



# Ciencia de Datos

MDataSc(c). Ing. Natalia Botto Pérez

# Bienvenidos al Curso de Ciencia de Datos

## Objetivo

El participante logrará comprender los desafíos que conlleva la implementación de Ciencia de Datos e incorporará los conocimientos necesarios para obtener información relevante a partir de fuentes de datos sin procesar.

Será capaz de definir los objetivos y el problema de negocio, aplicar técnicas de obtención, análisis, limpieza y transformación de datos logrando traducir los resultados en alternativas de toma de decisión y acción.

# Bienvenidos al Curso de Ciencia de Datos

## Temario

- Conceptos básicos y reseña histórica del origen de la Ciencia de Datos.
- Diferencias entre Ciencia de Datos, Análisis de Datos, Inteligencia Artificial y Aprendizaje Automático.
- ¿Cómo se lleva a cabo la Ciencia de datos y de qué manera la Ciencia de Datos está transformando los negocios?
- Aplicaciones más conocidas de Aprendizaje Automático y Ciencias de Datos en el mundo real.
- Herramientas para Ciencia de Datos.
- Plataformas para proyectos de Ciencia de Datos.
- Casos de éxito. Analítica para la toma de decisiones y competitividad.
- Metodología para proyectos de Analítica.

# Bienvenidos al Curso de Ciencia de Datos

## Temario

- Preparación de datos estructurados.
- Analítica de Procesos.
- Conceptos e implementación de KPIs.
- Elaboración de Paneles de Control de Gestión (Control Dashboards).
- Herramientas de visualización, análisis e interpretación de visualizaciones.
- Riesgos y consideraciones al realizar proyectos de Analítica de Datos.
- Principios y métodos de anonimización.
- Tecnologías para analizar grandes volúmenes de datos.

# Bienvenidos al Curso de Ciencia de Datos

- **Inicio:** 12 de setiembre de 2022.
- **Finalización:** 24 de octubre de 2022.
- **Régimen:** lunes y miércoles de 18 a 19:45.
- **Modalidad:** 100 % virtual, con clases en vivo.

# Presentación

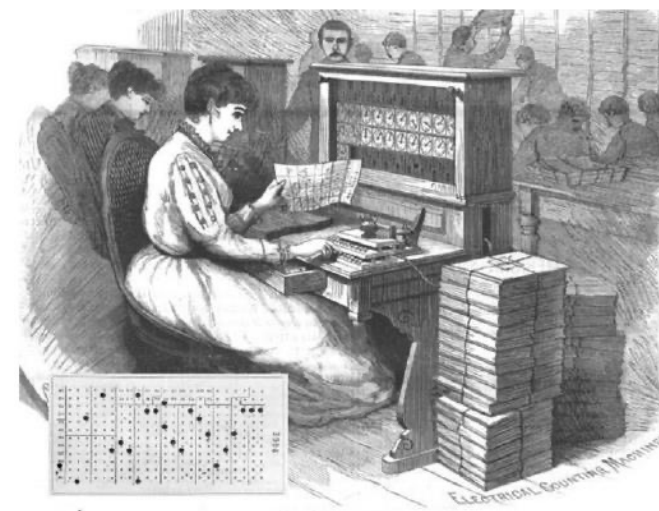
- **Nombre**
- **Profesión**
- **¿Qué espero de este curso?**

# Datos

Si repasamos en la historia como fue evolucionado el almacenamiento de datos, los primeros registros los encontramos en los medios físicos (libros, enciclopedias, diccionarios, etc.).

Con la llegada de la tecnología aparecieron las tarjetas perforadas, que permitieron el almacenamiento y procesamiento de datos, de forma mecánica.

Máquinas tabuladoras y tarjetas perforadas utilizadas para procesar el censo de los Estados Unidos de 1890.  
Tomado del libro "Next Generation Databases" de Guy Harrison

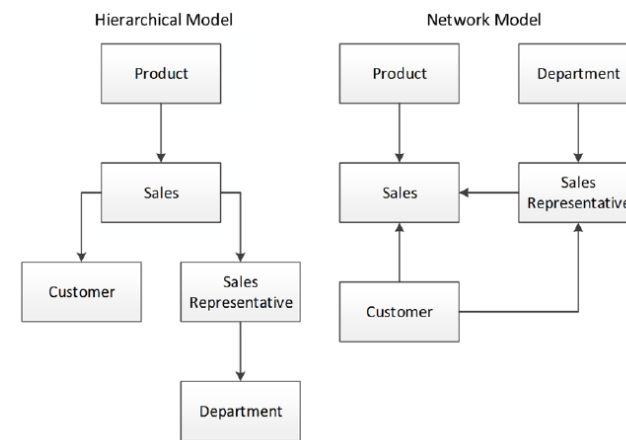


# Datos

Los primeros DBMS ofrecían la posibilidad de definir tanto la estructura de los datos como el acceso a navegar entre los registros. Eran de uso exclusivo de mainframes.

Las estructuras eran jerárquicas o de red, pero tenían el problema de ser complejos al momento de modificar las estructuras de datos y las capacidades de consulta. La codificación de reportes era sumamente compleja.

Estructura jerárquica y de red de bases de datos.  
Tomado del libro “Next Generation Databases” de Guy Harrison





# Datos

En 1962 John Tukey describe un campo llamado Data Analysis en su artículo The Future of Data Analysis, para explicar la evolución de la estadística matemática.

Luego en 1970 Edward Codd publicó el paper “A relational model of data for large shared data banks” que contenía las bases de los conceptos que luego se transformaron en las bases de datos relacionales.

Esto quiere decir estructuras pre definidas y relacionados entre si, orientadas a transacciones en las cuales se realizaban tareas CRUD (insertar, leer, actualizar o eliminar)

## The End of an Architectural Era (It's Time for a Complete Rewrite)

Michael Stonebraker  
Samuel Madden  
Daniel J. Abadi  
Stavros Harizopoulos  
MIT CSAIL

(stonebraker, madden, dms,  
stavros}@csail.mit.edu

Nabil Hachem  
AvantGarde Consulting, LLC  
nhachem@agdba.com

Pat Heiland  
Microsoft Corporation  
philand@microsoft.com

### ABSTRACT

In previous papers [SC05, SBC+07], some of us predicted the end of “one size fits all” as a commercial relational DBMS paradigm. These papers presented reasons and experimental evidence that showed that the major RDBMS vendors can be outperformed by 1-2 orders of magnitude by specialized engines in the data warehouse, stream processing, text, and scientific database markets.

Assuming that specialized engines dominate these markets over time, the current relational DBMS code lines will be left with the business data processing (OLTP) market and hybrid markets where more than one kind of capability is required. In this paper we show that current RDBMSs can be beaten by nearly two orders of magnitude in the OLTP market as well. The experimental evidence comes from comparing a new OLTP prototype, H-Store, which we have built at MIT, to a popular RDBMS on the standard transactional benchmark, TPC-C.

We conclude that the current RDBMS code lines, while attempting to be a “one size fits all” solution, in fact, stand at nothing. Hence, they are 25 year old legacy code lines that should be retired in favor of a collection of “from scratch” specialized engines. The DBMS vendors (and the research community) should start with a clean sheet of paper and design systems for tomorrow’s requirements, not continue to patch code lines and architectures designed for yesterday’s needs.

### 1. INTRODUCTION

The popular relational DBMSs all trace their roots to System R, from the 1970s. For example, DB2 is a direct descendant of System R, having used the RDS portion of System R, intact in their first release. Similarly, SQL Server is a direct descendant of Sybase System 5, which borrowed heavily from System R. Lastly, the first release of Oracle implemented the user interface from System R.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the VLDB copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Very Large Database Endowment. To copy otherwise, or to republish, to post on servers or to redistribute to lists, requires a fee and/or special permission from the publisher, ACM.

VLDB ’07, September 23-28, 2007, Vienna, Austria.  
Copyright 2007 VLDB Endowment, ACM 978-1-59593-592-0/07

All three systems were architected more than 25 years ago, when hardware characteristics were much different than today. Processors are thousands of times faster and memories are thousands of times larger. Disk volumes have increased enormously, making it possible to keep essentially everything, if one chooses so. However, the bandwidth between disk and main memory has increased much more slowly. One would expect this relentless pace of technology to have changed the architecture of database systems dramatically over the last quarter of a century, but surprisingly the architecture of most DBMSs is essentially identical to that of System R.

Moreover, at the time relational DBMSs were conceived, there was only a single DBMS market, business data processing. In the last 25 years, a number of other markets have evolved, including data warehouses, text management, and stream processing. These markets have very different requirements than business data processing.

Lastly, the main user interface device at the time RDBMSs were architected was the dumb terminal, and vendors imagined operators inputting queries through an interactive terminal prompt. Now it is a powerful personal computer connected to the World Wide Web. Web sites that use OLTP DBMSs rarely run interactive transactions or present users with direct SQL interfaces.

In summary, the current RDBMSs were architected for the business data processing market in a time of different user interfaces and different hardware characteristics. Hence, they all include the following System R architectural features:

- Disk-oriented storage and indexing structures
- Multithreading to hide latency
- Locking-based concurrency control mechanisms
- Log-based recovery

Of course, there have been some extensions over the years, including support for compression, shared-disk architectures, bitmap indexes, support for user-defined data types and operators, etc. However, no system has had a complete redesign since its inception. This paper argues that the time has come for a complete rewrite.

A previous paper [SBC+07] presented benchmarking evidence that the major RDBMSs could be beaten by specialized architectures by an order of magnitude or more in several application areas, including:

# Datos

A principios del 2000, las bases de datos relacionales dejaron de cubrir las necesidades de las empresas que manejaban **grandes volúmenes de datos** y que requerían también **velocidad de respuesta** para ese volumen de datos.

Las primeras en tener estas necesidades fueron empresas como Google, Amazon, Facebook, MySpace; las necesidades de escalabilidad que requería el negocio no podía ser cubierto por las bases de datos relacionales.

En particular, la diferencia en las arquitecturas de aplicaciones entre la era cliente-servidor y la era de las aplicaciones masivas a escala web crearon presiones en la base de datos relacional que no se pudieron aliviar a través de la innovación incremental.

En 2001 William S. Cleveland publico el paper *Data Science: An Action Plan for Expanding the Technical Areas of the Field of Statistics*

# Datos

En 2007, Michael Stonebraker, pionero de los sistemas de bases de datos Ingres y Postgres, dirigió un equipo de investigación que publicó el paper “The End of an Architectural Era (It’s Time for a Complete Rewrite) ”.

Este paper señaló que las arquitecturas relacionales de consenso no aplicaban más, y que la variedad de cargas de trabajo de bases de datos modernas sugiere que una arquitectura única podría no aplicar en todos los sistemas.

## The End of an Architectural Era (It’s Time for a Complete Rewrite)

Michael Stonebraker  
Samuel Madden  
Daniel J. Abadi  
Stavros Harizopoulos  
MIT CSAIL

(stonebraker, madden, dina,  
stavros)@csail.mit.edu

Nabil Hachem  
AvantGarde Consulting, LLC  
nhachem@agdba.com

Pat Helland  
Microsoft Corporation  
phelland@microsoft.com

### ABSTRACT

In previous papers [SBC05, SBC+07], some of us predicted the end of “one size fits all” as a commercial relational DBMS paradigm. These papers presented reasons and experimental evidence that showed that the major RDBMS vendors can be outperformed by 1-2 orders of magnitude by specialized engines in the data warehouse, stream processing, text, and scientific database markets.

Assuming that specialized engines dominate these markets over time, the current relational DBMS code lines will be left with the business data processing (OLTP) market and hybrid markets where more than one kind of capability is required. In this paper we show that current RDBMSs can be beaten by nearly two orders of magnitude in the OLTP market as well. The experimental evidence comes from comparing a new OLTP prototype, H-Store, which we have built at MIT, to a popular RDBMS on the standard transactional benchmark, TPC-C.

We conclude that the current RDBMS code lines, while attempting to be a “one size fits all” solution, in fact, excel at nothing. Hence, they are 25 year old legacy code lines that should be retired in favor of a collection of “from scratch” specialized engines. The DBMS vendors (and the research community) should start with a clean sheet of paper and design systems for tomorrow’s requirements, not continue to patch code lines and architectures designed for yesterday’s needs.

### 1. INTRODUCTION

The popular relational DBMSs all trace their roots to System R, from the 1970s. For example, DB2 is a direct descendant of System R, having used the RDS portion of System R, intact in their first release. Similarly, SQL Server is a direct descendant of Sybase System S, which borrowed heavily from System R. Lastly, the first release of Oracle implemented the user interface from System R.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the VLDB copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Very Large Database Endowment. To copy otherwise, or to republish, to post on servers or to redistribute to lists, requires a fee and/or special permission from the publisher, ACM.

VLDB ’07, September 23–28, 2007, Vienna, Austria.  
Copyright 2007 VLDB Endowment, ACM 978-1-59593-445-1/07/00

All three systems were architected more than 25 years ago, when hardware characteristics were much different than today. Processors are thousands of times faster and memories are thousands of times larger. Disk volumes have increased enormously, making it possible to keep essentially everything, if one chooses to. However, the bandwidth between disk and main memory has increased much more slowly. One would expect this relentless pace of technology to have changed the architecture of database systems dramatically over the last quarter of a century, but surprisingly the architecture of most DBMSs is essentially identical to that of System R.

Moreover, at the time relational DBMSs were conceived, there was only a single DBMS market, business data processing. In the last 25 years, a number of other markets have evolved, including data warehouses, text management, and stream processing. These markets have very different requirements than business data processing.

Lastly, the main user interface device at the time RDBMSs were architected was the dumb terminal, and vendors imagined operators inputting queries through an interactive terminal prompt. Now it is a powerful personal computer connected to the World Wide Web. Web sites that use OLTP DBMSs rarely run interactive transactions or present users with direct SQL interfaces.

In summary, the current RDBMSs were architected for the business data processing market in a time of different user interfaces and different hardware characteristics. Hence, they all include the following System R architectural features:

- Disk oriented storage and indexing structures
- Multithreading to hide latency
- Locking-based concurrency control mechanisms
- Log-based recovery

Of course, there have been some extensions over the years, including support for compression, shared-disk architectures, bitmap indexes, support for user-defined data types and operators, etc. However, no system has had a complete redesign since its inception. This paper argues that the time has come for a complete rewrite.

A previous paper [SBC+07] presented benchmarking evidence that the major RDBMSs could be beaten by specialized architectures by an order of magnitude or more in several application areas, including:

# Datos

La explosión surgió en 2008 – 2009 con la salida de múltiples motores.

El término NoSQL se empezó a utilizar para representar los modelos de base de datos que no son relacionales, pero es un término que puede no resultar feliz, ya que sugiere bases de datos que no utilizan el lenguaje SQL (en muchos de los modelos es así), pero no solo incluye este tipo de modelos sino otros que incluyen variaciones de los modelos relacionales y no relacionales.

Por esta razón se suele decir que NoSQL corresponde a No Solo SQL.



# Datos

## **¿Porque queremos tantos datos?**

Para tomar decisiones, mediante el análisis de esos datos.

Encontrar modelos estadísticos del comportamiento de esos datos para predecir como se pueden llegar a comportar.

Para eso adquirimos – procesamos – analizamos - visualizamos

# Datos

## **¿Porque queremos tantos datos?**

Para tomar decisiones, mediante el análisis de esos datos.

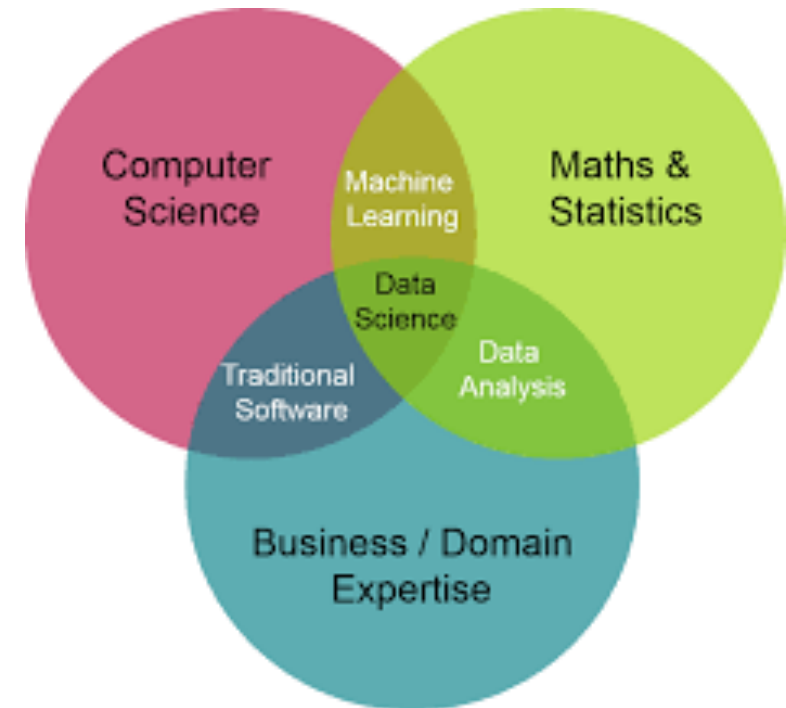
Encontrar modelos estadísticos del comportamiento de esos datos para predecir como se pueden llegar a comportar.

Para eso adquirimos – procesamos – analizamos - visualizamos

# Ciencia de Datos (Data Science)

La ciencia de datos es un campo interdisciplinario que involucra métodos científicos, procesos y sistemas para extraer conocimiento o alcanzar un mejor entendimiento de datos en sus diferentes formas (estructurados y no estructurados). Para ello se basa en algunos campos del análisis de datos como la estadística, la minería de datos, el aprendizaje automático y la analítica predictiva.

Es la conjunción de ciencias computacionales, matemática y estadística y conocimiento del negocio.

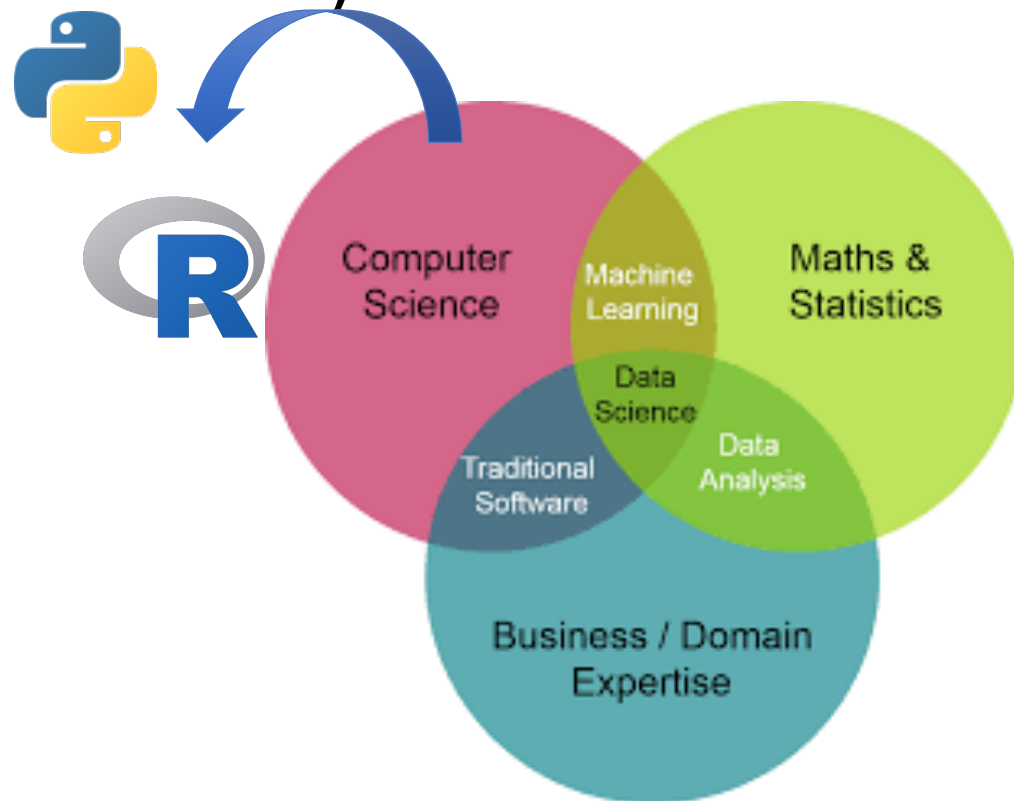




# Ciencia de Datos (Data Science)

La ciencia de datos es un campo interdisciplinario que involucra métodos científicos, procesos y sistemas para extraer conocimiento o alcanzar un mejor entendimiento de datos en sus diferentes formas (estructurados y no estructurados). Para ello se basa en algunos campos del análisis de datos como la estadística, la minería de datos, el aprendizaje automático y la analítica predictiva.

Es la conjunción de ciencias computacionales, matemática y estadística y conocimiento del negocio.





# Científico de Datos (Data Scientist)

El Data Scientist o Científico de Datos, es una persona capaz de analizar e interpretar datos complejos, así como utilizar técnicas de estadística y aprendizaje automático para comprender mejor estos datos y extraer conclusiones que permitan resolver un problema de la realidad.

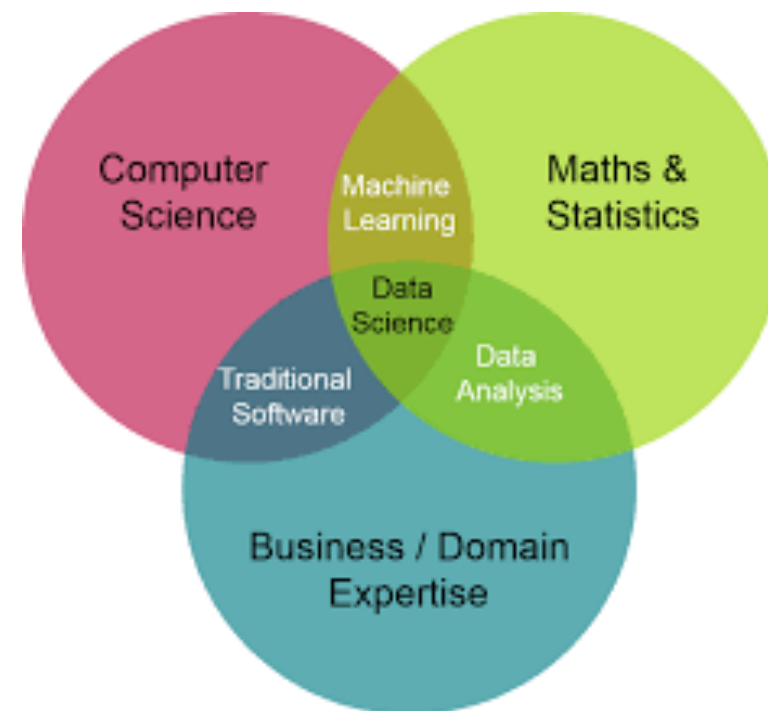
Combina una sólida formación teórica y práctica en las materias fundamentales asociadas al análisis avanzado de datos: pensamiento analítico, comprensión de problemas de la realidad, estadística, programación, tratamiento de bases de datos, trabajo con algoritmos y comunicación efectiva, preparado para encarar problemas de la realidad y convertirlos en soluciones utilizando datos.

Es muy común colocar a un data scientist en la intersección de las siguientes áreas de conocimiento: (i) Ciencias de la Computación, (ii) Matemáticas y Estadística y (iii) Conocimiento de un dominio específico.

# Análisis de Datos (Data Analysis)

Generación de indicadores de forma automática a través de consultas, la ejecución de procedimientos o algoritmos.

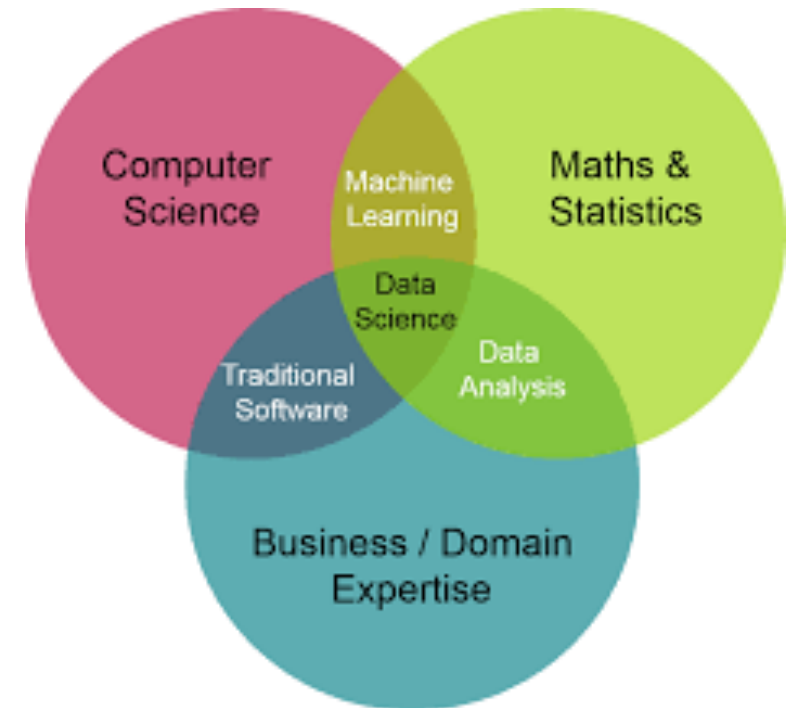
Incluye la identificación de dependencias entre las variables así como patrones ocultos en los sets de datos.



# Aprendizaje Automatico (Machine Learning)

Técnicas de generación de **algoritmos** que permitan el entrenamiento de computadoras para **predecir comportamientos**.

Forma parte de las técnicas de Inteligencia Artificial (Artificial Intelligence).



# Aprendizaje Automatico (Machine Learning)

Dentro de las formas de entrenar a nuestros algoritmos, existen:

- aprendizaje supervisado
  - Se basa en analizar datos históricos para identificar que variables afectan los comportamientos, y de esa manera entrenar modelos
- aprendizaje no supervisado
  - Se busca encontrar patrones en un set de datos sin etiquetas preexistentes y con un mínimo de supervisión humana

# Aprendizaje Automatico (Machine Learning)

Dentro del Aprendizaje Automatico, existe otra forma de generar predicciones y es a través del Aprendizaje Profundo (Deep Learning).

El comportamiento es similar, pero en este caso se utilizan redes neuronales para los procesamientoos.



# Ciencia de Datos

MDataSc(c). Ing. Natalia Botto Pérez

# Metodología de Trabajo

- Es importante contar una metodología o marco de trabajo que me permita sistematizar ciertas etapas como recolectar datos, limpiarlos, generar un modelo predictivo y determinar acciones. Un proceso estándar que traduzca un problema de la vida real en tareas abordables por un equipo de científicos de datos.
- Existen varias metodologías disponibles, a su vez muchas empresas generan sus propios procesos.

# Metodología de Trabajo

En general se basan en definir diferentes etapas:

- **Entender el negocio**
  - Comprensión de los objetivos y requisitos del proyecto desde una perspectiva empresarial, y luego convertir este conocimiento en una definición del problema de ciencia de datos, y un plan preliminar diseñado para alcanzar los objetivos.
- **Recolección y comprensión de datos**
  - Recolección inicial de datos y procesos con actividades con el objetivo de familiarizarse con los mismos, identificar problemas en la calidad de los datos, descubrir primeros insights en los datos, o detectar subconjuntos de datos para formular hipótesis sobre datos ocultos.
- **Preparación de los datos**
  - Actividades para construir el conjunto de datos de entrenamiento. Estas tareas son ejecutadas en múltiples oportunidades y sin orden. Las tareas incluyen selección y transformación de tablas, registros y atributos, y limpieza de datos para las herramientas de modelado.



# Metodología de Trabajo

- **Modelado**
  - Se seleccionan y aplican varias técnicas de modelado y se ajustan los parámetros para mejorar los resultados. Hay varias técnicas que tienen requerimientos específicos sobre la forma de los datos, por lo que puede ser necesario volver a la fase de preparación de datos.
- **Evaluación**
  - Evaluación del modelo (o modelos) construidos, que parecen tener gran calidad desde una perspectiva del análisis de datos.
- **Despliegue**
  - Esta fase depende de los requerimientos, pudiendo ser simple como la generación de un reporte o compleja como la implementación de un proceso de explotación de información que atraviese a toda la organización.

# Metodología de Trabajo

## DataSet

- Un dataset corresponde a los contenidos de una única tabla de base de datos o una única matriz de datos de estadística, donde cada columna de la tabla representa una variable en particular, y cada fila representa a un miembro determinado del conjunto de datos que estamos tratando.
- En un dataset tenemos todos los valores que puede tener cada una de las variables, como por ejemplo la altura y el peso de un objeto, que corresponden a cada miembro del conjunto de datos. Cada uno de estos valores se conoce con el nombre de dato. El conjunto de datos puede incluir datos para uno o más miembros en función de su número de filas.

# Metodología de Trabajo

## DataFrame

- Un DataFrame se organiza en una hoja de datos, en los que cada fila corresponde a un objeto de la muestra y cada columna a una variable. Esta característica de organización de datos es la misma que en los datasets.
- Si hablamos de la estructura de un dataframe es muy similar a la de una matriz.

# Metodología de Trabajo

## **Entender el negocio**

Es importante tener un conocimiento del negocio involucrado a los requerimientos recibidos, para entender la problemática y poder traducir los requerimientos en un requerimiento de ciencia de datos.

# Metodología de Trabajo

## Entender el negocio

- Problemática involucrada
  - Entender el problema y cuales son las expectativas sobre la solución a diseñar
  - Entender la realidad permite definir objetivos y alcance
- Interesados
  - Usuarios que utilizaran la solución, o que pueden verse afectados por la solución diseñada
- Criterios de aceptación
  - ¿Cuáles son los parámetros de aceptación para el proyecto?

# Metodología de Trabajo

## Entender el negocio

- Recursos
  - ¿Cuáles son las fuentes de datos?
  - ¿Los datos están disponibles o tenemos que investigar como obtenerlos?
  - ¿Los datos vienen en un formato legible?
  - ¿Debemos hacer limpieza sobre esos datos?
  - ¿Cuentan con infraestructura para ejecutar la solución diseñada o correr los modelos?
  - ¿Cuentan con personal a disposición para consultas?

# Metodología de Trabajo

## Entender el negocio

- Requisitos, supuestos y restricciones
  - ¿Existen restricciones legales o de seguridad? (por ejemplo en el acceso de los datos)
  - ¿Existen supuestos sobre la calidad de los datos?
  - ¿Existen supuestos o requisitos de como se deben desplegar los resultados?
- Análisis de riesgos
  - ¿Qué riesgos pueden estar involucrados en este proyecto?
    - ¿Tiempo de diseño de la solución?
    - ¿Recursos?
    - ¿Posibilidad de acceso a los datos?

# Metodología de Trabajo

## Recolección de los datos

La recolección de datos es el proceso de obtener o importar los datos que se pretenden analizar desde las diferentes fuentes, y almacenarlos temporal o permanentemente para su posterior análisis.

Como resultado de esta operación, el científico de datos tiene a su disposición uno o varios set de datos en un formato comprendido y manejado por este (archivos CSV, resultado de consultas SQL, data frames, etc), listo para comenzar con el análisis y el entendimiento de los datos.



# Metodología de Trabajo

## Fuentes de Datos

Algunas fuentes de datos:

- Bases de datos
  - Relacionales/No Relacionales
- Recuperacion de datos utilizando APIs
- Sitios Web
  - Técnicas de web crawling y web scraping
- Datos Abiertos
  - Archivos CSV (archivo de texto donde los datos viene con un separador que puede ser ; , | )
  - Archivos JSON (archivo de texto con una estructura jerarquica, donde los datos vienen con una estructura de clave-valor)
  - Otros formatos

# Metodología de Trabajo

## Web Scraping

Es el proceso de extraer información de sitios web.

El web scraping está muy relacionado con la indexación de la web, la cual indexa la información utilizando un robot y es una técnica universal adoptada por la mayoría de los motores de búsqueda.

Sin embargo, el web scraping se enfoca más en la transformación de datos sin estructura en la web (como el formato HTML) en datos estructurados que pueden ser almacenados y analizados en una base de datos central, en una hoja de cálculo o en alguna otra fuente de almacenamiento.

# Metodología de Trabajo

## Web Crawling

Es el proceso de encontrar y recuperar "web links", desde una lista inicial de URLs, navegando el contenido de la lista inicial y aplicando técnicas de web scraping para encontrar nuevos links.

Existen varias herramientas que implementan estas técnicas, una de las más populares es la librería de Python Scrapy, pero también existen herramientas para el lenguaje R.

# Metodología de Trabajo

## Datos abiertos

Los datos abiertos son aquellos que se encuentran disponibles en formatos estándares, abiertos y disponibles en la web para que cualquier persona lo pueda consumir.

Suelen encontrarse en diferentes formatos, pero los principales son: CSV, XML, JSON, TXT y PDF.

Dependiendo de la naturaleza pueden encontrarse también en formatos de imagen (usualmente en datasets de imágenes en el área Computer Vision) u otros formatos mas complejos.

# Metodología de Trabajo

## Fuentes de Datos abiertos

**Datos abiertos estatales:** Son varios los países que ponen a disposición de la comunidad conjuntos de datos abiertos para su explotación

- Catálogo datos abiertos Uruguay 🇺🇾 → <https://catalogodatos.gub.uy/>
- Datos abiertos Argentina 🇲🇪 → <https://datos.gob.ar/>
- Datos abiertos USA 🇺🇸 → <https://www.data.gov/>
- **Kaggle:** Comunidad de data scientist donde podrás encontrar data sets, corpus armados para algoritmos de AI, entre otros → <https://www.kaggle.com/>

# Metodología de Trabajo

## Fuentes de Datos abiertos

- **UCI:** Repositorio de data sets para Machine Learning → <https://archive.ics.uci.edu/ml/index.php>
- **DBpedia:** Información de Wikipedia estructurada de forma semántica  
→ <https://wiki.dbpedia.org/>
- **Yelp:** Datos abiertos de la aplicación Yelp, Reviews de usuarios, Fotos, etc.  
→ <https://www.yelp.com/dataset>
- **Unicef:** Datos abiertos Unicef → <https://data.unicef.org/resources/resource-type/datasets/>
- **NASA:** Catalogo de datos abiertos de la NASA → <https://data.nasa.gov/>
- **Google DataSets Search:** Herramienta de Google, que busca datasets en la Web. En particular, indexa resultados de otras plataformas, como Kaggle, Datos Abiertos Estatales, etc.
  - <https://toolbox.google.com/datasetsearch>

# Metodología de Trabajo

## Desafíos con las fuentes de Datos

Existen ciertos desafíos que siempre están presentes en la etapa de adquisición de datos, los principales son:

- **Acceso a los datos:** Problemas de brindar el acceso a los datos por parte del cliente.
- **Formato de los datos:** Los datos están en un formato que no comprendemos o no entendemos.
- **Problemas de encoding, escapeo de caracteres:** Que pasa si el encoding de un CSV no es UTF-8, si tiene columnas sin escapeo y algunos valores contienen el caracter separador. Suele llevar muchas veces una cantidad considerable de tiempo pre-procesando los datos para poder ingerirlos y empezar la etapa de comprensión y análisis.

# Metodología de Trabajo

## Comprensión de los datos

Previo a poder empezar a trabajar con los datos, debemos recopilar y estudiar de cerca los datos disponibles.

Identificar que datos tenemos, explorarlos con ayuda de graficos o tablas, poder describir los datos disponibles y verificar la calidad de los mismos permite saber como avanzar.



# Metodología de Trabajo

## Metadata de los datos

La metadata se refiere a los datos de los datos, o sea nos dice a que corresponde cada dato, que representa, que tipo de datos incluye y en algunos casos, que valores puede tomar (ejemplo, codigueras).

Es muy importante contar con metadata o documentación sobre los datos. En caso de no contar con documentación de la metadata, es conveniente generarla.

# Metodología de Trabajo

## **Preparación de los datos**

La preparación de los datos es de las tareas que lleva el mayor tiempo y esfuerzo del proyecto (entre el 50% y 70%)

Debemos analizar los datos que son relevantes a los objetivos (de negocio y de Data Science).

# Metodología de Trabajo

## Preparación de los datos

Seleccionar las filas relevantes:

- Cada fila estaría representando un registro
- No todos los registros son relevantes (ya sea por el rango de fechas en el que fue obtenido, por falta de información, errores, etc.)
- Necesidad de estudiar un subconjunto específico

# Metodología de Trabajo

## Preparación de los datos

Seleccionar las columnas relevantes:

- Cada columna representa un tipo de dato específico de un registro
- En base a la información, tomaremos decisiones sobre el uso de atributos
- Pueden existir restricciones
- Puede surgir la necesidad de crear nuevos valores (nuevas columnas) a partir de las actuales (por ejemplo, si queremos tener una columna con el valor de la edad)

# Metodología de Trabajo

## Calidad de los datos

El siguiente paso es analizar la calidad de los datos que tenemos. No es el objetivo hacer un análisis profundo, pero si identificar problemas comunes que puedan complicar el análisis.

En esta etapa se busca identificar problemas con los datos y tomar decisiones, ya sea eliminando registros o agregando valores donde se encuentran los problemas.

# Metodología de Trabajo

## Calidad de los datos

- Datos perdidos
  - Valores vacíos o “sin respuesta” (nulos)
  - Requiere decidir si los registros que tienen algún valor en esta situación se deben eliminar o sustituir por un valor (usualmente valor medio o moda)
- Errores en datos
  - Errores tipográficos por acción humana (errores de ortografía, letras intercambiadas)
  - Requiere decidir si los registros que tienen algún valor en esta situación se deben eliminar o sustituir por el valor que se considera “correcto”
- Error de mediciones
  - Sistema de medidas incongruente
  - En general implica eliminar estos registros

# Metodología de Trabajo

## Calidad de los datos

- Incoherencias en codificación
  - Diferente representación para valores iguales (representación de un departamento en diferentes formas, MVD – mont – Montevideo – etc )
  - Requiere trabajar sobre el dataset re clasificando los datos
    - Conviene siempre mantener el dataset original, dentro de lo posible trabajar sobre una copia
    - Pueden quedar registros sin poder ser clasificados, requiere tomar la decision si eliminarlos o no
- Metadatos erróneos
  - Columnas que no tienen el tipo de dato adecuado
  - Requiere trabajar sobre el dataset modificando el tipo de dato de la columna
  - Puede implicar re clasificar datos, por encontrarse datos inconsistentes con el nuevo tipo de dato

# Metodología de Trabajo

## Data Profiling

Data Profiling es un proceso para examinar los datos disponibles de una fuente de datos, y recopilar estadísticas o un resumen informativo sobre esos datos, para analizar la calidad de los datos.

Se utilizan métodos de estadísticas descriptivas como mínimo, máximo, media, moda, percentil, desviación estándar, frecuencia, variación, agregados como count y suma, e información de metadatos adicional obtenida durante el perfil de datos como tipo de datos, longitud, valores discretos, unicidad, ocurrencia de valores nulos, patrones de cadenas típicos y reconocimiento de tipo abstracto.

Los metadatos se pueden utilizar para descubrir problemas como valores ilegales, errores ortográficos, valores perdidos, representación de valores variables y duplicados.



# Metodología de Trabajo

## Modelado y evaluación

Implica generar los modelos de aprendizaje automático para poder responder las necesidades del negocio.

Se deben elegir los modelos adecuados en base a los datos que se tienen, así como las respuestas que se quieren obtener.

Este proceso es iterativo, se generan modelos, se evalúan los resultados y se ajustan los parámetros de los modelos hasta llegar a las condiciones aceptables que se requieren.

Puede implicar llegar a la conclusión que no es posible obtener modelos con los datos que se tienen, o que no es posible mejorar llegar a mejores niveles para los datos obtenidos.

# Metodología de Trabajo

## **Despliegue de Datos**

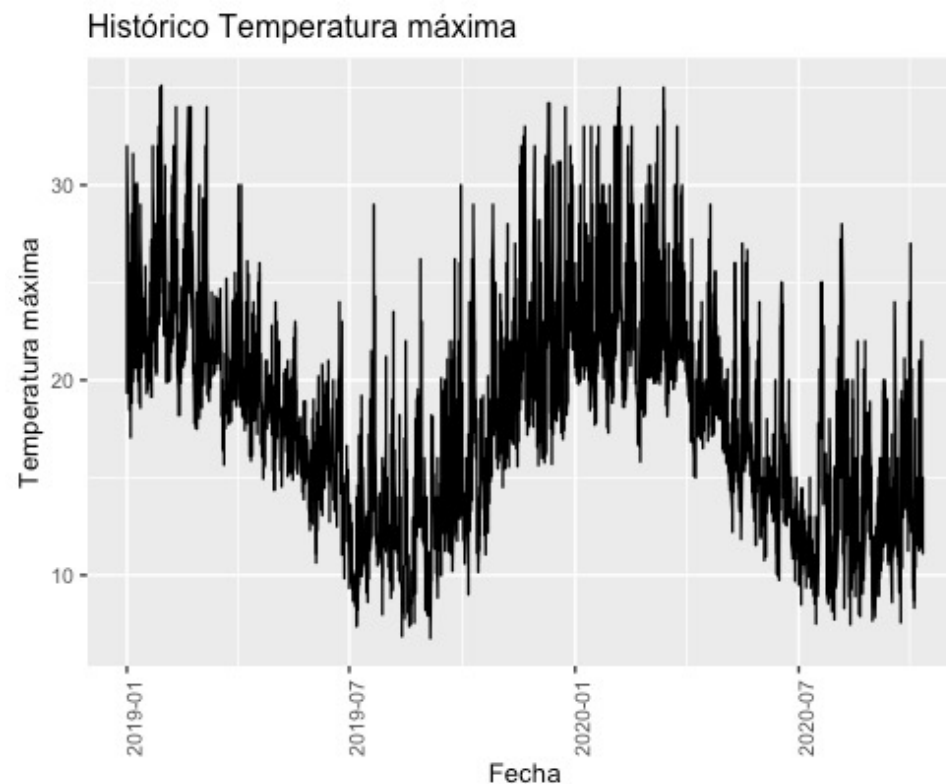
En esta etapa es cuando se presentan los datos, ya sea a través de gráficos, dashboards o disponibilizando los resultados para ser consumidos por otros sistemas.

Para el caso de generar gráficos, es importante elegir gráficos adecuados que permitan representar la realidad a presentar.

# Metodología de Trabajo

## Algunas Formas de Visualización de Datos

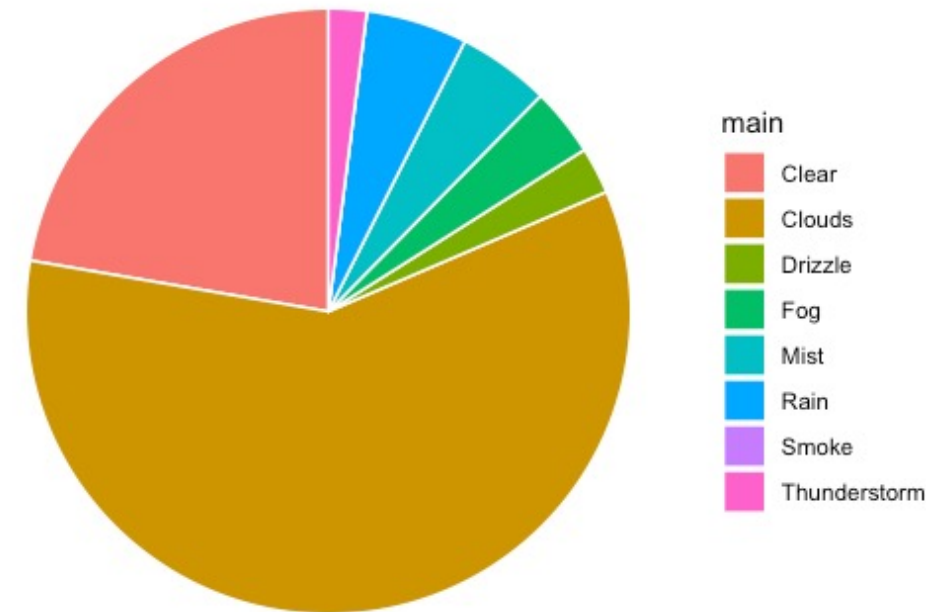
*Gráficos de Línea:* Es la representación gráfica más simple y utilizada para mostrar la evolución de una variable cuantitativa en función de otra variable



# Metodología de Trabajo

## Algunas Formas de Visualización de Datos

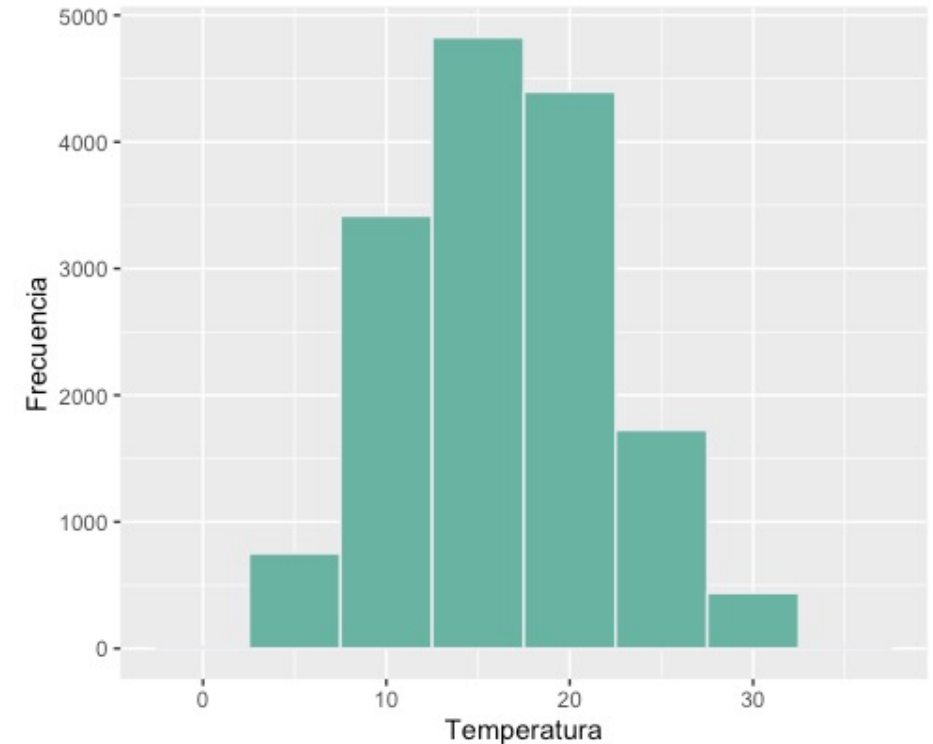
*Gráficos de Torta:* Permite analizar de forma básica la composición porcentual de una variable categórica



# Metodología de Trabajo

## Algunas Formas de Visualización de Datos

*Histograma:* Un histograma es una representación de una variable numérica en forma de barras, donde se representa la frecuencia (repeticiones) en el rango graficado



# Metodología de Trabajo

## Algunas Formas de Visualización de Datos

*Nube de Palabras (Word Cloud):* Muestra los diferentes valores de una variable categórica, en particular una variable de texto libre, donde para cada palabra se agrega una tercera dimensión que es su tamaño.

Usualmente se cuenta la cantidad de apariciones de la palabra dentro del dataset, y ese valor es utilizado como peso de la categoría, para determinar el tamaño.





# Ciencia de Datos

MDataSc(c). Ing. Natalia Botto Pérez

# Herramientas



- R es un entorno y lenguaje de programación para procesamiento estadístico de datos y generar visualizaciones
- Software Libre
- Funciona en Windows, Linux y MacOS
- El IDE para trabajar se llama RStudio
- <https://www.r-project.org/>



# Herramientas



- Python es un lenguaje de programacion
- Software Libre
- Funciona en Windows, Linux y MacOS
- Existen varios IDEs para poder trabajar con Python
- <https://www.python.org/>

# Herramientas



- Anaconda es una distribución de R y Python, utilizada principalmente para DataScience y Machine Learning
- Software Libre
- Funciona en Windows, Linux y MacOS
- <https://www.anaconda.com/products/individual>

# Herramientas



IBM Watson Studio Cloud

- Watson Studio es una plataforma de Data Science
- IBM
- <https://www.ibm.com/cloud/watson-studio>



# Herramientas

- Power BI es una herramienta de análisis de datos
- Microsoft
- Tiene versión de escritorio gratuita para Windows, en la nube tiene costo
- <https://powerbi.microsoft.com/es-es/>



# Ciencia de Datos

MDataSc(c). Ing. Natalia Botto Pérez

# Aplicaciones

## **Filtros de SPAM para mail**

Algoritmos de IA para generar filtros de antispam mas robustos y confiables.

El funcionamiento de estos algoritmos es entrenarlos en base a un dataset histórico, en el cual el algoritmo clasificara algunos como SPAM y otros no. En caso que se identifique algún correo clasificado incorrectamente, esto se indica para poder mejorar el reconocimiento.

Se basan en encontrar ciertos patrones en los correos (anuncios, descuentos, etc.), errores en palabras o en el idioma y otros.

# Aplicaciones

## **Autocompletar**

Este tipo de funcionalidad permite completar palabras o frases en función de lo que el usuario va escribiendo.

Se utilizan herramientas de IA que sugieren como completar la escritura del usuario en algunos casos en base a frases mas utilizadas o las preferencias del usuario.

# Aplicaciones

## Autocorrectores

El autocorrector permite reemplazar texto a medida que uno escribe, utilizando herramientas de IA permite que las sugerencias sean mas acertadas en base al contexto y las características del usuario.

El proceso de autocorrección implica cuatro pasos principales, identificar una palabra mal escrita, luego encontrar las cadenas mientras se calcula la distancia de edición mínima de cada una de ellas, filtrando los posibles candidatos para la selección de la palabra correcta. Y finalmente, calcular las probabilidades de la palabra para pronosticar la mejor predicción posible para la palabra en particular.



# Aplicaciones

## Chatbots

Los chatbots son utilizados en sitios web, whatsapp para interactuar con los usuarios humanos que llegan a los sitios específicos. El objetivo es generar una interacción eficaz y brindar información con instrucciones detalladas y guías con respuestas espontáneas.

Existen chatbots que interpretan los textos escritos por los usuarios o generan mejores predicciones utilizando herramientas de IA para generar las posibles respuestas.

# Aplicaciones

## **Análisis de riesgo**

En el caso de las finanzas ya se vienen utilizando herramientas para poder resolver problemas financieros y económicos complejos en tiempo real, como las predicciones del mercado de valores. El uso de IA permite obtener predicciones mucho mas precisas.

También estas herramientas se utilizan para detección de fraude, análisis de riesgo en prestamos y otros.

El beneficio principal es por la posibilidad de analizar múltiples fuentes de datos y la identificación de variables (o interacción entre variables) que afectan los resultados que visualmente por una persona serían imposible de detectar.

# Aplicaciones

## **Medicina en general**

IA y ciencia de datos se utiliza para múltiples aplicaciones, principalmente buscando analizar resultados de exámenes médicos, análisis de imágenes (radiografías, tomografías, etc.) para poder clasificar a los pacientes de forma de identificar enfermedades específicas, tumores u otros.

# Aplicaciones

## Robótica

Esta es un área en desarrollo por las posibilidades que implica.

La integración de proyectos de ciencia de datos junto con robots tiene un enorme potencial para hacer tareas manuales, a nivel de fabricación de productos buscando mejorar el rendimiento a nivel humano en muchas tareas preprogramadas. Los avances en IoT y la comunidad también son muy beneficiosos para la integración de la IA en robótica para desarrollar dispositivos inteligentes y efectivos.

# Aplicaciones

## Deportes

En muchos deportes se utiliza data science para la toma de decisiones en los deportes, ya sea para definir estrategias, mejorar el rendimiento de los deportistas y prevenir lesiones.

# Aplicaciones

## Segmentación de Clientes

Ya sea que se utilice en redes sociales o para una base de clientes, es posible utilizar ciencia de datos e inteligencia artificial para poder segmentar los clientes a quienes queremos enviar ciertas publicidades.

El objetivo es no solamente segmentar en base a características básicas (ubicación, edad, etc.) sino también en base a sus gustos, preferencias o incluso sus comentarios y permitiendo cruzar todas las variables disponibles.

# Aplicaciones

## **Autos autónomos**

En este caso el uso de aprendizaje profundo a través del análisis de millones de datos, es lo que esta desarrollado esta tecnología. Los resultados que se generan deben ser probados en ambientes seguros.

# Aplicaciones

## Reconocimiento facial

Para nosotros es muy fácil reconocer cualquier imagen y reconocer caras, pero para una computadora, las imágenes son solo una serie de valores numéricos y es por eso que utiliza algoritmos de procesamiento de imágenes para buscar patrones en imágenes digitales (videos, gráficos o imágenes fijas); sobretodo porque la cara de una persona no tiene una única imagen, sino múltiples (en base a expresiones).

Lo que realizan estas herramientas es identificar puntos en el rostro humano para poder comparar con una base de datos de personas.

Ha generado múltiples debates en base a los casos de sesgos que se generaron en todo el mundo.



# Aplicaciones

## **Análisis Sentimental**

Es una tecnología que permite clasificación de sentimientos, extracción de opiniones y análisis de emociones.

A partir de las palabras utilizadas se clasifican en positivas, negativas o neutrales.

Con esta clasificación se pueden tomar decisiones, esto es muy utilizado en redes sociales.

# Aplicaciones

## **CLUE**

Es una aplicación que utiliza data science para analizar y predecir cuales serán los ciclos menstruales y reproductivos de las usuarias, a partir de los datos históricos y de otras variables de estado general (estados de humor, condición del pelo y otros)

# Aplicaciones

## **Calculo de rutas y alertas de trafico – Google Maps / Waze**

Utilizando datos históricos y factores del estado actual del transito, dada la ubicación del conductor, velocidad promedio del vehículo, día de la semana, hora del día, si existe alguna ocasión especial; con todos estos datos se generan las recomendaciones de rutas.

# Aplicaciones

## **UPS – automatización de entrega de paquetes**

UPS utiliza la ciencia de datos para optimizar el transporte de paquetes desde la entrega hasta la entrega. Su última plataforma para hacerlo, Network Planning Tools (NPT), incorpora aprendizaje automático e inteligencia artificial para resolver como la forma en que los paquetes deben redirigirse para evitar el mal tiempo o los cuellos de botella del servicio, entre otros.

Según un pronóstico de la compañía, la plataforma podría ahorrarle a UPS entre 100 y 200 millones de dólares para 2020.

# Aplicaciones

## **Netflix – sugerencias de contenidos**

Netflix realiza sugerencias en base a

- Que búsquedas se realizan,
- Características de las elecciones previas (titulo, genero, categoría, actores, año y otros)
- Hora del día que se visualiza el contenido
- Tipos de dispositivos que se utilizan
- Durante cuanto tiempo se ven los contenidos

En base a todos estos datos, que se ingresan como datos de entrada al algoritmo y se generan las recomendaciones. No se incluyen variables de ubicación geográfica o edad al definir las sugerencias.

# Aplicaciones

## **Lymph Node Assistant (LYNA) – Google AI**

LYNA es una herramienta de aprendizaje profundo que permite identificar cáncer de mama en estado avanzado a partir del análisis de imágenes.

LYNA examinó imágenes de las células de un paciente y se entrenó para reconocer las características de los tumores con la ayuda de diapositivas patológicas que se usaron como conjuntos de datos. Lo que podría ser de mayor interés es que LYNA pudo reducir el tiempo dedicado a revisar una diapositiva de dos minutos a solo uno.

### Referencia

# Aplicaciones

## **Deep Nostalgia**

Herramienta que anima fotos.

[Referencia](#)

# Aplicaciones

**This person does not exist**

Creación de imágenes sintéticas a partir de rasgos físicos de otras personas.

[Referencia](#)



# Aplicaciones

## **HERTA security (h&o tecnología)**

Es una solución de video vigilancia, realiza reconocimiento facial de las personas que son filmadas por sus sistemas. Para ello usa herramientas de aprendizaje profundo. Requiere tener una base de datos para comparar las imágenes identificadas.

### [Referencia](#)

# Aplicaciones

## **Dictation**

Permite identificar lo que se dice y lo escribe en pantalla. Se pueden agregar signos de puntuación, emoticones con comandos de voz.

## Referencia

# Aplicaciones

## **Foot Levelers – automatización de diseños de plantillas**

Utilizando la base de datos existente de diseño de plantillas y las decisiones de los técnicos que se tomaron previamente, diseñaron una solución utilizando varias técnicas de IA, algoritmos tradicionales de machine learning y soluciones de computer vision.

En vez de realizar las medidas de los pies de forma manual, se analizan imágenes y se generan los diseños, automatizando estas tareas manuales.

### [Referencia](#)

# Aplicaciones

## **Mercado Libre – clasificación de productos**

Utilizando herramientas de aprendizaje supervisado, en base a la información indicada por el usuario, se realiza una sugerencia de la categoría que debería integrar el producto.

### [Referencia](#)

# Aplicaciones

## **Instituto Nacional de Estadística**

Análisis y visualización de grandes volúmenes de datos.

[Referencia](#)

# Aplicaciones

## **Detección de especies de murciélagos en Uruguay**

En este trabajo se generó un algoritmo de detección de especies de murciélagos, a partir de variables acústicas.

### Referencia



# Ciencia de Datos

MDataSc(c). Ing. Natalia Botto Pérez

# Datos estructurados

Los datos estructurados son aquellos que están altamente organizados y formateados de tal manera que se pueden almacenar y buscar fácilmente en bases de datos relacionales (RDBMS).



# Datos estructurados

Estos datos son almacenados en el modelo relacional, presentado por Edward Codd en 1970.

El Modelo Relacional se basa en representar los datos por medio de **tablas relacionadas**, cuyas **filas** representan los registros y las **columnas** las variables.

Esto garantiza la integridad de los datos (controlando los tipos de datos que se almacenan), evita la redundancia de datos y permite normalizar el modelo de datos.

# Datos estructurados

Cada dato se le asocia un nombre, un valor y un tipo de dato:

- Texto (string)
- Numérico
  - Entero (int)
  - Decimal (float, double)
- Fecha hora (date)
- otros

idEstudiantes	Nombre	Apellido	FechaNacimiento
1	Natalia	Botto	1982-01-28 00:00:00
NULL	NULL	NULL	NULL

# Datos estructurados

Podemos pensar en la analogía de una **planilla electrónica**, donde también representamos los datos en una estructura de hojas (que serian nuestras tablas), donde en las filas representamos los registros de los datos, y en las columnas a que corresponde cada valor del registro.

La mayor diferencia entre una planilla electrónica y una base de datos es que las diferentes hojas de una planilla o los diferentes archivos **no están relacionados entre si**, a su vez, no hay un control sobre los tipos de datos asociados a cada una de esas columnas.

# Datos estructurados

Las bases de datos relacionales se caracterizan por organizarse en **tablas**, tienen un esquema determinado que define cómo son las tablas en las que se almacenan los datos, qué tipo de campos tienen y cómo se relacionan entre ellas

- Cada tabla indica que columnas almacenara internamente
- Cada columna de cada tabla tendrá asociado un tipo de dato
- Cada fila identifica un registro
- Se podrán indicar restricciones para los valores a almacenar en cada columna

# Datos estructurados

- Todas las tablas están relacionadas entre si
- En general, cada tabla tiene indicado una columna que es el identificador único de cada registro
  - No puede repetirse en la tabla
- Esta columna se utiliza para relacionar tablas
  - Si quiero indicar que un registro de una tabla esta relacionado con otro, no agrego nuevamente todo los datos, sino que agrego su clave primaria

# Datos estructurados

Al definir la estructura de mi tabla, algunas columnas se definen como alguna de estas claves.

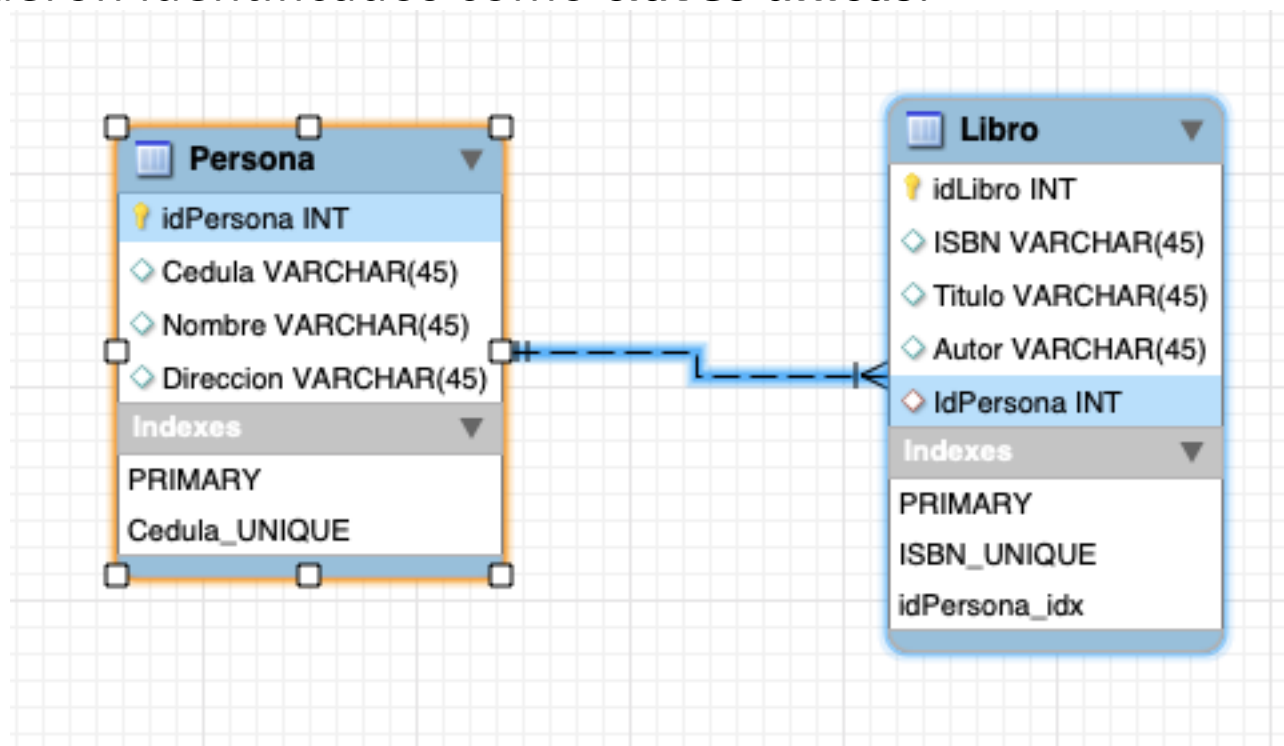
**CLAVES PRIMARIAS:** son identificadores únicos de los registros de una tabla de una base de datos. Todas las tablas deberían tener una clave primaria, ya que permite asegurarse no ingresar repetidos los registros y relacionar los registros con otras tablas. En muchas tablas suelen ser registros numéricos.

**CLAVES FORANEAS O SECUNDARIAS:** cuando quiero relacionar dos tablas entre si, debo definir en una de ellas una columna como clave foránea, de forma de generar la relación entre ambas. Una columna se identifica como clave foránea cuando los valores que va a tomar corresponden a los valores de la columna que es clave primaria de la otra tabla.

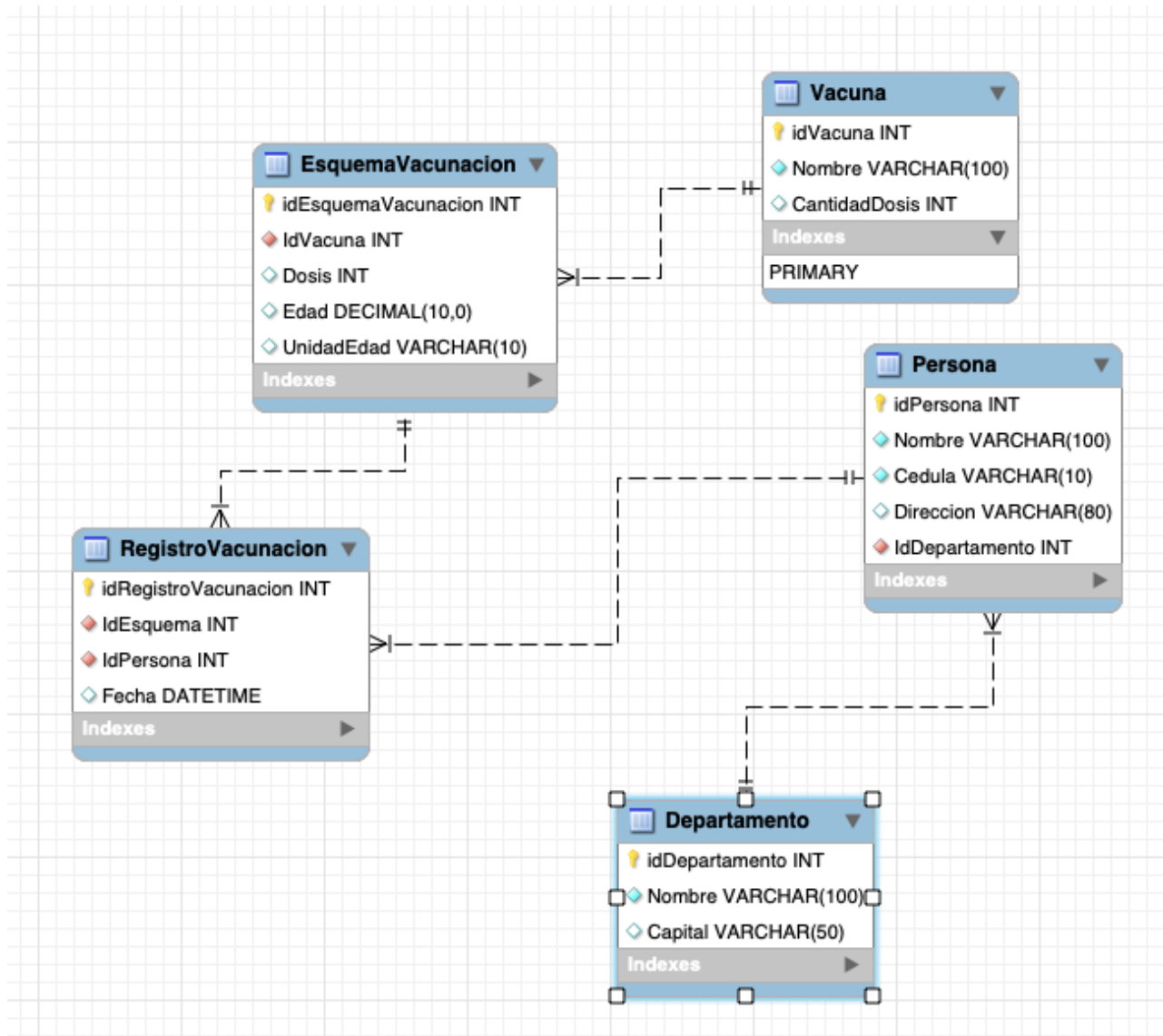
**CLAVES UNICAS:** las claves únicas son columnas en una tabla que no son claves primarias, pero los valores de esa columna no se pueden repetir

# Datos estructurados

En la imagen se puede ver las entidades Persona y Libro convertidas en **tablas**, donde se puede ver que se le agrego una **clave primaria** a Persona (idPersona), una **clave primaria** a Libro (idLibro) y en la tabla Libros podemos encontrar idPersona como **clave foránea** (es lo que permite relacionar Persona con Libro). Las columnas Cedula en la tabla Persona e ISBN de la tabla Libro fueron identificados como **claves únicas**.



# Datos estructurados





# Datos estructurados

Estos datos se gestionan mediante un tipo de lenguaje de programación estructurado, conocido como **SQL (Structured Query Language)** diseñado precisamente, para administrar y recuperar información de los sistemas de gestión de bases de datos relacionales.

# Datos estructurados

## CSV (comma-separated values)

Son un formato de documentos abierto muy sencillo que permite representar datos en forma de tabla.

Cada dato viene separado por un separador (puede ser una coma, punto y coma u otro separador).

Al abrir este tipo de archivos muchas veces se debe indicar el separador y como se indican los números decimales (punto, coma).

Son muy utilizados como entrada para proyectos de DataScience.

# Datos semi estructurados

Los datos semi estructurados no tienen un esquema previamente definido. No representan una estructura de tablas/columnas/filas sino que se organizan mediante etiquetas, para agruparlos en forma de jerarquías.

Este tipo de datos son los que se almacenan en las bases de datos No Relacionales.

Para proyectos de DataScience es posible leerlos con herramientas usuales, pero suele requerirse “aplanarlos” para poder analizarlos.

# Datos semi estructurados

## XML (eXtensible Markup Language)

Es un formato en el cual los datos se representan con etiquetas, siguiendo un [estandar](#).

Una etiqueta consiste en un nombre y un valor, y se representan en formato de árbol.

Algunos dataset se obtienen en este formato, pero requiere que sean procesados para poder llevarlos al formato de un dataframe.

# Datos semi estructurados

## JSON (JavaScript Object Notation)

Es un formato de texto sencillo para el intercambio de datos.

Similar a XML, estructura los datos con etiquetas, donde cada etiqueta tiene un nombre y un valor asociado.

Se utiliza principalmente para el intercambio de datos vía web services.

También varias bases de datos no relacionales (de tipo documental) tomaron como base el formato nativo JSON para generar sus propias estructuras)

# Datos no estructurados

Los datos no estructurados son aquellos que no tienen una estructura previamente definida que facilite su acceso y procesamiento.

Suponen el mayor volumen de datos que se generan, algunos ejemplos son

- Documentos de texto PDF, Word, etc
- Imágenes
- Videos
- Audios
- otros



# Ciencia de Datos

MDataSc(c). Ing. Natalia Botto Pérez

# Herramientas para Big Data

El manejo de grandes volúmenes de datos, y la gran variedad de datos a almacenar, generaron que aparecieran nuevas herramientas para el almacenamiento de datos. Estos modelos buscan flexibilizar algunos de los requerimientos de los sistemas de bases de datos relacionales, buscando escalabilidad y capacidad de recuperación.



# Herramientas para Big Data

Por ejemplo, las bases de datos no relacionales se basan en los siguientes conceptos:

- Disponibilidad todo el tiempo (**Basic Availability**): la base de datos está disponible y funcional todo el tiempo
- Estado flexible (**Soft – State**): Las bases de datos no tienen que ser consistentes con la escritura, ni las diferentes réplicas deben ser mutuamente consistentes todo el tiempo
- Consistencia eventual (**Eventually consistent**): En un preciso momento puede ser que los datos no sean consistentes, pero a lo largo del tiempo si lo será

# Herramientas para Big Data

## MODELO CLAVE - VALOR

Las bases de datos que utilizan el modelo clave-valor son el tipo más básico de base de datos no relacional. Cada elemento en la base de datos se almacena como el nombre de un atributo, o clave, junto con su valor.

El valor, sin embargo, es completamente opaco para el sistema; los datos solo pueden ser consultados por la clave. Este modelo puede ser útil para representar datos polimórficos y no estructurados, ya que la base de datos no aplica un conjunto esquema a través de pares clave-valor.

El atractivo de estos sistemas es su rendimiento y escalabilidad, que pueden ser optimizados debido a lo simple de la estructura de los datos y la opacidad de los datos en sí.

# Herramientas para Big Data

## MODELO BASADO EN COLUMNAS

Si pensamos en herramientas para la toma de decisiones, como vistas, cubos y esquema en estrella (DW) son utilizados para acelerar el proceso de búsqueda y recuperación de datos, y ayudar al usuario a navegar una serie compleja de uniones, pero los datos se almacenan para que las filas se puedan leer de la manera más eficiente posible, no las columnas.

Buscar y leer datos desde el disco es uno de los aspectos más lentos de cualquier sistema de base de datos. El corazón del problema es que el DBMS tiene que leer cada fila, descartando información no deseada (o al menos moviendo la posición de lectura para que omita datos no deseados) para llegar a los datos en la columna deseada.

# Herramientas para Big Data

## MODELO BASADO EN COLUMNAS

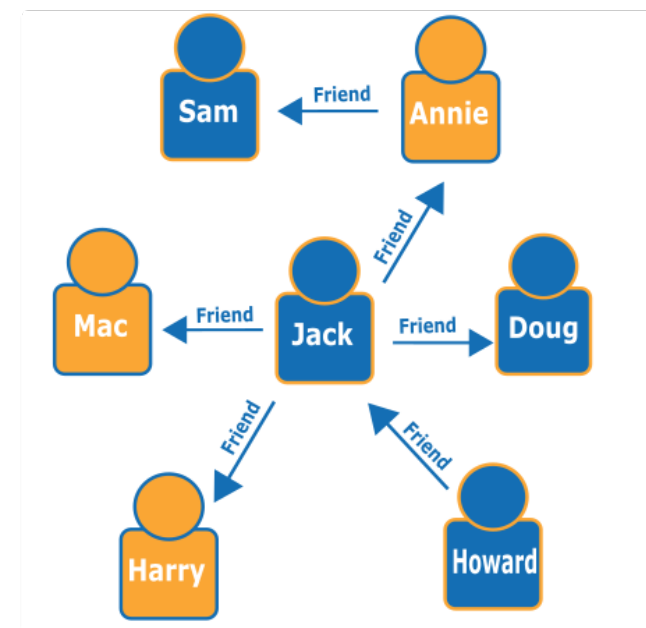
La esencia de este modelo es buscar una estructura que permita que los datos sean almacenados en disco de forma consecutiva, optimizando la búsqueda de datos, y esto se resuelve si los datos son almacenados en columnas, en vez de filas. Por lo tanto, el proceso de lectura es muy simple y solo hay que preocuparse por los punteros de inicio y lectura y no saltar a diferentes segmentos de disco para saltar sobre los datos no deseados.

# Herramientas para Big Data

## MODELO GRÁFICO

El modelo lógico de una base de datos de gráficos es conceptualmente bastante simple. Los nodos (o "vértices"), que serían los diferentes objetos, están conectados entre sí por relaciones (o "bordes"). Tanto los nodos como las relaciones pueden tener propiedades: conjuntos de pares nombre: valor. Son presentadas como una alternativa importante cuando el análisis de las relaciones entre objetos tiene tanta importancia como los propios objetos.

La teoría de gráficos proporciona notación matemática para definir o eliminar nodos o relaciones del gráfico, y para realizar operaciones para encontrar nodos adyacentes. Estas operaciones primitivas se pueden usar para realizar un recorrido transversal de gráficos para explorar la red.



# Herramientas para Big Data

## MODELO GRÁFICO

Utiliza ciertos lenguajes para realizar las consultas llamados SPARQL, Cypher y Gremlin, los cuales están optimizados para operaciones de cruce de gráficos, ofreciendo una sintaxis que nos permite recorrer el gráfico sin requerir las uniones recursivas necesarias para las bases de datos relacionales.

Las bases de datos de gráficos son útiles en los casos en que las relaciones transversales son fundamentales para la aplicación, como la navegación por las conexiones de redes sociales, las topologías de red o las cadenas de suministro.

# Herramientas para Big Data

## MODELO DOCUMENTAL

Las bases de datos basadas en el modelo documental es una base de datos no relacional que almacena los datos como documentos estructurados, generalmente en XML (eXtensible Markup Language) o formatos JSON (JavaScript Object Notation).

Modelar los datos como documentos es similar a como se manejan los datos en la programación orientada a objetos: cada documento es efectivamente un objeto.

# Herramientas para Big Data

## MODELO DOCUMENTAL

Los documentos contienen uno o más campos, donde cada campo contiene un valor escrito, como texto, fecha, binario, valor decimal o matriz.

En vez de representar los registros en varias columnas y en tablas relacionadas, cada registro y sus datos relacionados se almacenan en un solo documento jerárquico.

Una particularidad de las bases de datos documentales es que el esquema es dinámico, cada documento puede contener diferentes campos, no existe una estructura fija para los documentos.



# Herramientas para Big Data

## MODELO DOCUMENTAL

Esta la flexibilidad puede ser particularmente útil para modelar datos no estructurados y polimórficos. También lo hace más fácil para desarrollar una aplicación durante su ciclo de vida, como agregar nuevos campos.

Además, algunas bases de datos de documentos permiten utilizar sentencias similares a las de las bases de datos relacionales para acceder a los registros. En particular, los datos pueden ser consultados basado en cualquier combinación de campos en un documento, con índices secundarios que proporcionan rutas de acceso eficientes a admite casi cualquier patrón de consulta.

# Herramientas para Big Data

## MODELO DOCUMENTAL

Este modelo de base de datos es de uso general, útil para una amplia variedad de aplicaciones debido a la flexibilidad del modelo de datos, la capacidad de consultar en cualquier campo y el mapeo natural del modelo de datos del documento a los objetos en los lenguajes de programación modernos.

# Herramientas para Big Data

## MODELO DOCUMENTAL

```
{
  "glossary": {
    "title": "example glossary",
    "GlossDiv": {
      "title": "S",
      "GlossList": {
        "GlossEntry": {
          "ID": "SGML",
          "SortAs": "SGML",
          "GlossTerm": "Standard Generalized Markup Language",
          "Acronym": "SGML",
          "Abbrev": "ISO 8879:1986",
          "GlossDef": {
            "para": "A meta-markup language, used to create markup languages such as DocBook.",
            "GlossSeeAlso": ["GML", "XML"]
          },
          "GlossSee": "markup"
        }
      }
    }
  }
}
```

# Herramientas para Big Data

## PERSISTENCIA POLÍGLOTA

La persistencia políglota es un enfoque que implica que los datos, en vez de almacenarse en una única base de datos, se almacenan en varias bases.

Lo que se busca es seleccionar el modelo que mejor resuelva el almacenamiento de cada conjunto de datos, dependiendo de sus características.

Esto tiene la desventaja de que requiere una gran cantidad de conversiones de datos para que todos los formatos sean los mismos, al trabajar con ellos.

# Herramientas para Big Data

## BASES DE DATOS MULTIMODELO

Son las bases de datos que soportan varios modelos de almacenamiento para los datos (documentales, grafos, relacional, clave valor) dentro de su estructura.

Estas bases buscan ofrecer las ventajas del modelado de datos de la persistencia políglota (polyglot persistence), sin sus desventajas. Buscan evitar el problema de la coherencia entre los datos (por las conversiones entre datos) y la complejidad operativa, ya que se utiliza un único almacén de datos

# Herramientas para Big Data

## MongoDB

- Es una base de datos NoSQL orientada a documentos
- De código abierto
- De uso gratuito, tiene versiones (motor y cliente) desktop y cloud
- De uso sencillo, fácil de aprender y de instalar
- Brinda soporte para múltiples tecnologías y plataformas
- Confiable y de bajo costo

# Herramientas para Big Data



## [Amazon DynamoDB](#)

- Es una base de datos NoSQL clave valor
- Se ofrece como parte del portafolio de servicios de AWS (con almacenamiento disponible para accesos gratuitos a AWS)
- Rápido y flexible

# Herramientas para Big Data



## Apache Couch DB

- Es una base de datos NoSQL orientada a documentos
- De código abierto
- Múltiple plataforma
- Facilidad de uso y una arquitectura escalable





# Herramientas para Big Data

## Apache Cassandra

- Es una base de datos NoSQL orientada a columnas
- De código abierto y gratuito
- Construido para gestionar grandes volúmenes de datos distribuidos en numerosos servidores básicos, ofreciendo alta disponibilidad
- Los datos se replican automáticamente en múltiples nodos para tolerancia a fallas
- Es una de las mejores herramientas de big data que es más adecuada para aplicaciones que no pueden permitirse perder datos, incluso cuando todo un centro de datos está inactivo
- Utiliza su propio lenguaje CQL (Cassandra Structure Language)



# Herramientas para Big Data

## [neo4j](#)

- Es una base de datos NoSQL de tipo grafico
- Open Source
- Tiene versiones Community y Enterprise

# Herramientas para Big Data



## Apache Hadoop

- Es un entorno de trabajo para programar aplicaciones distribuidas
- Maneja un sistema de archivos en cluster para procesar conjuntos de datos de big data, mediante el modelo de programación MapReduce
- Open Source
- Apache Software Foundation
- La fortaleza principal de Hadoop es su HDFS (Hadoop Distributed File System) que tiene la capacidad de almacenar todo tipo de datos: video, imágenes, JSON, XML y texto sin formato en el mismo sistema de archivos



# Ciencia de Datos

MDataSc(c). Ing. Natalia Botto Pérez

# SQL

SQL (Structured Query Language) es un lenguaje, originalmente basado en algebra y calculo relacional, que permite definir, manipular y controlar los datos de una base de datos relacional. Permite realizar todas las funciones CRUD sobre una base de datos (creacion, lectura, modificacion y eliminacion de esquemas, estructuras y datos).

Un punto muy importante es que en 1986 paso a ser un estandar ANSI y en 1987 un estandar OSI, esto genera que no importa la base de dato que vaya a utilizar, las sentencias que creo en un motor de base de datos relacional, funciona para cualquier otro tipo de motor de base de datos (siempre y cuando se utilicen las sentencias del estandar).

# SENTENCIAS SQL – CREAR TABLA PERSONA

Creo la tabla Persona

Las palabras en negrita son reservadas

```
CREATE TABLE `Persona` (  
  `idPersona` int NOT NULL,  
  `Cedula` varchar(45) DEFAULT NULL,  
  `Nombre` varchar(45) DEFAULT NULL,  
  `Direccion` varchar(45) DEFAULT NULL,  
  PRIMARY KEY (`idPersona`),  
  UNIQUE KEY `Cedula_UNIQUE` (`Cedula`)  
)
```

Le indico una columna de tipo texto que el valor por defecto es vacio (DEFAULT NULL)

Le indico una columna de tipo Entero que no puede tener valores vacios (NOT NULL)

Le indico que columna es clave unica

Le indico que columna corresponde a la clave primaria

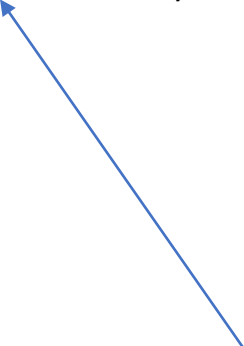
# SENTENCIAS SQL – CREAR TABLA PERSONA

Creo la tabla Libro



Las palabras en negrita son reservadas

```
CREATE TABLE `Libro` (  
  `idLibro` int NOT NULL,  
  `ISBN` varchar(45) DEFAULT NULL,  
  `Titulo` varchar(45) DEFAULT NULL,  
  `Autor` varchar(45) DEFAULT NULL,  
  `IdPersona` int DEFAULT NULL,  
  PRIMARY KEY (`idLibro`),  
  UNIQUE KEY `ISBN_UNIQUE` (`ISBN`),  
  CONSTRAINT `idPersona` FOREIGN KEY (`IdPersona`) REFERENCES `Persona` (`idPersona`)  
)
```



Le indico la clave foranea, indicandole que columna y que tabla se hace referencia

# INSERT

Para ingresar registros en una base de dato relacional, usamos la sentencia INSERT, que tiene las siguiente sintaxis:

**INSERT INTO**

**nombre\_tabla**

**(nombre\_columna1, nombre\_columna2, ... , nombre\_columnaN)**

**VALUES**

**(valor1, valor2, ... , valor N);**



# INSERT

Algunas consideraciones:

- Las palabras en azul son reservadas, y se tienen que incluir en las sentencias
- No es imprescindible que se incluyan los nombres de las columnas, pero en ese caso hay que asegurarse de colocar los valores en el mismo orden que fueron creadas las columnas
- El motor lo único que va a chequear es que el tipo de dato del valor, corresponda con la columna indicada
- Existe una forma de hacer un INSERT en una tabla recuperando registros de otra tabla, utilizando la sentencia SELECT. En ese caso la sentencia queda como la siguiente:

## INSERT INTO

**nombre\_tabla**

## SELECT

nombre\_columna1, nombre\_columna2, ... , nombre\_columnaN

## FROM

tabla1

## WHERE

condición

# SELECT

Para recuperar registros en una base de dato relacional, usamos la sentencia SELECT, que tiene las siguiente sintaxis:

**SELECT**

**nombre\_columna1, nombre\_columna2, ... , nombre\_columnaN**

**FROM**

**tabla1**

**WHERE**

**condición**

# SELECT

Algunas consideraciones:

- Cuando queremos recuperar todas las columnas de una tabla, podemos usar \* y no incluir ningún nombre de columna
- Si luego de la palabra SELECT, incluimos la palabra DISTINCT, nos recuperara los registros que NO traigan valores repetidos
- Al final de la sentencia se puede agregar la opción ORDER BY nombre\_columna, para indicar porque columna queremos ordenar los resultados
- Si queremos recuperar valores de varias columnas, las colocamos separadas por comas, solo tenemos que hacer referencia a que tabla pertenece cada columna de la siguiente forma
  - tabla1.columna1
- La condición que se coloca en el WHERE puede ser:
  - El valor de la columna es igual a cierto valor (utilizamos =)
  - El valor de la columna es mayor, menor, mayor o igual o menor o igual a cierto valor (>, <, >=, <=), solo para columnas numéricas o fecha
  - El valor de la columna contiene ciertos caracteres (LIKE), solo para columnas texto
  - El valor de la columna comienza o termina con cierto carácter (LIKE "%a" o LIKE "a%")
- Si hay varias condiciones WHERE para utilizar, usamos el operador AND si se deben cumplir todas las condiciones, o el operador OR si se deben cumplir alguna de las condiciones

# UPDATE

Para modificar registros en una base de dato relacional, usamos la sentencia UPDATE, que tiene las siguiente sintaxis:

**UPDATE**

**tabla1**

**SET**

**columna1 = valor1,**

**columna2 = valor2,**

**...,**

**columnaN = valorN**

**WHERE**

**condición**

# UPDATE

Algunas consideraciones:

- Los registros que se modificaran serán todos aquellos registros de la tabla que cumplan con la condición, por eso a veces es preferible primero hacer un SELECT y verificar que efectivamente sean los registros que queremos modificar (ya que luego que se modifican, no hay rollback)
- La actualización de datos de registros se realiza de a una tabla por vez
- La actualización de registros comprobara que no se rompen las reglas de integridad referencial, o sea, que no si hay columnas que están relacionadas con otra a través de claves foráneas, no estar agregando un valor que rompa las reglas
- También se validara que los tipos de datos sean los correctos

# DELETE

Para borrar registros en una base de dato relacional, usamos la sentencia DELETE, que tiene las siguiente sintaxis:

**DELETE FROM**

**tabla1**

**WHERE**

**condición**

# DELETE

Algunas consideraciones:

- Similar a las sentencias UPDATE, los registros que se borrarán serán todos aquellos registros de la tabla que cumplan con la condición, por eso a veces es preferible primero hacer un SELECT y verificar que efectivamente sean los registros que queremos modificar (ya que luego que se borran, no hay rollback)
- La eliminación de registros comprobada que no se rompen las reglas de integridad referencial, o sea, que no si hay columnas que están relacionadas con otra a través de claves foráneas, no estar agregando un valor que rompa las reglas. Esto quiere decir que si quiero borrar un registro que tiene referenciado ese valor en otras tablas, no me lo va a permitir

# JOINS

Al realizar consultas, en muchas situaciones es necesario vincular tablas. Es posible realizar consultas, SELECT, y en el FROM incluir todas las tablas a las cuales vamos a consultar y en el WHERE colocar las condiciones de vinculación entre las tablas.

## SELECT

**tabla1. columna1, tabla2.columna1, tabla1.columna2, ...**

## FROM

**tabla1, tabla2**

## WHERE

**condición**

## AND

**tabla1.id = tabla2.id**



# JOINS

Dentro de las sentencias SQL tenemos los JOINS que nos permiten resolver estas situaciones directamente.

## SELECT

**tabla1. columna1, tabla2.columna1, tabla1.columna2, ...**

## FROM

**tabla1**

## (LEFT – RIGHT – INNER) JOIN

**tabla2 ON tabla1.id = tabla2.id**

## WHERE

**condición**

# JOINS

Los JOINS se pueden pensar como interceptar conjuntos, según ciertas condiciones, y según el tipo de JOIN que realice, es la respuesta que voy a tener.

Existen varios tipos de JOINS:

- INNER JOIN
- LEFT JOIN
- RIGHT JOIN
- FULL JOIN

# INNER JOIN

Permite combinar varias tablas y solo devuelve los registros que se encuentran en todas las tablas a la vez.

**SELECT**

**tabla1. columna1, tabla2.columna1, tabla1.columna2, ...**

**FROM**

**tabla1**

**INNER JOIN**

**tabla2 ON tabla1.id = tabla2.id**

**WHERE**

**condición**

# LEFT JOIN

El left join devuelve todos los resultados que coincidan en la primera tabla, con los datos que tenga de la segunda. En el caso de que falte algún dato, devolverá un valor null en lugar del dato, pero seguiremos teniendo el valor de la primera tabla.

**SELECT**

**tabla1. columna1, tabla2.columna1, tabla1.columna2, ...**

**FROM**

**tabla1**

**LEFT JOIN**

**tabla2 ON tabla1.id = tabla2.id**

**WHERE**

**condición**

# RIGHT JOIN

El right join es igual al left join, salvo que devuelve los valores de la tabla relacionada.

**SELECT**

**tabla1. columna1, tabla2.columna1, tabla1.columna2, ...**

**FROM**

**tabla1**

**RIGHT JOIN**

**tabla2 ON tabla1.id = tabla2.id**

**WHERE**

**condición**

# OUTER JOIN

En este caso se devuelven todos los registros de ambas tablas.  
MySQL NO soporta los OUTER JOIN, aunque muchos motores si lo aceptan.

**SELECT**

**tabla1. columna1, tabla2.columna1, tabla1.columna2, ...**

**FROM**

**tabla1**

**OUTER JOIN**

**tabla2 ON tabla1.id = tabla2.id**

**WHERE**

**condición**

# OUTER JOIN

En MySQL, el OUTER JOIN lo hacemos con un UNION (que une los resultados de dos sentencias SQL)

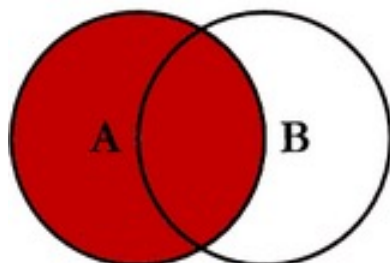
```
SELECT
    tabla1. columna1, tabla2.columna1, tabla1.columna2, ...
FROM
    tabla1
RIGHT JOIN
    tabla2 ON tabla1.id = tabla2.id
WHERE
    condición
```

## UNION

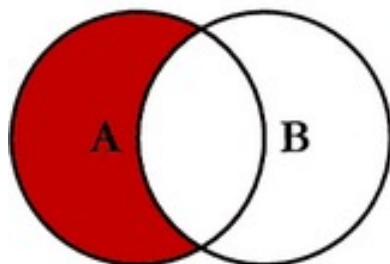
```
SELECT
    tabla1. columna1, tabla2.columna1, tabla1.columna2, ...
FROM
    tabla1
LEFT JOIN
    tabla2 ON tabla1.id = tabla2.id
WHERE
    condición
```

# SQL JOINS

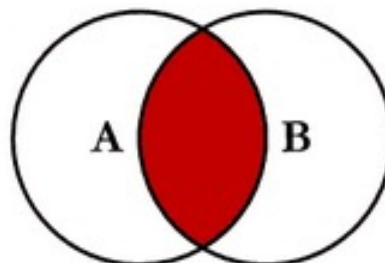
## SQL JOINS



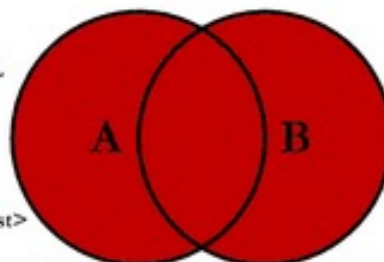
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
```



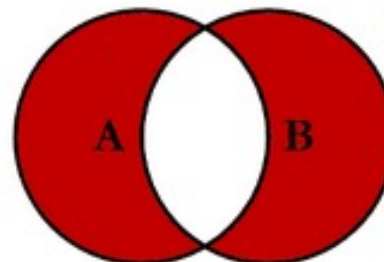
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```



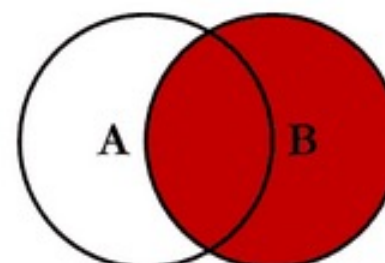
```
SELECT <select_list>
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```



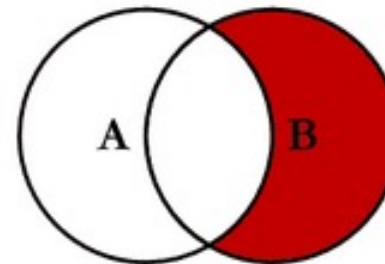
```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
OR B.Key IS NULL
```



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
```





# Ciencia de Datos

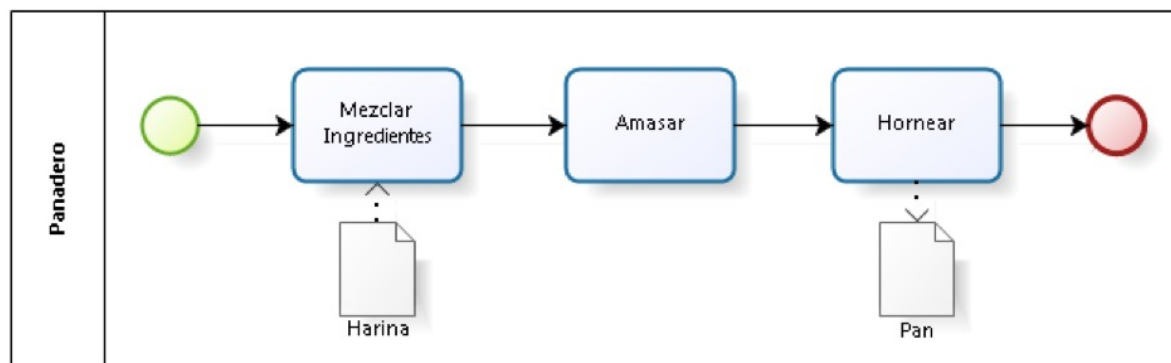
MDataSc(c). Ing. Natalia Botto Pérez

# Procesos de Negocio (BPM)

Conjunto de tareas relacionadas que se ejecutan en una secuencia específica, y que produce como resultado un producto o servicio.

La secuencia de tareas pueden incluir bifurcaciones, las cuales se ejecutan siguiendo reglas del negocio, basadas en los datos de entrada de cada una de las tareas.

Las diferentes tareas de un proceso de negocio implican el ingreso de datos (de forma manual o automática), reglas de negocio, tareas de ejecución manual o automática.



# Procesos de Negocio (BPM)

Un proceso de negocio incluye

- Actividades, Tareas, Datos, Información
- Relaciones entre ellas (secuencia, reglas, restricciones)
- Recursos para realizarlas (personas, materiales)
- Entradas, salidas (requerimientos de cliente, servicios)
- Conjunto de objetivos (organización, específicos del proceso de negocio)

# Business Process Management Notation (BPMN2)

- Notación estándar que puede ser comprendida por el área de negocio y el área de software
- Permite visualizar los procesos de negocio de la organización y todos los elementos que intervienen en el mismo (actividades, flujos, participantes, bifurcaciones, datos, subprocessos)
- Permite intercambiar modelos realizados en distintas herramientas de modelado
- Permite ejecución de procesos (en los motores de ejecución BPMN2)

# Sistema para la Gestión de Procesos de Negocio (BPMS)

- Software genérico guiado por diseños explícitos de procesos para ejecutar y gestionar procesos de negocio
- Estos sistemas incluyen (en general)
  - Modelador
  - Simulador
  - Motor de procesos

# Analítica de Procesos

La analítica de procesos busca evaluar el desempeño real de ejecución de los procesos de negocio, y definir acciones de cambio para mejorar e innovar su entrega de valor.

# Analítica de Procesos

- La minería de procesos utiliza un conjunto de técnicas para **extraer** y **analizar** los datos asociados a la ejecución de los procesos
- Los datos de ejecución de los procesos pueden ser analizados para descubrir cómo ha sido y está siendo ejecutado un proceso, para analizar cuál es su desempeño, y para facilitar la toma de decisiones de mejora

# Analítica de Procesos

## **Descubrimiento automático de modelos de proceso**

- Las técnicas para el descubrimiento automático generan modelos de proceso a partir de los datos de ejecución del proceso.
- Los datos de ejecución del proceso se consolidan en un registro de eventos (log de eventos) a partir de los datos almacenados en diferentes sistemas de información (o del BPMS)
- Aplicando diferentes algoritmos se busca analizar distintas dimensiones del proceso: flujo de control (qué se ejecuta), organizacional (quién lo ejecuta), y de casos/instancias (cómo se ejecuta)



# Analítica de Procesos

## Análisis de desempeño del proceso

- Usualmente los logs de eventos a menudo contiene un amplio conjunto de información relacionada con el proceso más allá de las tareas que se han realizado, incluidas, por ejemplo, las **marcas de tiempo** de las tareas y los **recursos** que ejecutaron estas tareas
- Se pueden identificar información de rendimiento, como la duración y frecuencia de las actividades, y la forma en que interactúan los recursos entre sí
- Analizar diferentes métricas de tiempo, costo, y calidad nos permite identificar desperdicios o ineficiencias

# Analítica de Procesos

## **Análisis de conformidad**

- Un componente esencial del análisis de procesos es revisar y asegurar que las distintas regulaciones y medidas de control incluidas en el proceso se estén cumpliendo
- Por ejemplo
  - validar el cumplimiento de reglas respecto a actividades obligatorias
  - validar el cumplimiento o no de los acuerdos de nivel de servicio establecidos
  - analizar el cumplimiento en la correcta división de tareas

# Analítica de Procesos

## **Análisis de variantes del proceso**

- Una variante del proceso puede ser la ejecución del mismo proceso según la región que se ejecutaron, si fueron ejecutadas por ciertos canales etc.
- Analizar las variantes permite obtener información como ser ¿Por qué algunas instancias tomaron más tiempo en ejecutarse que otras?, ¿Por qué ciertas instancias terminaron satisfactoriamente y otras terminaron de forma negativa, o no terminaron?, ¿Por qué el recurso es más/menos eficiente en ciertas tareas que los demás? Estas observaciones nos permiten identificar en donde canalizar los esfuerzos de mejora del proceso.

# Analítica de Procesos

## Simulación basada en datos

- La simulación de procesos nos permite estimar el **desempeño total** que tendrá un proceso, a partir de información acerca del rendimiento regular en la ejecución de sus actividades
- El desempeño total está expresado en las métricas de tiempo, costo, y utilización de recursos asociadas al proceso
- Los modelos de simulación de procesos además nos permiten definir **escenarios what-if**, los cuales ayudan a identificar el mejor escenario de configuración del proceso a partir de cambios de mejora propuestos

# Analítica de Procesos

## Simulación basada en datos

- Cuando utilizamos una aproximación basada en datos, el modelo de simulación se descubre automáticamente a partir de los datos de ejecución del proceso
- Tanto el modelo de proceso como los parámetros de simulación descubiertos reflejan la realidad de ejecución y por ende mayor precisión en las estimaciones de desempeño
- En contraste a una aproximación tradicional en la cual un analista debe configurar manualmente los parámetros del modelo de simulación basado en supuestos, la creación y ajuste de modelos de simulación en una aproximación basada en datos es mucho más rápida, y permite escalar la cantidad y calidad de escenarios what-if que quieren ser validados por el negocio para mejorar el proceso.

# Analítica de Procesos

## **Lo que buscamos es encontrar:**

- La cantidad o porcentaje de caminos frecuentes y caminos anómalos (ya que un mapa tiene muchos posibles recorridos)
- Los tiempos más altos de espera o tiempo en el cual los recursos no están realizando trabajo
- La identificación de cuellos de botella

# Analítica de Procesos

Al analizar los datos, se pueden separar los **diferentes tipos de datos**

- Un filtro a nivel de caso o instancia nos permite agrupar instancias completas de acuerdo a un criterio aplicado a la instancia.
- Un filtro a nivel de evento nos permite agrupar fragmentos de instancias de acuerdo a un criterio aplicado a los eventos.

# Analítica de Procesos

## Tiempo de procesamiento del proceso

- La medida del tiempo de procesamiento de todo proceso, es una metrica que implica la suma de los tiempos invertidos en ejecutar cada una de las tareas del proceso.
- Para cada tarea, el tiempo de procesamiento se mide desde el momento que se activa (o sea, desde que un recurso toma una tarea) hasta que finaliza su ejecución
- Para una tarea en particular, el tiempo entre su finalización y la activación de la siguiente tarea.



# Analítica de Procesos

## Tiempo de ejecución de tareas

En los procesos de negocio es normal que sea ejecutados por múltiples personas, equipos, grupos, y unidades de negocio.

- Los tiempos de espera largos pueden ser causados por
  - falta de información para iniciar una tarea
  - ausencia de notificaciones indicando tareas activas o en espera
  - muchas tareas activas, entre otras.
- Una inadecuada transferencia de trabajo, genera ineficiencias en el uso de recursos, e incrementa la duración total de ejecución del proceso.
- Algo importante a analizar son las transiciones entre tareas, o recursos del proceso, pueden generar tiempos de espera en su ejecución

# Analítica de Procesos

## **Tiempo de espera**

- Corresponde a la suma de todos los tiempos que ocurren entre tareas
- Para una tarea en particular, es el tiempo entre su finalización y la activación de la siguiente tarea

# Analítica de Procesos

## Desempeño real del proceso

El desempeño real del proceso se mide a través de dos métricas, el tiempo de ciclo y su eficiencia asociada

- El **tiempo de ciclo** indica la duración total para una instancia o ejecución del proceso, y corresponde a **la suma del tiempo de procesamiento y del tiempo de espera**.
- La **eficiencia del tiempo de ciclo** es la tasa entre el tiempo de procesamiento y el tiempo de ciclo

Esto permite analizar qué parte del tiempo realmente se invierte en la ejecución y qué parte corresponde a ineficiencias de los recursos

# Analítica de Procesos

## **Cuellos de botella**

Dentro de este análisis, se busca identificar cuellos de botella, lo que retrasa más pasos del flujo de trabajo y genera bloqueos en las ejecuciones del proceso.

Un cuello de botella se identifica cuando el tiempo de una actividad tiene una duración elevada, un alto tiempo de espera

Las causas comunes son recursos indisponibles, recursos ocupados en más procesos o sobrecarga de recursos

# Analítica de Procesos

## Utilización de recursos

Para entender el desempeño de un proceso podemos, además, utilizar una métrica de utilización de recursos. Esto busca medir cuantas horas de trabajo esta asociado a llos recursos

La tasa de utilización de recursos impacta directamente el tiempo de espera de un proceso, ya que una tarea puede tener que esperar a que un recurso esté disponible.

- Un análisis de distribución de carga de trabajo permite identificar aquellos recursos sub-utilizados en la ejecución del proceso o sobre-utilizados, y generar reasignaciones

La simulacion de procesos permite pensar optimizaciones

# Analítica de Procesos

## Simulación de procesos

Un modelo de procesos de simulación consiste, además del mapa de procesos, los siguientes elementos:

- **El tiempo promedio inter-arribo** y su función de probabilidad
- **La distribución de probabilidad del tiempo de procesamiento de cada actividad**
- Para cada compuerta de decisión del modelo de proceso, es necesario la **probabilidad para cada camino de ejecución o camino condicional**
- **El pool de recursos** que es responsable de ejecutar cada actividad en el proceso de negocio
- **El horario** (tabla de tiempos) para cada pool de recursos, indicando los periodos de tiempos durante los cuales cada recurso está disponible para realizar las actividades del proceso

# Analítica de Procesos

## Simulación de procesos

- **Características de las aproximaciones tradicionales de Simulación**
  - La configuración es manual, en base a la información conocida, brindada por usuarios
  - Tiene más altas probabilidades de contener errores dado que depende de la percepción humana más no de la realidad por sí misma
- **Características de la Simulación basada en datos (Data-driven)**
  - Se toman datos de la realidad (datos históricos) para simular tiempo de duración estimado de actividades, tiempos inter-arribo, utilización de recursos, timetables, entre otros



# Ciencia de Datos

MDataSc(c). Ing. Natalia Botto Pérez



# Anonimización de Datos

La anonimización de datos es la forma de **eliminar** las posibilidades de **identificación** de las personas.

El avance de la tecnología y la información disponible hacen difícil garantizar el anonimato absoluto, especialmente a lo largo del tiempo, pero, en cualquier caso, la anonimización va a ofrecer **mayores garantías** de privacidad a las personas.

La finalidad del proceso de anonimización es por tanto **eliminar o reducir al mínimo los riesgos de reidentificación** de los datos anonimizados manteniendo la veracidad de los resultados del tratamiento de los mismos.

# Anonimización de Datos

Además de evitar la identificación de las personas, los datos anonimizados deben garantizar que cualquier operación o tratamiento que pueda ser realizado con posterioridad a la anonimización **no conlleve una distorsión de los datos reales**.

Un análisis de los datos que pueda derivar de los datos anonimizados, **no debería diferir** del análisis que pudiera obtenerse si hubiera sido realizado con datos no anonimizados.

# Anonimización de Datos

**Dato personal:** información de cualquier tipo referida a personas físicas o jurídicas determinadas o determinables.

**Dato sensible:** datos personales que revelen origen racial y étnico, preferencias políticas, convicciones religiosas o morales, afiliación sindical e informaciones referentes a la salud o a la vida sexual

**El derecho a la protección de datos personales es un derecho amparado por la Constitución y por la Ley No 18.331.**

# Anonimización de Datos

Principios que deben orientar el uso de datos personales

- **Legalidad**

- Las bases de datos personales deben cumplir con la normativa vigente e inscribirse en el registro a cargo de la Unidad Reguladora y de Control de Datos Personales

- **Veracidad**

- Los datos registrados deberán ser veraces, adecuados, ecuanímenes (imparciales) y no excesivos en relación con la finalidad para la que se han obtenido. Será excesivo, por ejemplo, si se requiere preferencia política para afiliarse a un club deportivo

- **Finalidad**

- Los datos no deben utilizarse para fines diferentes a los solicitados y cumplida su finalidad deberán eliminarse.

# Anonimización de Datos

Principios que deben orientar el uso de datos personales

- **Previo consentimiento informado**

- Se debe contar con el consentimiento del titular para tratar sus datos.
- El consentimiento debe ser:
  - Libre (podrá brindarlo o no)
  - Previo (recabado antes de solicitar los datos)
  - Expreso (no tácito o implícito)
  - Documentado (verificable)
  - Informado (conocer la finalidad por la que se recolectan los datos y dónde ejercer sus derechos)

- **Seguridad**

- La normativa señala que se deben adoptar medidas de seguridad para proteger los datos recolectados

# Anonimización de Datos

Principios que deben orientar el uso de datos personales

- **Reserva**
  - Los datos personales deben ser tratados en forma reservada y utilizarse únicamente para la finalidad para la que se obtuvieron
- **Responsabilidad**
  - El responsable de la base de datos deberá responder por cualquier violación a las disposiciones de la Ley

# Anonimización de Datos

Datos que deben ser especialmente protegidos

- Datos sensibles
- Datos de salud
- Datos relativos a las telecomunicaciones
- Datos relativos a bases de datos con fines publicitarios
- Datos relativos a actividad comercial o crediticia

# Anonimización de Datos

La Unidad Reguladora y de Control de Datos Personales (**URCDP**)

- Asesorar al Poder Ejecutivo y recomendar políticas en el tratamiento, seguridad y manipulación de los datos personales
- Informar sobre el alcance y los mecanismos de defensa previstos por la Ley
- Inscribir las bases de datos y los códigos de conducta a través de la página [www.datospersonales.gub.uy](http://www.datospersonales.gub.uy)
- Autorizar las transferencias de datos personales a países sin niveles de protección adecuados en la materia
- Inspeccionar a las entidades públicas y privadas en relación con el tratamiento de los datos personales
- Sancionar las infracciones según el marco jurídico existente en materia de protección de datos personales.



# Anonimización de Datos

Las personas tenemos derecho a:

- Recibir información previa acerca de para qué se solicitan los datos
- Conocer qué datos poseen sobre cada uno
- Rectificarlos o cancelarlos cuando sean inexactos o incompletos
- Que nuestros datos no sean comunicados sin nuestro consentimiento, salvo las excepciones que la ley prevé

# Anonimización de Datos

Las personas tenemos derecho a:

- Impugnar aquellas valoraciones personales con efectos jurídicos, que afectan de manera significativa y que se basan únicamente en un tratamiento automatizado de datos que evalúan determinados aspectos como el rendimiento laboral, crédito, fiabilidad, conducta, entre otros. La persona afectada tiene **derecho a ser informada sobre el criterio de valoración y el programa utilizado para ello**
- No recibir publicidad no deseada
- Denunciar ante la URCDP la violación de cualquiera de estos derechos
- Consultar gratuitamente el Registro de base de datos de la URCDP

# Anonimización de Datos

Según la Guía sobre Anonimización de Datos publicada por AGESIC en 2020 se recomienda que el proceso de anonimización se incluyan estas cinco etapas

- Definición del equipo de trabajo
  - Se deben definir los diferentes perfiles o roles
  - En lo posible, que trabajen de forma independiente
  - Algunos roles
    - Responsable de la fuente de datos inicial
    - Responsables de protección de datos o referente de protección de datos personales
    - Destinatario o responsable del tratamiento de la información personal anonimizada
    - Equipo de evaluación de riesgos
    - Equipo de preanonimización y de anonimización
    - Equipo de seguridad de la información y del proceso de anonimización

# Anonimización de Datos

Según la Guía sobre Anonimización de Datos publicada por AGESIC en 2020 se recomienda que el proceso de anonimización se incluyan estas cinco etapas

- Definición de objetivos y finalidad de la información anonimizada
  - Objetivos que deberá cumplir la información anonimizada en función de los intereses de su destinatario
  - El diseño del proceso de anonimización estara condicionado a los objetivos

# Anonimización de Datos

Según la Guía sobre Anonimización de Datos publicada por AGESIC en 2020 se recomienda que el proceso de anonimización se incluyan estas cinco etapas

- Preamonimización
  - Definir las variables a anonimizar
  - Identificar la información que será incluida en el proceso de anonimización
  - No todos los involucrados deberán participar de esta etapa

# Anonimización de Datos

Según la Guía sobre Anonimización de Datos publicada por AGESIC en 2020 se recomienda que el proceso de anonimización se incluyan estas cinco etapas

- Anonimización
  - No puede establecerse vínculo alguno entre el dato y su titular sin un esfuerzo desproporcionado
  - No puede ser reversible
  - Que en la práctica sea equivalente al de un borrado permanente
  - Que lleve implícito un factor de riesgo que se debe tener en cuenta al valorar las técnicas de anonimización, además de considerarse la gravedad y probabilidad del riesgo en sí mismo
  - Se deberán aplicar las técnicas seleccionadas, los algoritmos necesarios, realizar pruebas de calidad y entregar el resultado al responsable para su aprobación
  - El objetivo final de la anonimización es proveer los datos desagregados para ser utilizados, sin generar conflictos con los titulares de los datos

# Anonimización de Datos

Según la Guía sobre Anonimización de Datos publicada por AGESIC en 2020 se recomienda que el proceso de anonimización se incluyan estas cinco etapas

- Control
  - Implica controlar lo realizado, de forma periodica, en virtud de la aparición de las nuevas tecnologías y métodos para prevenir y evitar los posibles riesgos de reidentificación
  - Dependiendo del resultado de esta etapa, es posible volver a replantearse distintos escenarios para la etapa de preanonimización y anonimización, para luego volver a realizar un nuevo control

# Definiciones

## **Microdatos**

Cada registro de una base de datos o en un conjunto de datos, contiene información relacionada a un individuo específico.



# Definiciones

## **Datos tubulares**

El conjunto de datos muestra valores para distintos grupos de individuos (ejemplo resultado de estadísticas).

# Definiciones

## **Bases de datos interactivas**

Los datos no son liberados directamente, sino que se presentan a los usuarios a través de una interfaz.

# Definiciones

## Atributos

Los atributos pueden ser **Continuos** (valores numéricos) o **Categoricos** (valores, en general de tipo texto, que pueden asignárseles una cantidad finita de valores)

# Definiciones

## Atributos

Los atributos **Identificadores** son aquellos que permiten identificar de forma univoca a un registro (y por lo tanto un individuo).

Los atributos **Quasi-Identificadores** por si solos no permiten reidentificar a un individuo, pero que la combinación de varios QI podría llevar a la identificación inequívoca de algunos individuos.

Los atributos **Confidenciales** son aquellos que contienen información sensible.

Los atributos **No Confidenciales** son aquellos que no pertenecen a ninguna de las anteriores.

# Definiciones

## Riesgos de divulgación

La **divulgación de identidad** refiere a que un atacante es capaz de asociar un registro perteneciente al conjunto anonimizado con uno del conjunto original. Permitiendo así reidentificar un individuo del conjunto.

La **divulgación de atributos** alude a que un atacante es capaz de determinar el valor de un atributo confidencial de un individuo con un gran nivel de seguridad.

# Técnicas de Anonimización

Las técnicas de anonimización de datos se dividen en dos grandes familias

- las técnicas sin perturbación
  - Las técnicas sin perturbación mitigan los riesgos de divulgación suprimiendo ciertos valores y/o reduciendo el nivel de detalle de los datos del conjunto
- las técnicas con perturbación
  - Las técnicas con perturbación en lugar de suprimir valores del conjunto de datos, alteran la distribución estadística de los mismos

# Técnicas de Anonimización de Datos

## Aleatorización

Conjunto de técnicas que modifican la veracidad de los datos para eliminar el vínculo con su titular

- Adición de ruido
  - Modifica los atributos del conjunto de datos para que sean menos identificables (ejemplo, alterando ciertos valores dentro de cierto rango)
- Permutación
  - Mezcla los valores de los atributos para que puedan vincularse artificialmente con otros titulares
- Privacidad diferencial
  - Genera vistas anonimizadas de un conjunto de datos, al mismo tiempo que almacena una copia de los originales

# Técnicas de Anonimización de Datos

## Generalización

Generaliza o diluye los atributos de los interesados modificando las respectivas escalas u órdenes de magnitud. No siempre permite anonimizar de forma eficaz

- Agregación y anonimato K
  - Evita que los registros sean singularizados agrupándolos
  - Esto se hace ya sea suprimiendo valores (por ejemplo por un \*) o generalizándolos (por ejemplo, rangos de edades)
- Diversidad l y proximidad t
  - Se crean clases de equivalencia (al menos l diferentes) que se repiten tantas veces como sea necesario para mantener la distribución original





# Ciencia de Datos

MDataSc(c). Ing. Natalia Botto Pérez

# Riesgos y consideraciones

Todos los proyectos tienen riesgos o consideraciones para llevar adelante, en el caso de los proyectos de Ciencia de Datos surgen varias consideraciones específicas a considerar

# Riesgos y consideraciones

## Problemas con la definición del proyecto

- No definir correctamente las preguntas
- No abordar la causa raíz, solo tratar de mejorar el efecto de un proceso
- Análisis incorrecto en base a la información disponible
- No contar con perfiles con conocimiento del negocio para apoyar en el análisis e interpretación de resultados

# Riesgos y consideraciones

## Problemas asociados a los datos

- Problemas de calidad en los datos
- Problemas de acceso a los datos
- Datos donde se identifican errores (no corregibles)
- Identificar que no se tienen los datos adecuados

# Riesgos y consideraciones

## Problemas presupuestales

- No se planifica correctamente los recursos materiales que se van a necesitar (hardware)
  - Principalmente vinculado a proyectos de aprendizaje automático, vinculado al entrenamiento de modelos
- Falta de disponibilidad de personas con los perfiles adecuados

# Riesgos y consideraciones

## Problemas de seguridad

- Al generar proyectos de DataScience para toma de decisiones, muchas veces se generan grandes bases de datos con información muy relevante para las empresas u organizaciones
- Se deben generar niveles de seguridad para evitar el robo o la publicación de datos confidenciales

# Riesgos y consideraciones

## Problemas de privacidad de datos

- Parte del análisis de un proyecto de Ciencia de Datos es la necesidad de mantener la privacidad de los datos
- Procesos de anonimización incorrectos pueden generar problemas legales de privacidad

# Riesgos y consideraciones

## Problemas de sesgos

- Sesgos al momento de obtener los datos
- Trasladar sesgos al análisis de los datos



# Riesgos y consideraciones

## Problemas éticos

- Cada vez más la ética forma parte de los proyectos de análisis de datos
- Al plantear un proyecto, es necesario analizar que no hayan problemas éticos, principalmente cuando se tratan datos sensibles

# Riesgos y consideraciones

## Matriz de riesgos

#	Nombre	Descripción	P	I	PxI	Estrategia	Plan de acción
(numeración)	(nombre del riesgo)	(descripción detallada, en que situaciones se puede activar el riesgo)	(probabilidad de ocurrencia)	(Impacto)	(nivel de riesgo)	(Aceptar, Mitigar, Transferir, Evitar)	(descripción del plan de acción, ya sea acciones para mitigar / evitar / transferir el riesgo o acciones a realizar al momento que se active el riesgo)
1							
2							
3							

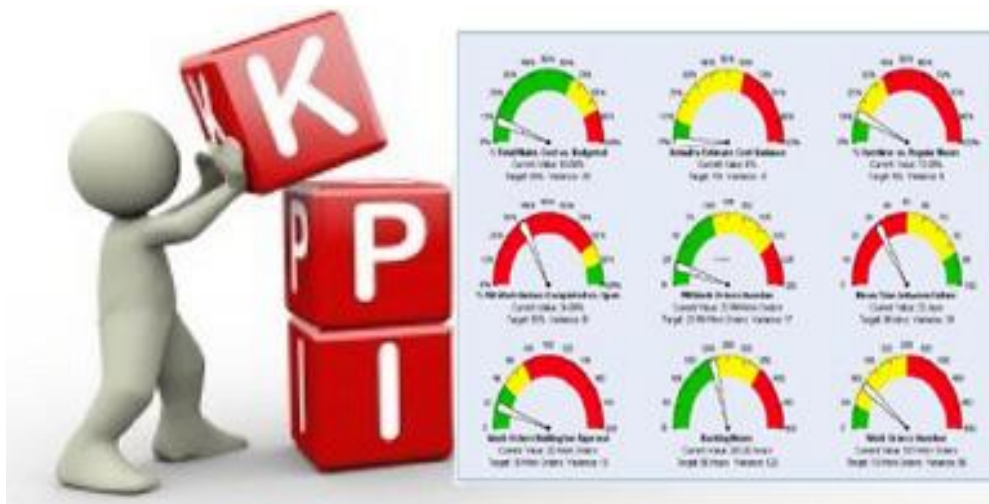
		Probabilidad		
		B	M	A
I m p a c t o	B			
	M			
	A			

# Ciencia de Datos

## KPIs

**Ing . Hugo Moreno**

# KPI: Definición, construcción e interpretación



# KPI: Definición

Datos o información utilizados para conocer o valorar las características e intensidad de un hecho o para determinar su evolución futura<sup>2</sup>

<sup>2</sup> diccionarios oxford

# KPIs: definiciones mas detalladas

- ▶ Un KPI (*key performance indicator*) es un indicador clave para el negocio, permite medir el desempeño y/o el rendimiento
- ▶ Es una forma de medir como se estan ejecutando mis procesos
- ▶ Esta relacionado con los objetivos con los cuales fueron definidos
- ▶ Un KPI es una cuantificación de un desafío empresarial importante

# KPIs:

## Definiciones con enfoque basado en liderazgo

**KPI=**

~~**Key Performance Indicators**~~

The New Leadership

**KPI**

Keep people interested

Keep people informed

Keep people involved

Keep people inspired

# KPIs

## Consideraciones en el ambiente corporativo

- ▶ Definir incorrectamente los KPIs pueden afectar los ingresos, costos, estrategias a diseñar u otros
- ▶ Estas métricas de rendimiento se han vuelto muy importantes para muchos negocio, pero es importante definirlas correctamente, y también saber medirlas
- ▶ Mientras que las métricas de alto nivel pueden estar orientadas al performance general del negocio, las de bajo nivel podrían estar dirigidos a departamentos o áreas puntuales.



# KPI: Definición, construcción e interpretación

- **Simple:** Busquemos que sea integral desde el nombre del indicador hasta su fórmula de cálculo y los recursos requeridos en la medición.
- **Objetivos y neutrales:** Midamos lo que realmente importa medir, y describamos el fenómeno. Busquemos también que sea independiente de otro, es decir, que no esté correlacionado con otro indicador.
- **sistemática**(periodicidad de análisis): Que sea posible recolectar la información en el tiempo para que, una vez que analicemos la información, tengamos tiempo para tomar decisiones.

# Razon principal de emplear KPIs

**Habilita para la toma de decisiones:**

*Lo que no se mide no se puede controlar.*

*Si no se controla, no se puede abordar.*

*Si no se aborda, no se puede mejorar.*

**Los indicadores son los medios, no la meta.**

# Cómo construir un KPI

Paso 1: ¿qué estás haciendo?

Paso 2: Definición del indicador: variables y fórmula de cálculo

Un indicador puede ser:

- **Absoluto:** Un número que dimensiona un evento o fenómeno de acuerdo a su naturaleza. (27 grados Celsius)
- **Proporción compuesta:** Es el cociente entre dos cantidades que no tienen elementos comunes o tienen atributo de diferencia. (4 camas por cada 6 personas)
- **Calificar:** Es el cociente entre dos variables analizadas en un lugar y tiempo específico. (La tasa de mortalidad es de 0,2 por cada 1000 habitantes al año.)
- **Proporción simple:** Es una relación donde el numerador está incluido en el denominador. (50% de los graduados en 2009 tiene trabajo)
- **Variación:** Establece dos elementos para establecer qué variación existe entre uno y otro. (las ventas aumentaron un 20% en comparación con el año anterior)

# KPIs

## Pasos para definir el indicador

- ▶ Al definir un indicador
  - ▶ Definir que preguntas quiere responder el indicador
  - ▶ Identificar que datos
  - ▶ Definir la metodología de medición y frecuencia de medición
  - ▶ Definir umbrales y metas -> *esto es lo que nos ayuda a saber hacia donde se mueve la aguja cuando finalmente queremos medir la ejecución!*
  - ▶ Comunicar el indicador

# KPIs

## Pasos para definir el indicador

- ▶ Al definir un KPI es importante considerar el acronimo SMART
  - ▶ ESpecíficos (*Specific*)
  - ▶ Medibles (*Measurable*)
  - ▶ Alcanzables (*Achievable*)
  - ▶ Relevantes (*Relevant*)
  - ▶ OporTunos (*Timely*)

# Tipos de KPIs

## ► Cualitativos y cuantitativos

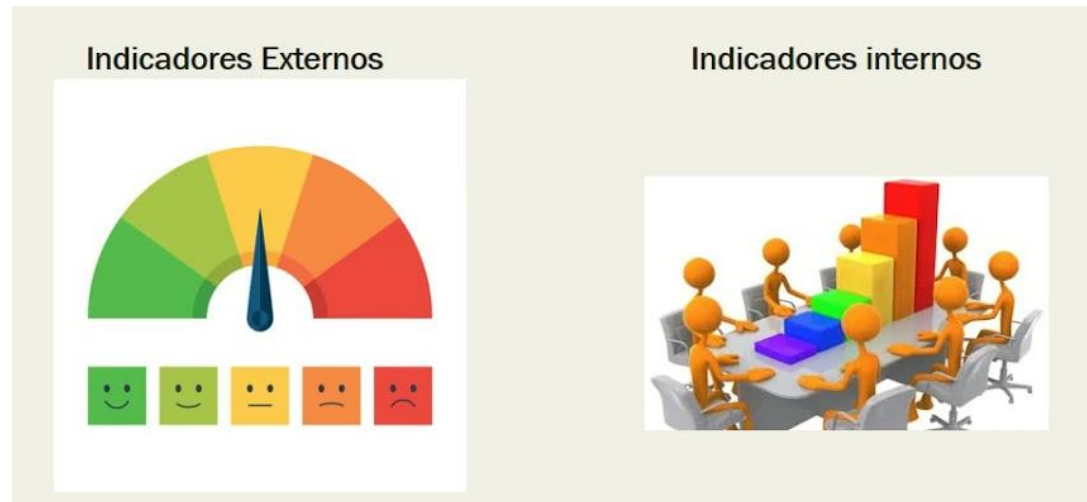
- Es uno de los enfoques más usados en la industria para gestionar organizaciones
- Los indicadores que tienen que ver con percepciones u opiniones son los considerados cualitativos y los demás están designados como cuantitativos.



# Tipos de KPIs

## ► Internos y externos

- Esta clasificación implica que si son internas a la organización pueden implicar cierto grado de confidencialidad e importancia
- Los externos son valores accesibles por el público ajeno a la compañía



# Tipos de KPIs

## ▶ Corto plazo y largo plazo

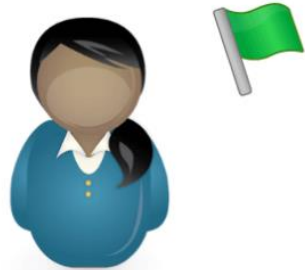
- ▶ KPIs sirven para medir lo que ocurre en el presente o lo que podría suceder en el futuro
- ▶ Pueden estar sujetos a objetivos a corto o largo plazo



# Tipos de KPIs

## ► Eficacia y eficiencia

- La eficacia refiere a la obtención de los resultados deseados
- La eficiencia incluye el esfuerzo, tiempo y dinero usado para conseguir un objetivo



**EFICAZ**  
Llega a la meta  
(pero tal vez no optimiza recursos)



**EFICIENTE**  
Optimiza recursos  
(pero tal vez no llega a la meta)

# Tipos de KPIs

## ► Predictivos e históricos

- Un indicador histórico hace referencia a una serie de valores dentro de un lapso del pasado
- Un indicador predictivo permite tomar decisiones a futuro. **Sirven como orientación** y muestran si se está produciendo una desviación en lo planificado

# Visualización de KPIs

Al momento de visualizar KPIs es importante acompañarlas con los niveles de rendimiento deseados y especificar cómo se interpreta la información

Sin dicho nivel de referencia la información sería confusa y no permitiría discernir si las metas se cumplieron o no y en que medida.

*Las mejores practicas aconsejan traducir el valor medido a un índice (proporción llevada a un valor cercano a uno) a un valor porcentual (con respecto al 100%) en lo que respecta al valor establecido como meta, o simplemente emplear una visual que permita evaluar de manera grafica que tan alejado esta dicho valor con respecto al valor esperado.*

# Dashboard

Es una representación gráfica de los principales indicadores (KPI) que intervienen en el logro de los objetivos de negocio, y que está orientada a la toma de decisiones para optimizar la estrategia de la organización.

Un dashboard debe permitir transformar los datos en información y esto en conocimiento para el negocio tal que permita la toma de acciones.

## La forma de control más sencilla mediante el uso de KPI: un Dashboard o Tablero de Control



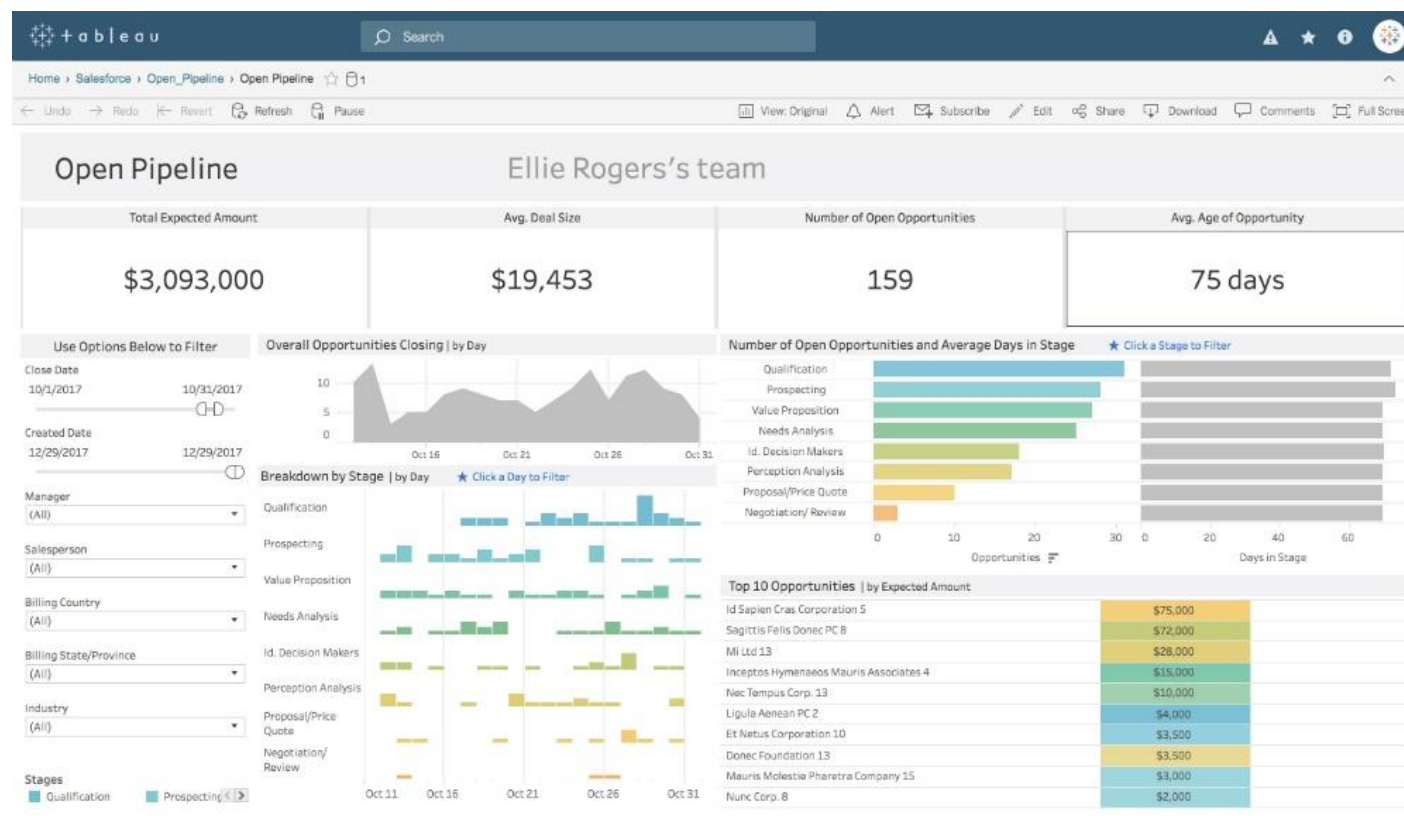
# Dashboards

Características que debe tener

- ▶ **Util**
  - ▶ Debe contener las metricas que permiten responder las preguntas del negocio
- ▶ **Visual**
  - ▶ Debe permitir una buena visualización de los datos
- ▶ **Comprensible**
  - ▶ Debe permitir la toma de decisiones de quienes esta destinado consuman el dashboard
- ▶ **Actual**
  - ▶ Actualizado en tiempo real, ya que los valores evolucionan rápidamente

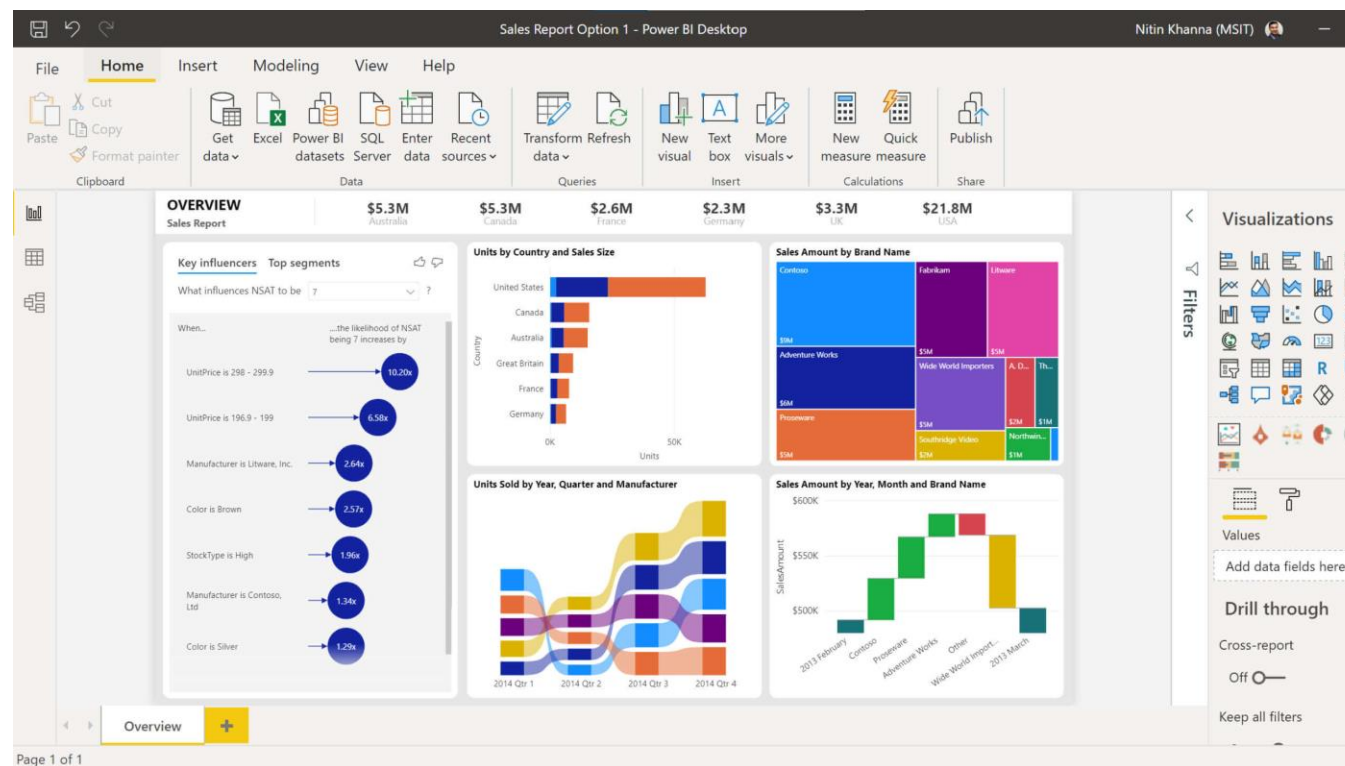
# Herramientas para generar Dashboards

## Tableau



# Herramientas para generar Dashboards

## Power BI





## Acerca de MS Power BI



*Hace siete años, Microsoft quería optimizar Power BI y hacerlo exitoso. Hicieron precisamente eso con James Phillips liderando su equipo de desarrollo para crear una herramienta de desarrollo elegante y simplificada llamada **Power BI Desktop**. Su mantra - **5 segundos para registrarse y 5 minutos para asombrarse con los resultados.***

## MS Power BI permite:

- Crear informes impresionantes con visualizaciones de datos interactivas.
- Acceder a datos de cientos de fuentes locales y basadas en la nube, como Dynamics 365, Salesforce, Azure SQL DB, Excel y SharePoint.
- Asegurarse que sus reportes estan siempre al dia con actualizaciones incrementales automatizadas.
- Profundizar en los datos y encontrar patrones que de otro modo podría haber pasado por alto y que conducen a información procesable.
- Proporcionar a los usuarios avanzados un control total sobre su modelo mediante el potente lenguaje de fórmulas DAX. Si está familiarizado con Excel, se sentirá como en casa en Power BI.
- Obtener análisis visuales para las personas que lo necesiten.
- Crear informes optimizados para dispositivos móviles para que los espectadores los consuman sobre la marcha

# Si ya domino Excel por qué necesito Power BI?:



*La tabla dinámica ha sido la principal herramienta analítica en Excel durante décadas. Todo el mundo lo usa a diario. Entonces, ¿por qué necesitamos otra herramienta analítica? Dos razones:*

Power BI elimina las limitaciones de la tabla dinámica y Excel (límite de un millón de filas, rendimiento lento, tamaños de archivo grandes, dificultad para compartir datos). Mejora las capacidades de Pivot Table multifold. **Ahora podemos administrar cientos de millones de filas de datos, usar los servidores de Microsoft extremadamente potentes y basados en la nube para procesar datos y crear informes interactivos en segundos..**

# Si ya domino Excel por qué necesito Power BI?

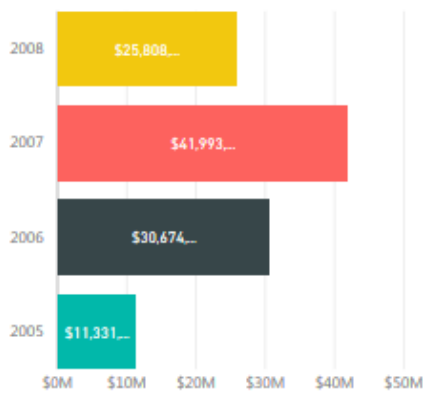
Adicionalmente, Power BI proporciona nuevas capacidades como trabajar con fechas, visualización geográfica, crear nuevos conocimientos mediante el aprendizaje automático, etc.

Los datos que utiliza a diario pueden no ser diferentes. Sin embargo, al usar Power BI, podrá desenterrar cada vez más información útil de los MISMOS datos. Esta información se estaba perdiendo hasta ahora. Nadie lo buscaba y no tenías la herramienta para hacerlo. ¡Es como perder oportunidades de negocio todos los días!

### Bar Chart

Total Sales by Calendar Year and Country

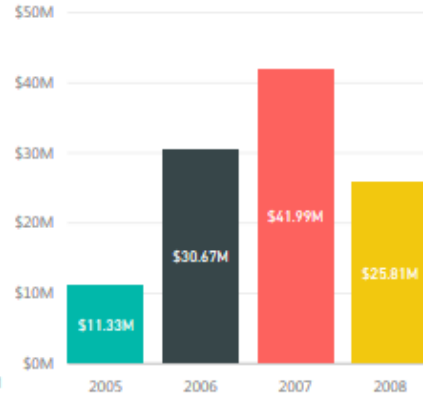
Calendar Year ● 2005 ● 2006 ● 2007 ● 2008



### Column Chart

Total Sales by Calendar Year and Country

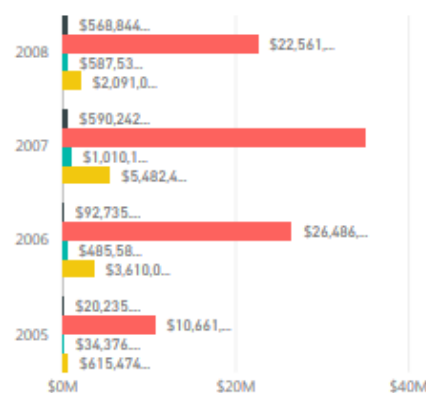
Calendar Year ● 2005 ● 2006 ● 2007 ● 2008



### Clustered Bar Chart

Total Sales by Calendar Year and Country

Product Categ... ● Accessories ● Bikes ● Clothing ▶



### Clustered Column Chart

Total Sales by Calendar Year and Country

Product Categ... ● Accessories ● Bikes ● Clothing ▶



### 100% Stacked Bar Chart

Total Sales by Calendar Year and Country

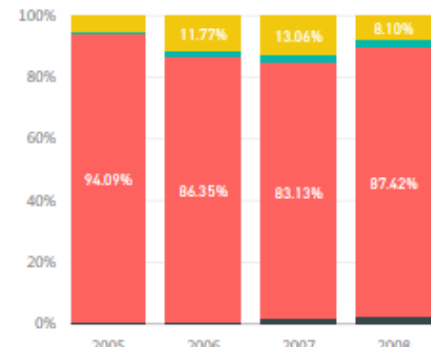
Product Categ... ● Accessories ● Bikes ● Clothing ▶



### 100% Stacked Column Chart

Total Sales by Calendar Year and Country

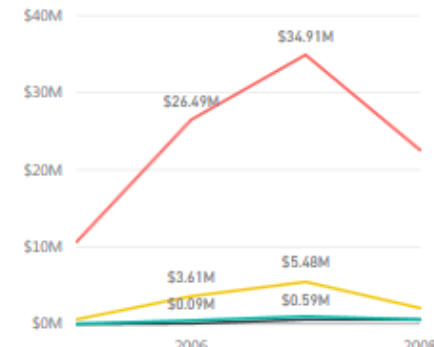
Product Categ... ● Accessories ● Bikes ● Clothing ▶



### Line Chart

Total Sales by Calendar Year and Country

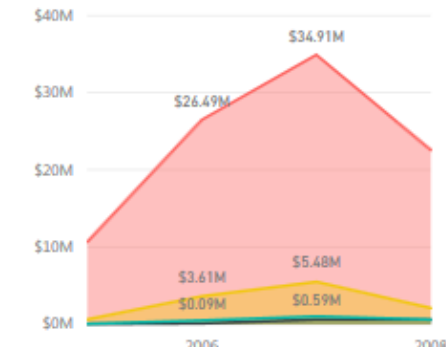
Product Categ... ● Accessories ● Bikes ● Clothing ▶



### Area Chart

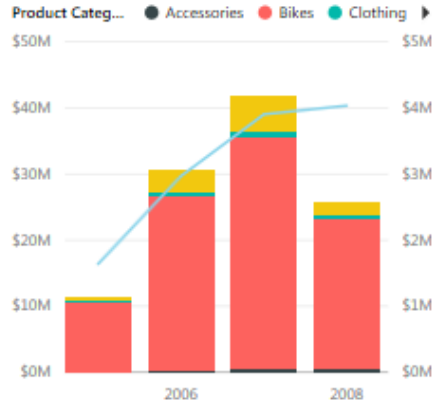
Total Sales by Calendar Year and Country

Product Categ... ● Accessories ● Bikes ● Clothing ▶



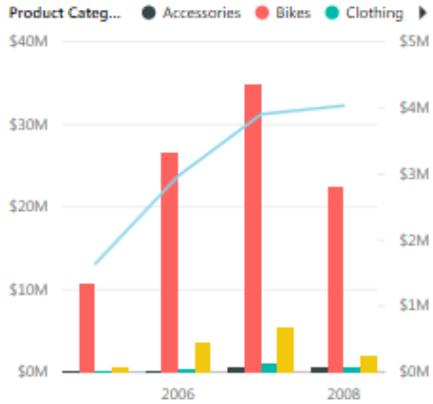
### Line and Stacked Column Chart

Total Sales and Total Gross Profit by Calendar Year and...



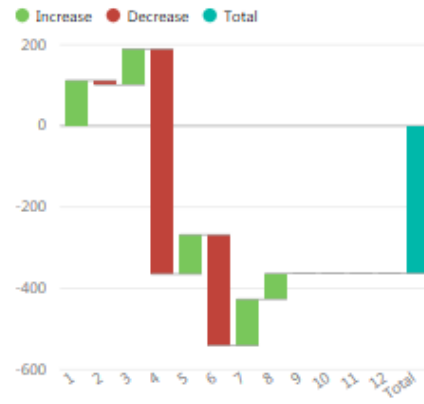
### Line and Clustered Column Chart

Total Sales and Total Gross Profit by Calendar Year and...



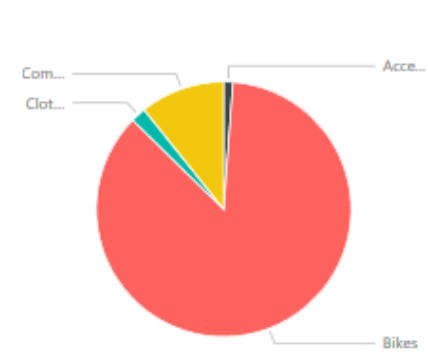
### Waterfall Chart

Total Units Movement by Month



### Pie Chart

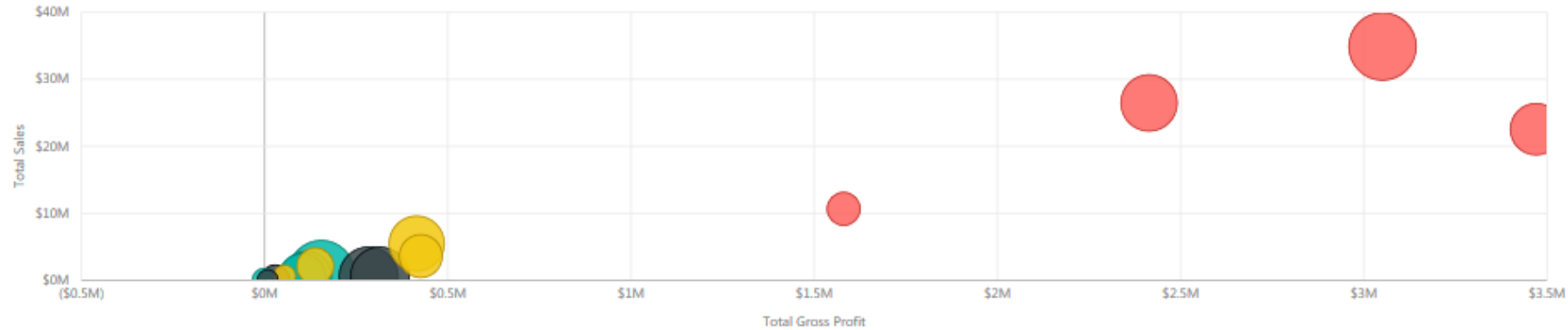
Total Sales by Product Category



### Scatter Chart

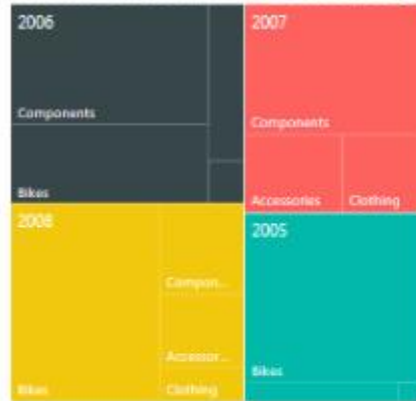
Total Gross Profit, Total Sales and Total Units Sold by Calendar Year and Product Category

Product Categ... ● Accessories ● Bikes ● Clothing ● Components



## Treemap

Total Gross Profit by Calendar Year and Product Category



## Map

Total Gross Profit by Country Name and Country Name

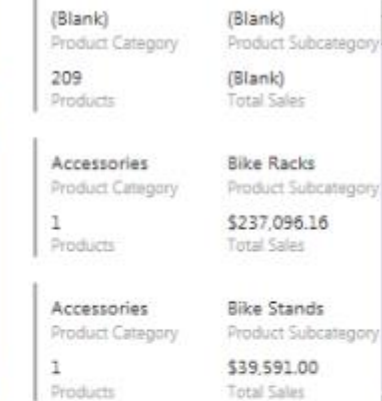


## Filled Map

Total Sales by State Name



## Multi Row Card

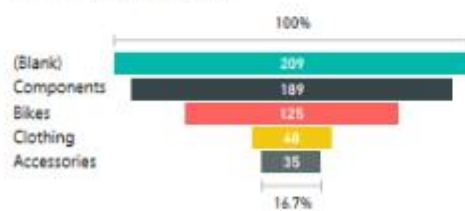


## Table

Calendar Year	Total Sales
2005	\$11,331,808.97
2006	\$30,674,773.18
2007	\$41,993,729.83
2008	\$25,808,962.35
<b>Total</b>	<b>\$109,809,274.32</b>

## Funnel

Products by Product Category



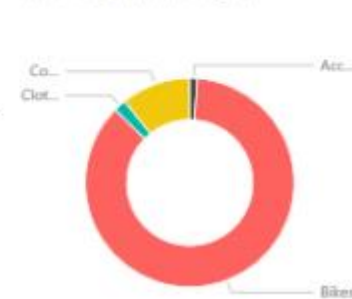
## Gauge

Total Sales, Minimum, Maximum and Sales...



## Donut Chart

Total Sales by Product Category



## Slicer

Calendar Year

- ☐ Select All
- ☐ 2005
- ☐ 2006
- ☐ 2007
- ☐ 2008
- ☐ 2009
- ☐ 2010

## Matrix

Calendar Year	1	2	3	4	Total
2005			\$4,647,156.86	\$6,684,652.11	\$11,331,808.97
2006	\$5,860,884.49	\$6,167,832.57	\$10,277,073.05	\$8,368,983.06	\$30,674,773.18
2007	\$6,679,873.81	\$8,357,874.87	\$13,670,536.66	\$13,285,444.49	\$41,993,729.83
2008	\$11,386,315.07	\$14,371,806.64	\$50,840.63		\$25,808,962.35
<b>Total</b>	<b>\$23,927,073.37</b>	<b>\$28,897,514.09</b>	<b>\$28,645,607.20</b>	<b>\$28,339,079.67</b>	<b>\$109,809,274.32</b>

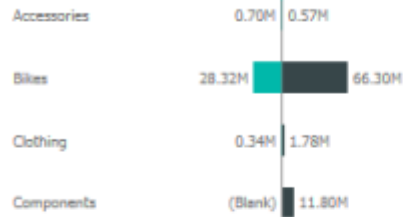
## Card

**606**  
Products



### Tornado

Internet Sales vs Reseller Sales by Product Category

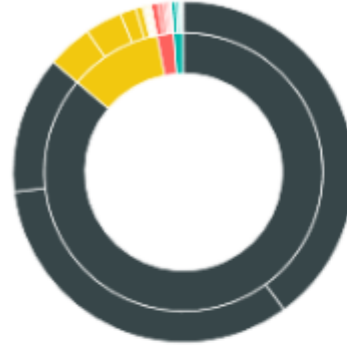


### Tadpole Spark Grid



### Sunburst

Total Sales by Product Category and Product Sub...



### Synoptic Panel

Total Sales by Product

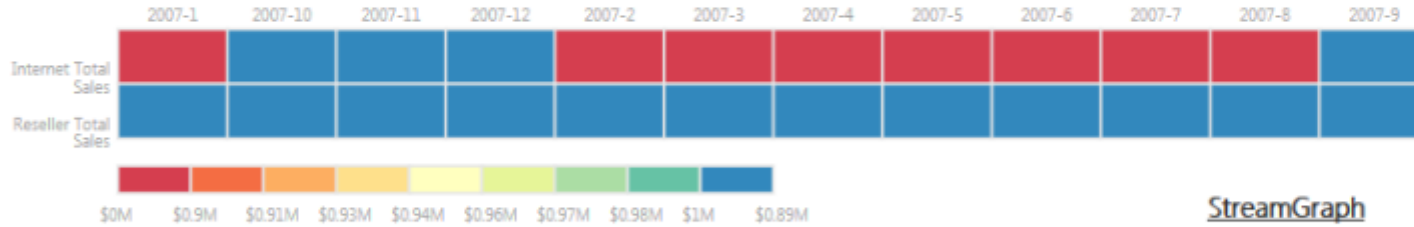
☒ SELECT MAP

☒ SELECT AREAS



### Table Heatmap

Internet Total Sales and Reseller Total Sales by YearMonth



### KPI Indicator w/ Status, Deviation and History



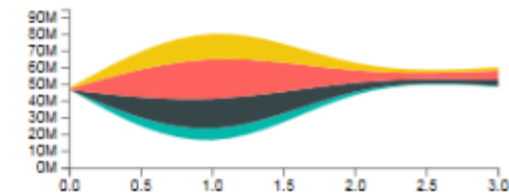
### KPI Indicator w/ Status, Deviation and History



### StreamGraph

Total Sales by Product Category and Calendar Year

Calendar Year: 2005 (teal), 2006 (black), 2007 (red), 2008 (yellow)





## Card with States by SQLBI

**\$14M**

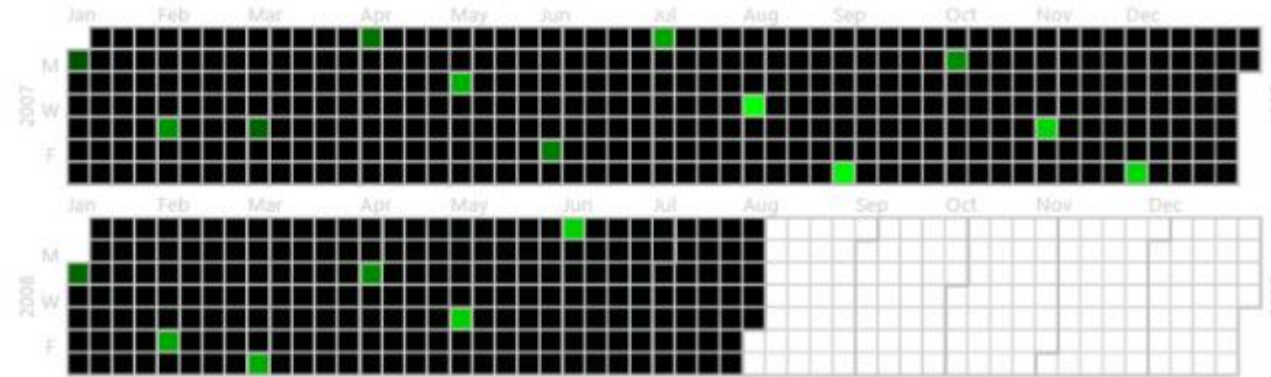
Total Current Quarter Sales

**\$23M**

Total Inventory Value

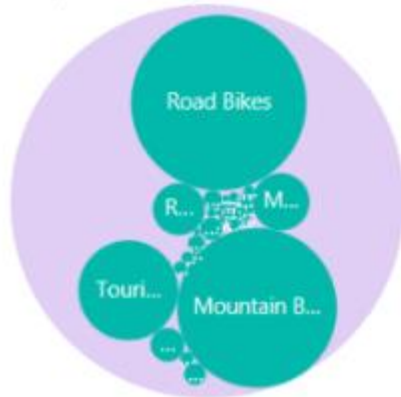
## Calendar Visualization

Total Sales by Date



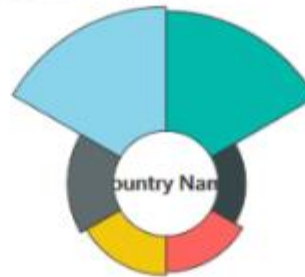
## Bubbles

Total Sales by Product Subcategory Name



## Aster Plot

Internet Total Sales by Country Name



## Bullet Chart

Sales vs Sales Quota

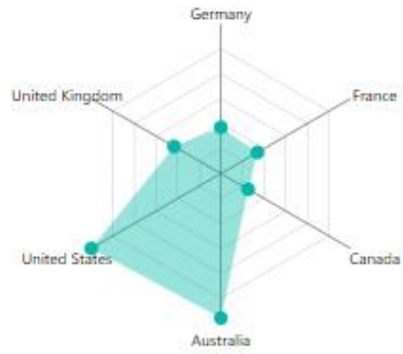


## Chiclet Slicer



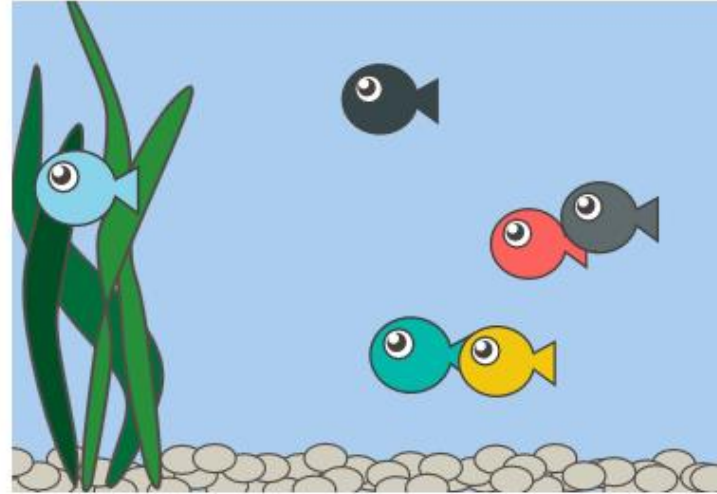
## Radar Chart

● Internet Total Sales



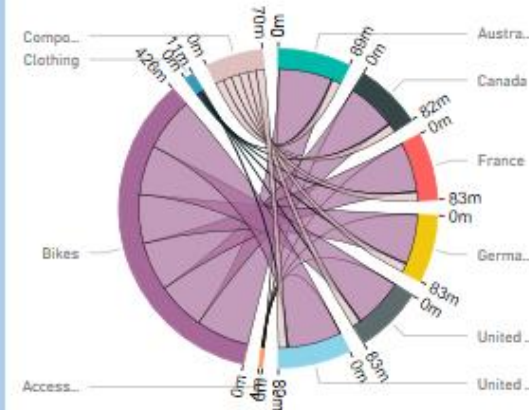
## Enlighten Aquarium

Total Sales and Total Units Sold by Country Name



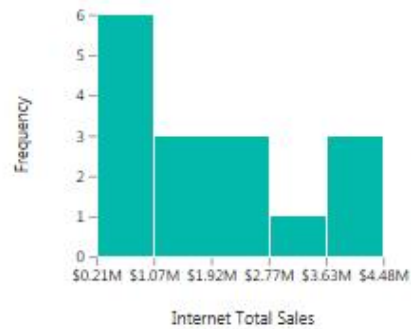
## Chord

Total Sales by Country Name and Product Category



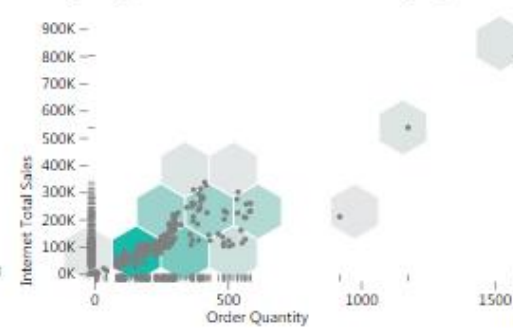
## Histogram

Internet Total Sales by Yearly Income



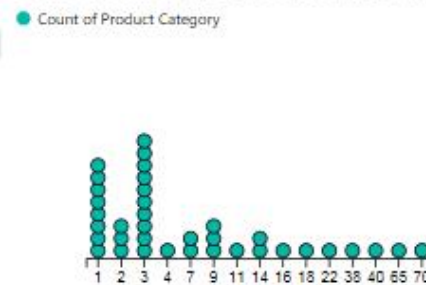
## Hexbin Scatterplot

Order Quantity, Internet Total Sales and Customers by City-Old



## DotPlot

Count of Product Category by Product Subcategory Name



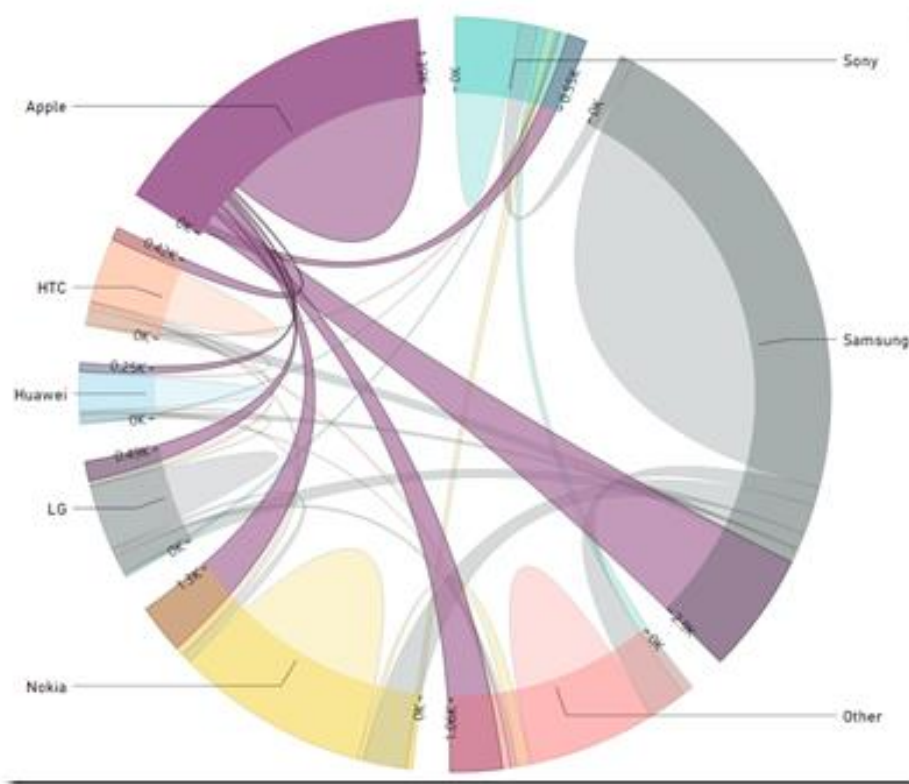
## Enhanced Scatterplot

Order Quantity, Internet Total Sales and Customer...



## Ejemplo práctico: informe de acordes aplicado al mercado de la telefonía móvil

*This shows that a lot of Samsung users remain Samsung customers, but it also shows that Apple is taking a portion of their market share.*



# Preguntas?

