almacenamiento multimedia y de todos aquellos objetos que no posean una estructura tabular, aunque también podrían almacenar de este último. Surgen ante el auge del streaming y la generación de imágenes y sonidos con las nuevas tecnologías. El procesamiento y análisis de estos datos no puede ser procesado con el SQL tradicional, por lo cual surgen otras tecnologías de avanzada, como el NoSQL.

  Por ejemplo, un repositorio de videos de una cámara de seguridad podría ser almacenado en una base sin estructura y analizado por alguna herramienta que permita obtener patrones de movimiento.

En nuestro curso introductorio a data analytics, conoceremos las bases de datos estructuradas.

Modelo Relacional

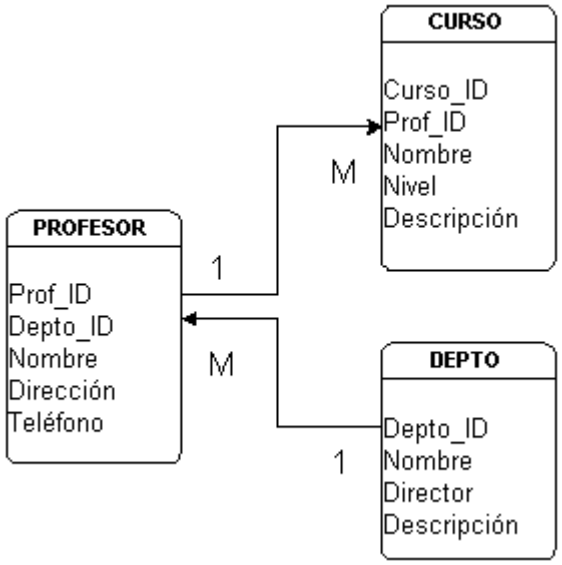
Un modelo de datos es un esquema que permite brindar una estructura lógica de la base, incluidas las relaciones y limitaciones que determinan cómo se almacenan los datos y cómo se accede a ellos. Típicamente, el diseño de un modelo de datos intenta conceptualizar una “realidad” (modelo de negocios o datos del mundo real que son almacenados).

El tipo de modelo de datos más famoso es el modelo relacional, que se lo diagrama mediante un esquema entidad-relación.

Las ventajas que ofrece son las siguientes:

- Evitar redundancia y duplicación de datos
- Seguridad y restricciones

- Permite segmentar los análisis, ya que puedo optar por explorar datos de las entidades que desee, sin utilizar toda la base completamente.



Los modelos se componen de entidades, claves y relaciones.

Las entidades son los objetos que existen en el mundo real y que son almacenados dentro de nuestro modelo de datos. Por ejemplo, si somos una empresa que vende automóviles, una entidad serían los “Vehículos” que se comercializan y otra entidad los “Vendedores” que los venden.

Dentro de cada entidad, existen atributos, que son variables propias de dicha entidad. Por ejemplo, la marca del vehículo sería un atributo de “Vehículos” y la fecha de nacimiento del “Vendedor”.

Las claves son atributos que permiten establecer vínculos entre las distintas entidades, recreando las relaciones que existen en el mundo real.

Pueden ser:

- Claves Primarias (o Primary Key): identifican unívocamente cada registro (fila) dentro de la entidad (tabla). No puede repetirse, solo puede estar asociado una

única vez. Por ejemplo: la patente en vehículos, el legajo en vendedores, el Documento de Identidad en personas.

- Claves Foráneas (o Foreign Key): permiten establecer el vínculo de la entidad actual, haciendo referencia a otra entidad exterior. Por ejemplo, si en la tabla Vehículo tenemos el DNI del dueño, ese atributo hará referencia a una tabla de Titulares donde mediante el DNI podremos obtener los datos propios de esa persona, como el Sexo o la Fecha de Nacimiento.

Por último, encontramos las relaciones, que son vínculos entre las entidades mediante las mencionadas claves. Dichas relaciones tienen restricciones de cardinalidad, es decir, cuántos registros de una entidad, pueden vincularse a la otra y viceversa.

Hay 3 tipos de relaciones:

1 a 1: Un registro de Entidad A se relaciona con un solo registro de Entidad B.

Por ejemplo:

Un alumno tiene un solo mail registrado en la entidad "Contacto"

Se coloca el "1" en cada extremo al documentar el diagrama.

1 a Muchos: Un registro de Entidad A se relaciona con muchos registros de Entidad B.

Por ejemplo:

Un profesor tiene asignado muchos cursos para brindar, y cada curso está asociado solamente a ese profesor.

En este caso, se coloca el 1 del lado de la entidad externa que provee información, y el * (asterisco) es la nomenclatura usada para identificar el "Muchos" (también puede utilizarse las letras N o M) en la tabla donde se les asocia la cantidad de registros superior a 1, en este caso, los cursos.

Muchos a Muchos: Muchos registros de A se relacionan con muchos registros de B. Esto requiere una tabla intermedia para su resolución, partiendo el vínculo en dos relaciones 1 a Muchos.

Por ejemplo:

Un alumno se inscribe en muchos cursos, pero cada curso puede tener muchos alumnos. (requiere tabla intermedia, conceptualizada como "Inscripciones")

En este tipo de relaciones, necesitamos una tercera tabla ya que es un ERROR utilizar un vínculo * a *. De esta forma, partimos ese problema en una tabla intermedia que tiene la relación final entre alumnos y cursos.

Lenguaje SQL

El Lenguaje de Consultas Estructurado (SQL, de sus siglas en inglés) es la funcionalidad central que ofrecen las bases de datos para el desarrollo, diseño, mantenimiento y uso de los modelos de datos.

Consiste en escribir consultas que envían peticiones a las bases de datos con diversas operaciones.

Sus principales ventajas son:

- Su curva de aprendizaje es corta
- Es universalmente aceptado, ya que existe hace décadas y se utiliza en todo el mundo como la herramienta central para la gestión de bases de datos
- Si los recursos asignados son adecuados, el tiempo de respuesta es muy alto a comparación a otras herramientas y/o lenguajes de programación

Su sintaxis consiste en sentencias o comandos que escribimos en inglés y luego la base de datos interpreta, dándonos una respuesta.

Las principales sentencias son:

- SELECT, que permite identificar qué columnas quiero.
- FROM, que define de dónde obtendría esas columnas.
- WHERE, donde se definen qué condiciones deben cumplir los registros que voy a visualizar.

Aplicaciones de SQL:

- Diseño y desarrollo de bases de datos
- Construcción de modelos relacionales
- Gestión de la seguridad de la base de datos
- Gestión del rendimiento de la base de datos

- Administración de accesos a los datos
- Carga, modificación y eliminación de datos
- Extracción y manipulación para el análisis de los datos

¡El analista de datos típicamente trabaja en el último punto!