

Exploración y análisis de datos

Febrero 2023



red.es

“El FSE invierte en tu futuro”
Fondo Social Europeo

Centro de
Referencia Nacional
en Comercio Electrónico
y Marketing
CRN
Digital

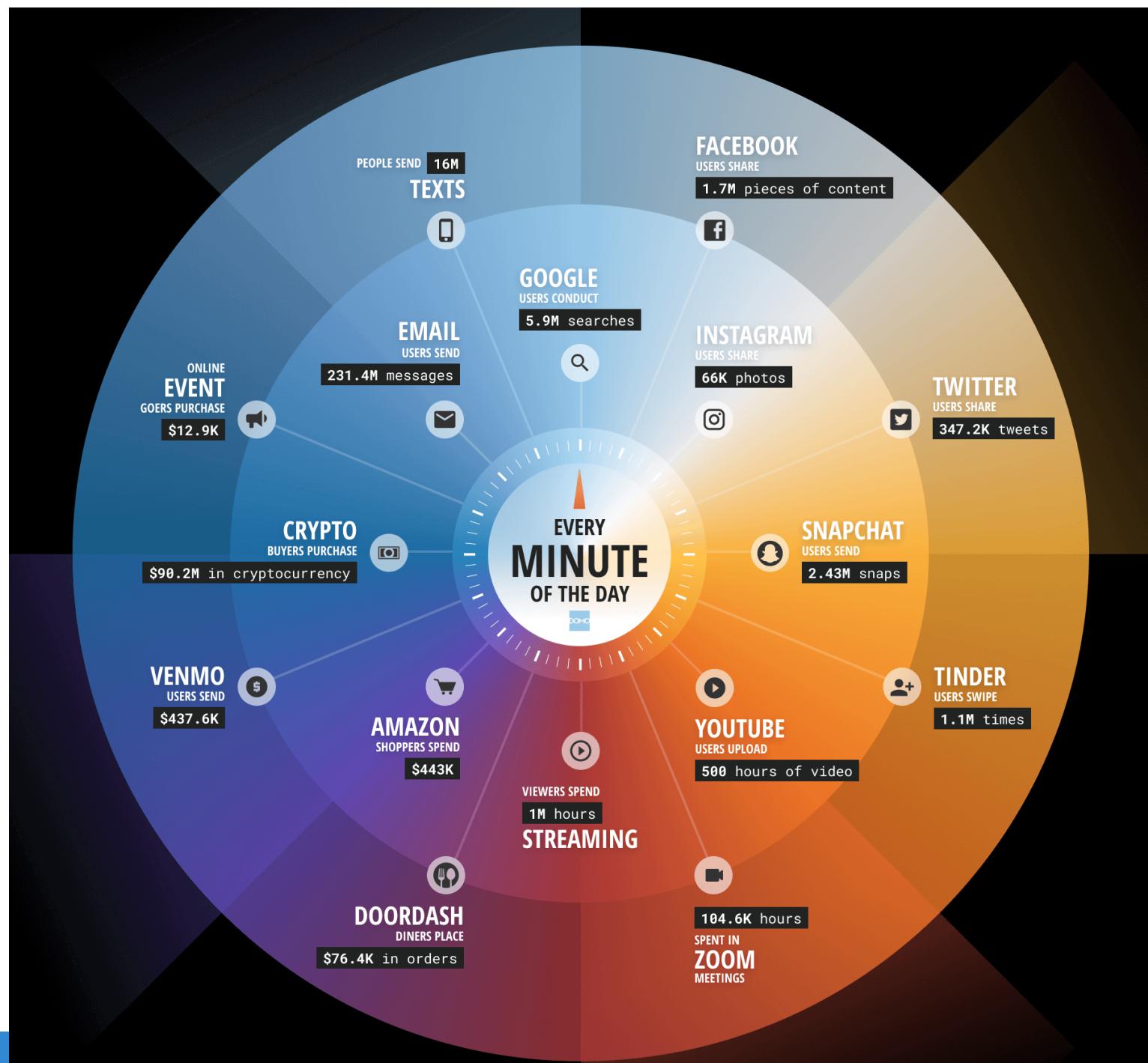



Barrabés

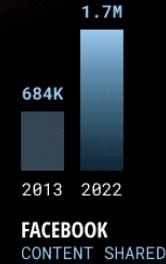
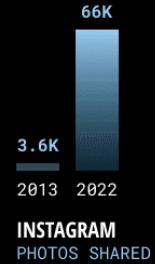
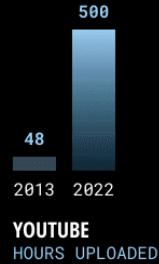
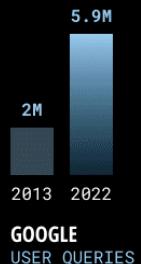

The Valley

1. Contextualización
2. Introducción al Aprendizaje Automático
 - 1.1 Motivación
 - 1.2 ¿Qué es el Aprendizaje Automático?
3. Conceptos fundamentales del ML
 - 2.1 Diagrama de bloques de un sistema de aprendizaje
 - 2.2 Elementos básicos
 - 2.3 Objetivo
 - 2.4 Conjuntos de entrenamiento y test
 - 2.5 Pasos
3. Repasemos EDA y Data Preparation
4. Herramientas para el ML
 - 4.1 Álgebra lineal
 - 4.2 Cálculo
 - 4.3 Programación
5. Ejemplo práctico

Contextualización



DATA NEVER SLEEPS 1.0 VS. 10.0



GLOBAL INTERNET POPULATION GROWTH

IN BILLIONS



As of April 2022, the internet reaches 63% of the world's population, representing roughly 5 billion people. Of this total, 4.65 billion - over 93 percent - were social media users. According to Statista, the total amount of data predicted to be created, captured, copied and consumed globally in 2022 is 97 zettabytes, a number projected to grow to 181 zettabytes by 2025.

To succeed in an increasingly digital world where the volume of data created keeps accelerating, businesses need the right tools to put that data to work right where work gets done. Domo gives you the power to rapidly unlock value from all your data, regardless of where it lives, and drive actions across your organization that will improve business outcomes. Every click, swipe, share, or like tells a story, and Domo helps you do something powerful with it.

LEARN MORE AT DOMO.COM

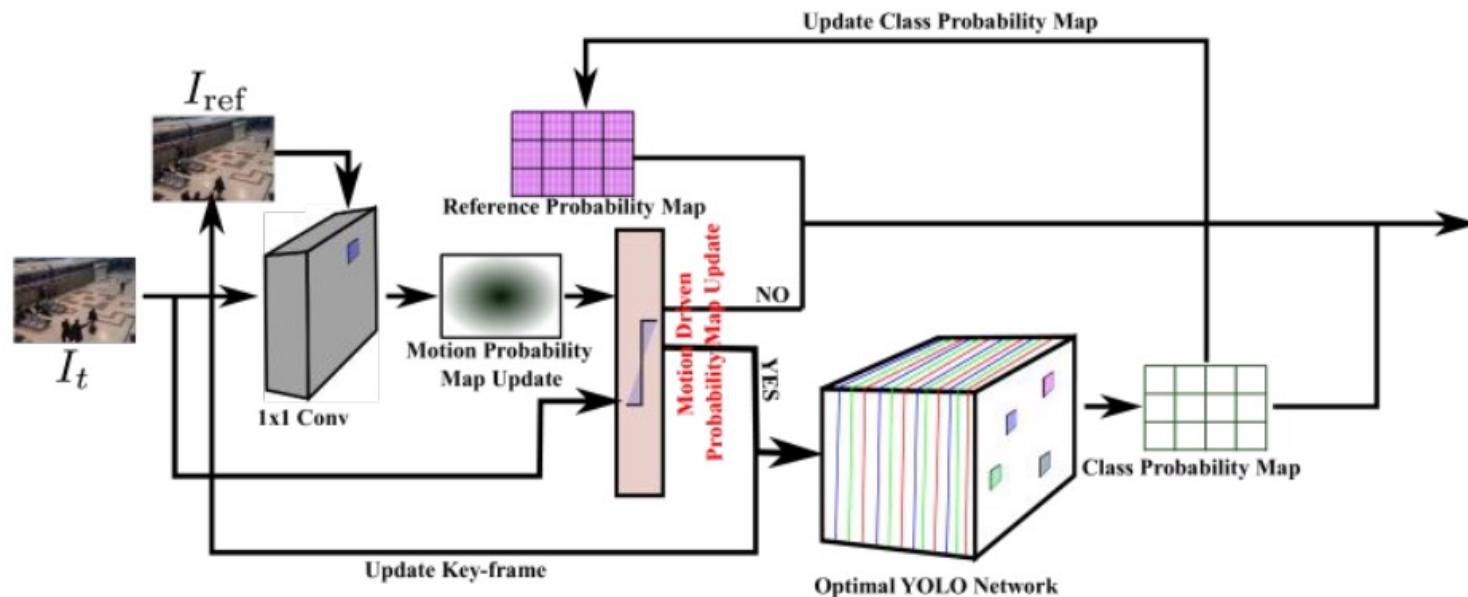
SOURCES

Global Media Insight, Oberlo, Hootsuite, Earthweb, Matthew Woodward.co.uk, Web Tribunal, Deadline.com, Local IQ, Business of Apps, Query Sprout, Young and the Invested, Dating Zest, IBIS World, DoorDash, TechCrunch, Statista, Data Never Sleeps 1.0



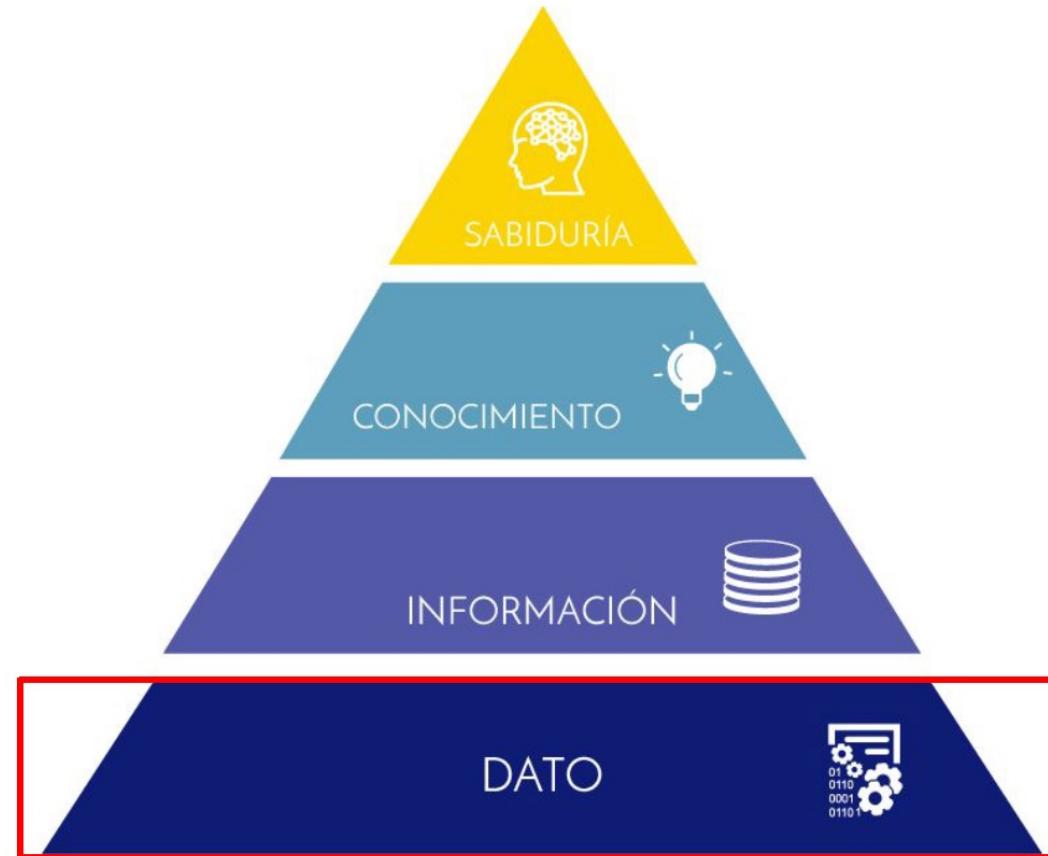
Científico de Datos

- Big Data Architect / Developer
- Data Engineer
- Data Analyst
- **Data Scientist**



Datos: pequeñas pinceladas

- 1. Identificarlos**
- 2. Obtenerlos**
- 3. Transformarlos**
- 4. Almacenarlos**
- 5. Consumirlos**
- 6. Analizarlos**



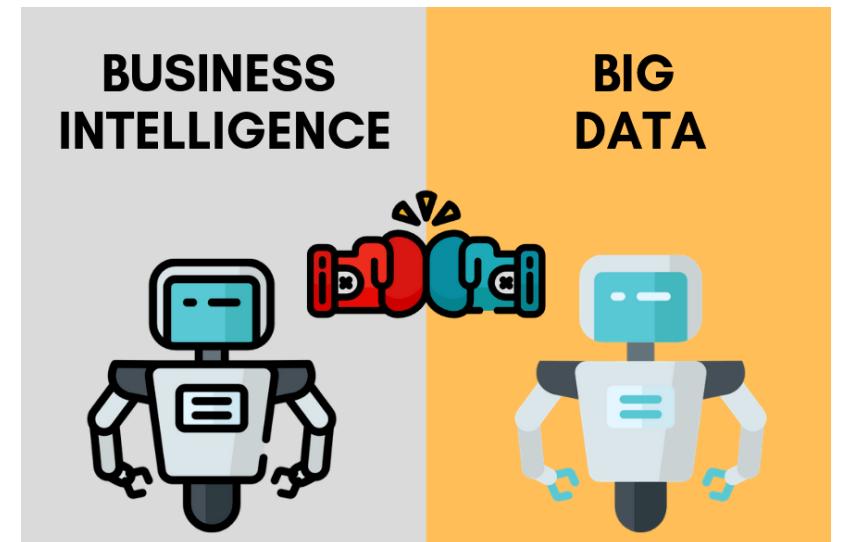
Business Intelligence

¿BI vs Big Data / Data Science?



Ambas se alimentan de datos masivos
pero ...

- **Big Data:** Captura, almacenamiento y procesamiento de datos masivo
- **Data Science:** Análisis predictivo y prescriptivo de esos datos
- **Business Intelligence:**
Aprovechamiento de los datos para optimizar las decisiones y el *reporting* de una compañía



Bases de datos

¿De dónde obtener los datos?

Kaggle -> [Regístrate aquí](#)



Programación

Python, ¿Por qué?



- **Más fácil**
- **Multipropósito**
- **Sencillo de interpretar**
- **Muchas librerías**
- **Tipado fuerte y dinámico**

Además, super interesante para:

- **IA, para hacer modelos de ML y algoritmos**
- **Big Data: Transformaciones de datos (ETL) y pequeños scripts**

Ejercicio

- ¿Qué es una **regresión lineal**?
- ¿Y un **dato perdido**?
- ¿Qué significa **entrenar un modelo**?
- ¿Qué es un **algoritmo**?
- ¿Y **minibatch**?

Duración estimada: 30min



Ejercicio

Recordemos Python!!

Notebook:

1. Estructuras de Datos en Python

Duración estimada: 1h



Introducción al Aprendizaje Automático

Introducción al Aprendizaje Automático

Motivación

La Era del ‘big data’

Estamos presenciando una explosión de aplicaciones donde el análisis de datos juega un papel fundamental. Tan importantes como las fuentes de datos son los algoritmos que extraen la información relevante.

Ejemplos de aplicaciones:

- Máquinas de búsqueda en Internet
- Biomedicina (clasificación automática de pacientes y enfermedades)
- Predicción de mercados financieros / Inversión
- Sistemas de recomendación

Introducción al Aprendizaje Automático

Motivación

El contenido que ves en la redes sociales



Introducción al Aprendizaje Automático

Motivación

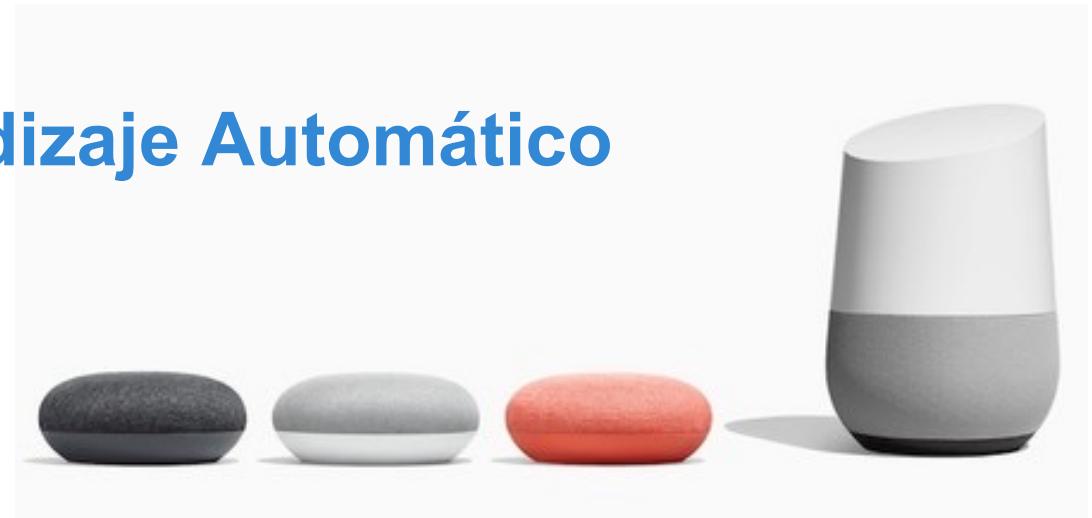
Recomendaciones de productos en tiendas online



Introducción al Aprendizaje Automático

Motivación

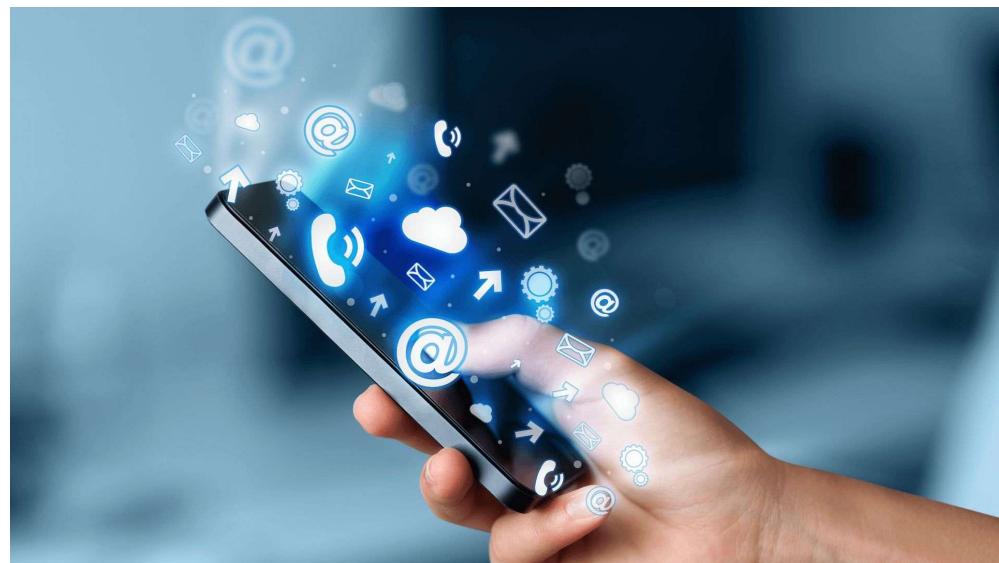
Tu asistente de voz



Introducción al Aprendizaje Automático

Motivación

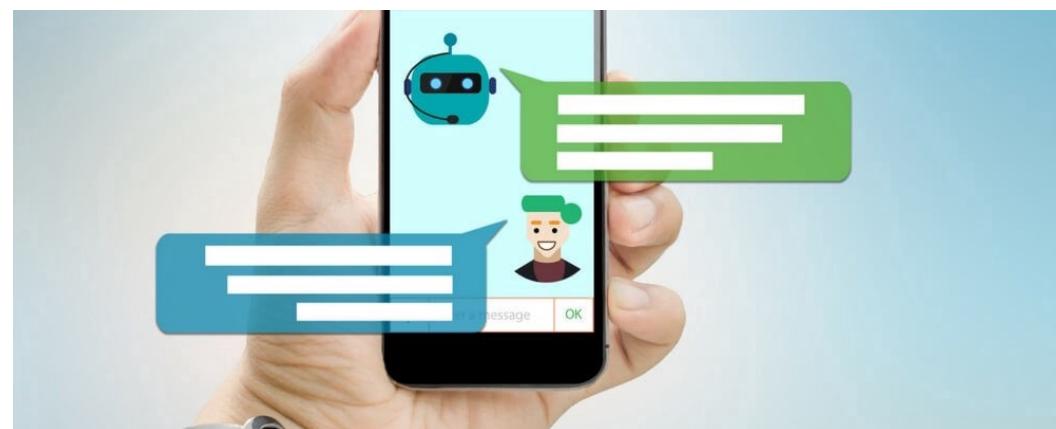
Nuestros smartphones



Introducción al Aprendizaje Automático

Motivación

Sistemas de atención al cliente



Introducción al Aprendizaje Automático

Motivación

Filtros de Spam



Introducción al Aprendizaje Automático

Motivación

Automatizaciones en el hogar



Introducción al Aprendizaje Automático

¿Qué es el Aprendizaje automático?

Aprendizaje Automático o Machine Learning (ML):

Es un conjunto de métodos que pueden automáticamente detectar patrones en datos, y utilizar los patrones descubiertos para predecir datos futuros, o realizar otros tipos de toma de decisiones bajo incertidumbre

All models are wrong, but some models are useful.
— George Box, 1987

Introducción al Aprendizaje Automático

¿Qué es el Aprendizaje automático?

Utiliza herramientas de teoría de probabilidad.

La teoría de la probabilidad se puede aplicar a cualquier problema que involucre incertidumbre.

- *¿Cuál es la mejor predicción sobre el futuro dados algunos datos del pasado?*
- *¿Cuál es el mejor modelo para explicar ciertos datos?*
- *¿Cuál medida debería ser la siguiente?*

Muy relacionado con estadística

Ejercicio

Individualmente

Diferencias entre:

- Machine Learning
- Artificial Intelligence
- Deep Learning
- ¿Qué son?

Duración estimada: 10min



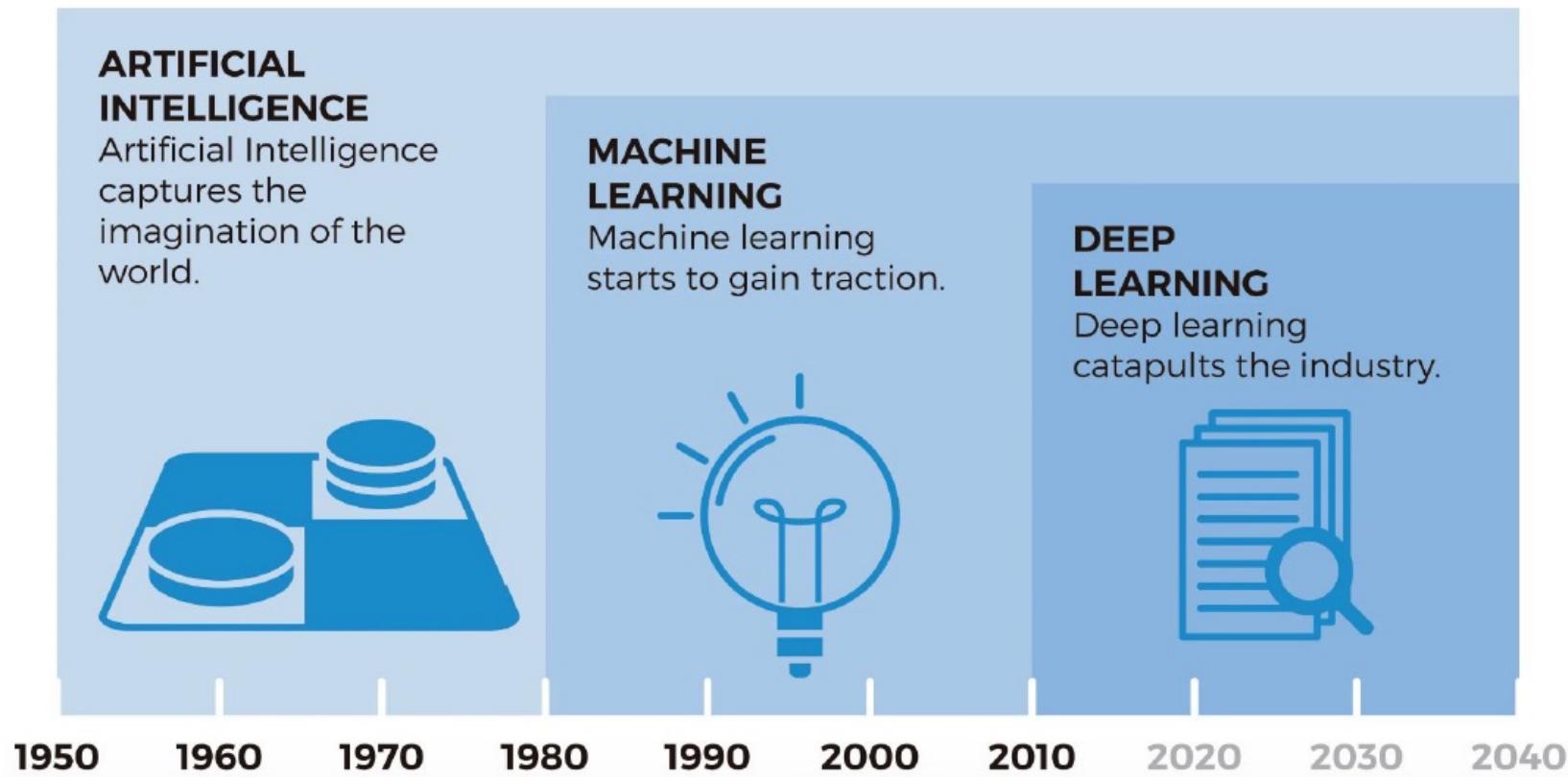
Machine Learning e Inteligencia Artificial

AI – ML – DL

- **Artificial Intelligence (AI) o Inteligencia Artificial:** Es una ciencia que tiene como objetivo imitar la mente humana.
- **Machine Learning (ML) o Aprendizaje Automático:** Trabaja con algoritmos que permiten conseguir el aprendizaje automático a través de los datos. Es el resultado de datos + variables + algoritmos.
- **Deep Learning (DL) o Aprendizaje Profundo:** Es un tipo de Machine Learning donde la máquina aprende por sí sola a través de un gran conjunto de algoritmos que imitan la red de neuronas del cerebro humano.

Machine Learning e Inteligencia Artificial

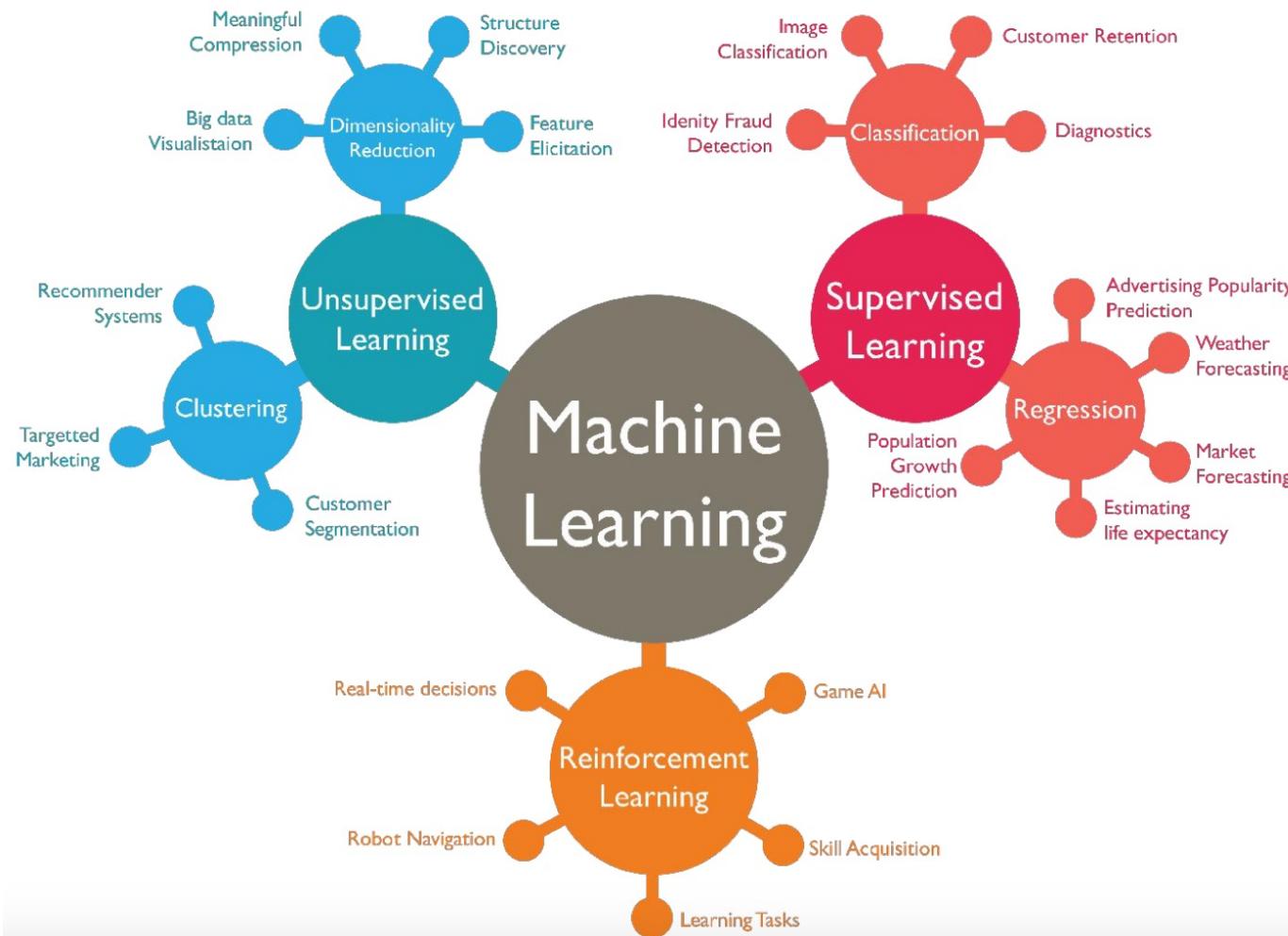
AI – ML – DL



Conceptos fundamentales del ML

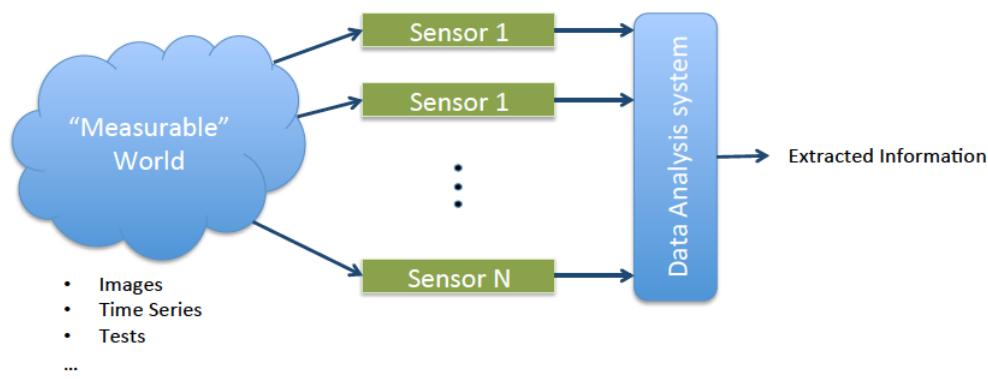
Conceptos fundamentales del ML

Clasificación de los modelos de ML



Conceptos fundamentales del ML

Diagrama de bloques de un sistema de aprendizaje



- La información deseada no se puede acceder directamente, así que hay que usar una serie de variable estadísticas relacionadas.
- El análisis de datos explota esta información estadística para proporcionar resultados precisos...
- ...pero algunos errores son generalmente inevitables.

Conceptos fundamentales del ML

Elementos básicos

Construir modelos que se ajustan a una colección de datos

- **Modelo:** objeto de una clase (programa informático) o función matemática (modelo estadístico) con parámetros libres.
- **Conjunto de entrenamiento:** Conjunto de ejemplos sacados de la distribución de datos que se pretende modelar.
- **Optimizador:** Método que ajusta los valores de los parámetros libres del modelo (entrena) para que capturen patrones informativos contenidos en los datos.

Conceptos fundamentales del ML

Objetivo

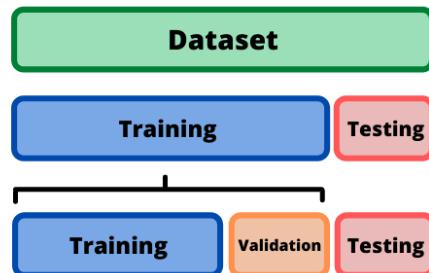
Conseguir que el modelo pueda hacer predicciones más o menos correctas cuando se le presenten datos que no se hayan usado durante el entrenamiento.

Generalización

Evitar el **sobreentrenamiento**, que provoca un error pequeño en entrenamiento pero elevado en test.

Conceptos fundamentales del ML

Conjuntos de...



Test

Conjunto de datos que nunca se usa durante el entrenamiento ni durante la optimización de los parámetros, sólo para calcular la puntuación de test tras la finalización de dicho proceso.

Entrenamiento

Conjunto de datos usado durante el entrenamiento del modelo.



Validación

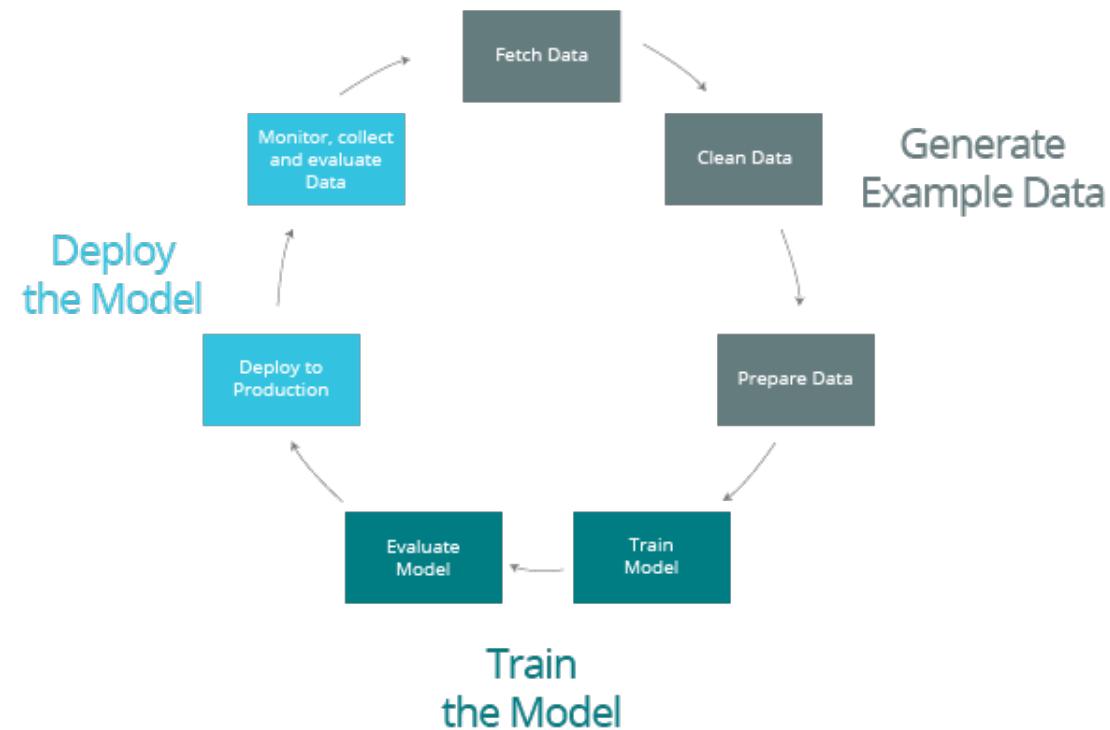
Conjunto de datos usado para evaluar el modelo durante el entrenamiento y elegir el mejor valor para los parámetros, los que obtengan mejor puntuación en dicho conjunto.



Conceptos fundamentales del ML

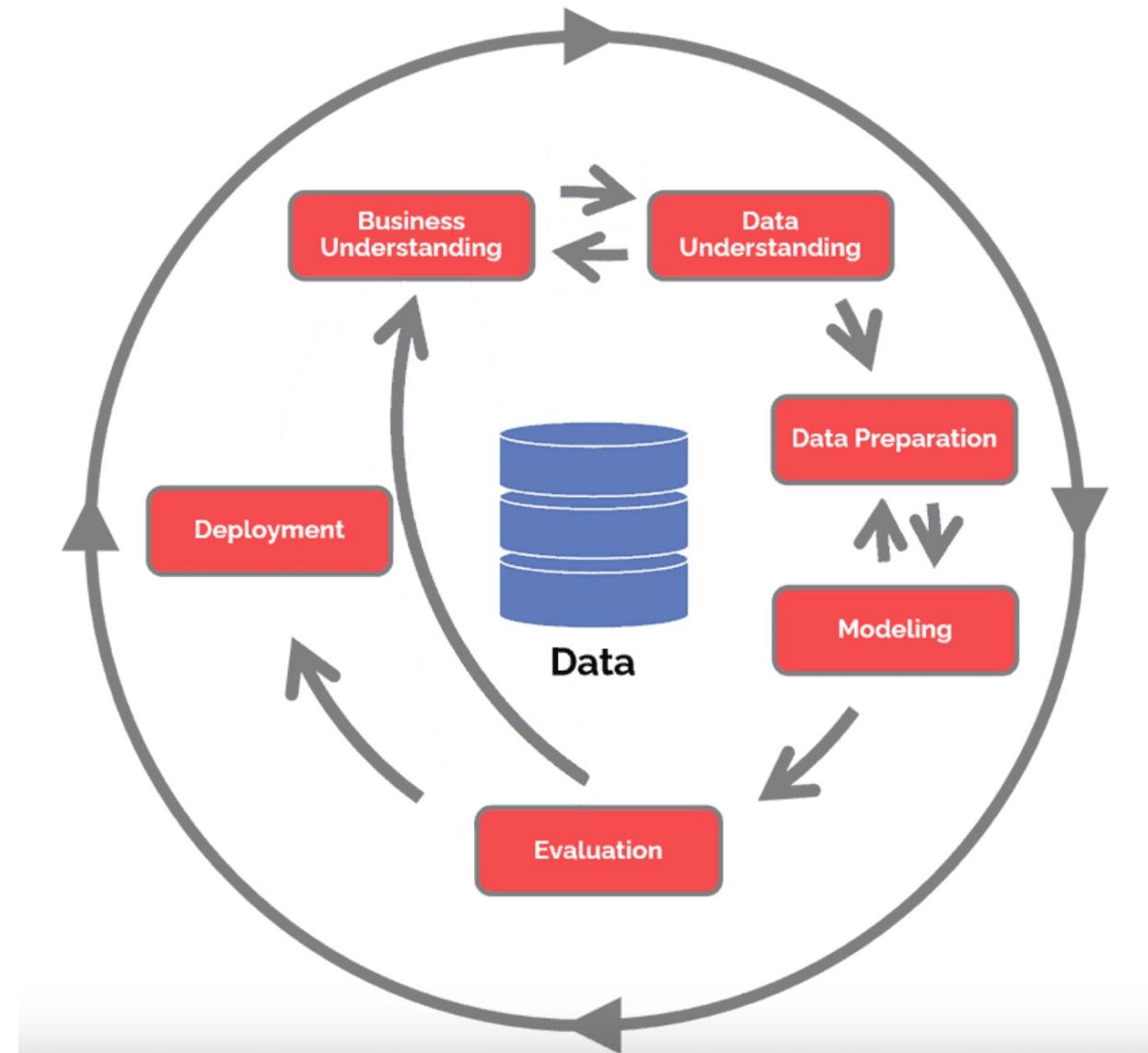
Pasos

Machine Learning Pipeline



Conceptos fundamentales del ML

CRISP-DM

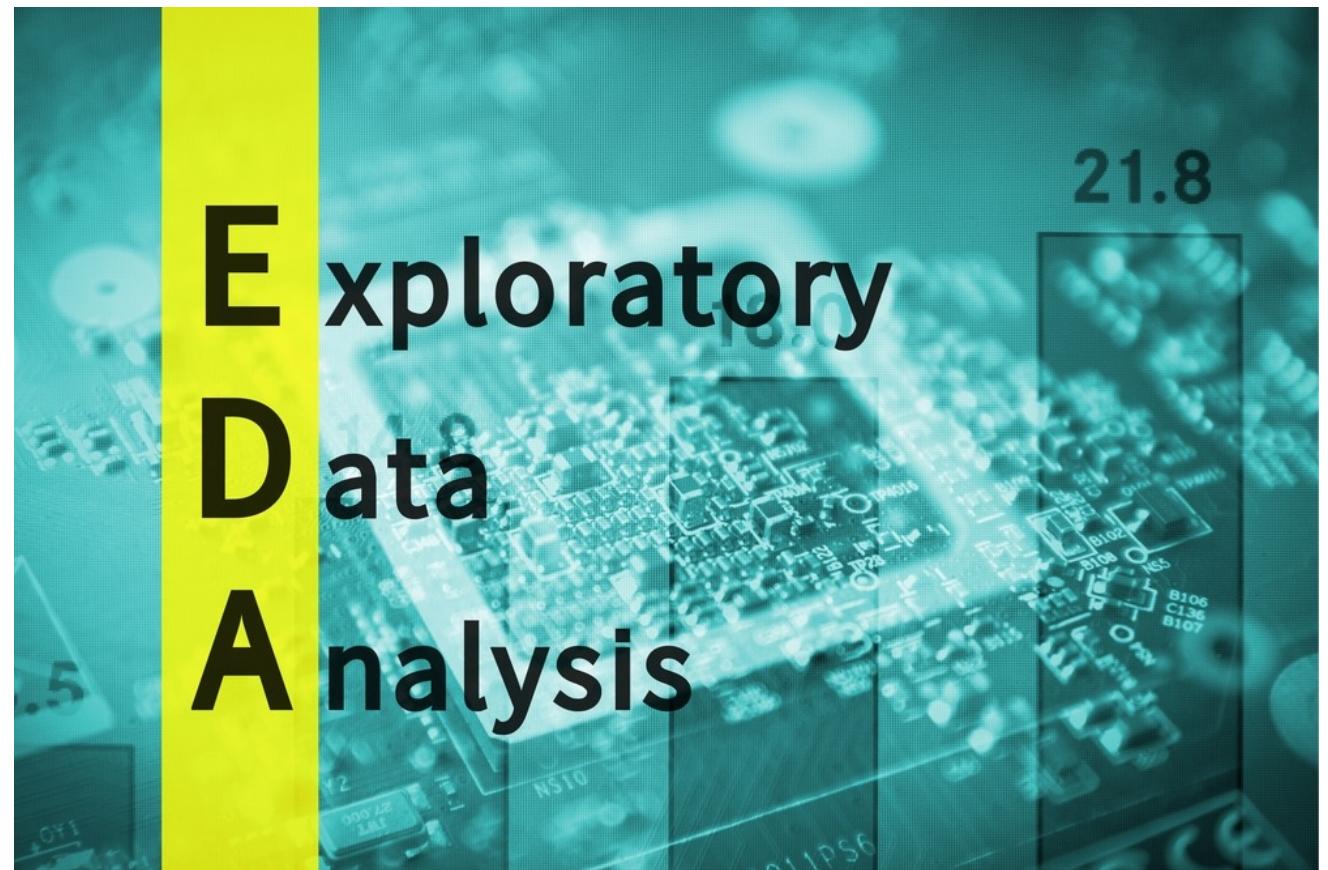


Repasemos EDA y Data Preparation

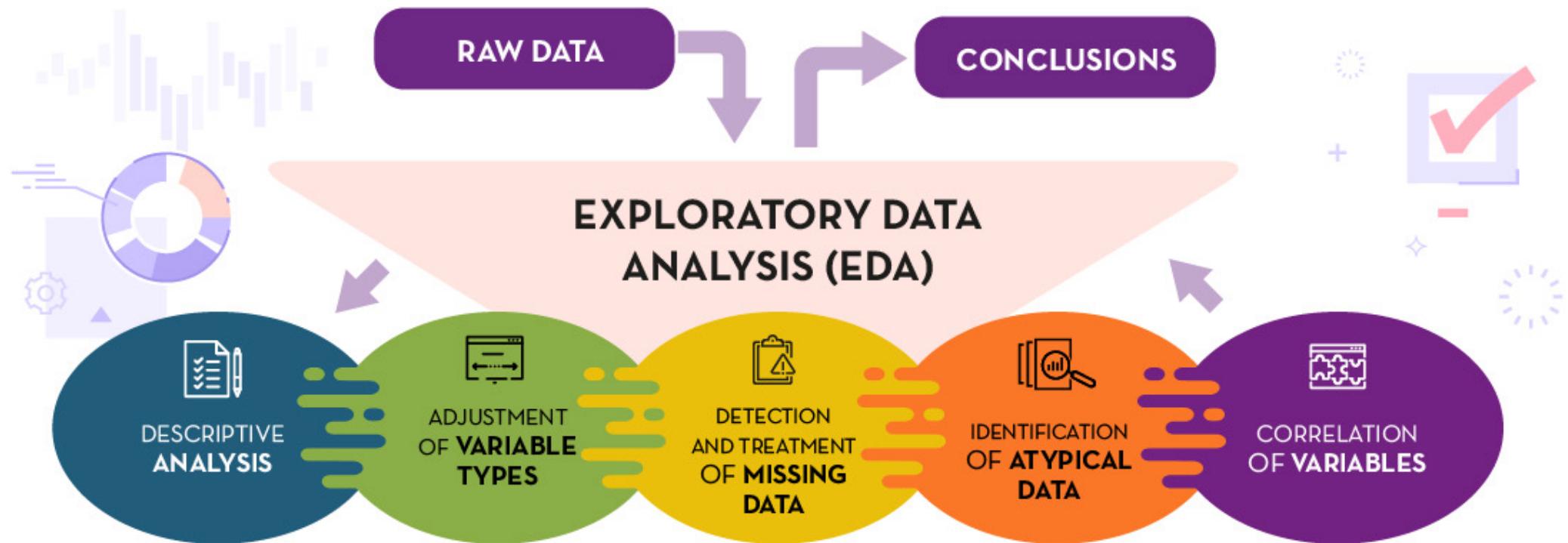
Conceptos fundamentales del ML

EDA

1. Conocer los datos
2. Identificar patrones
3. Detectar *outliers*



EDA



Definiciones importantes

Observaciones

También se les denomina **registros**.

Hace referencia a cada una de las **filas** de la base de datos.

Características/Variables

También se les denomina **features**.

Hace referencia a cada una de las **columnas** en una base de datos.

Definiciones importantes

Missing data

Son los valores perdidos de la base de datos.

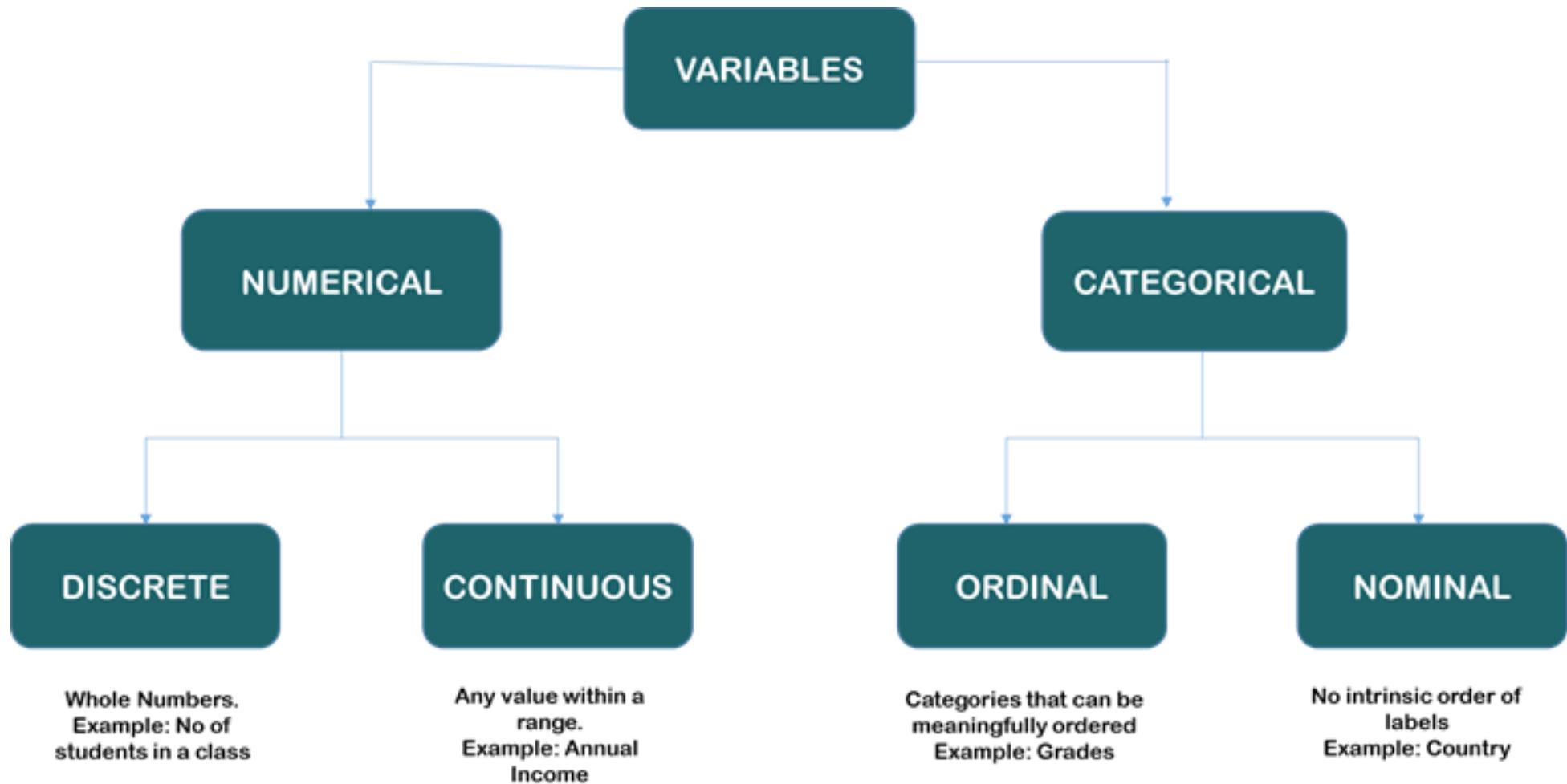
Se les puede llamar **nulos**.

A veces son campos vacíos, NaN, None, 0, -1, ...

Outliers

Son observaciones de la base de datos que se alejan de la distribución del resto. Es decir, son muestras muy diferentes a las demás.

Variables



Variables

Tipos de datos

Obtener el tipo de datos:

- **Int**
- **Float**
- **Double**
- **String**
- **Bool**

Data preparation

Proceso

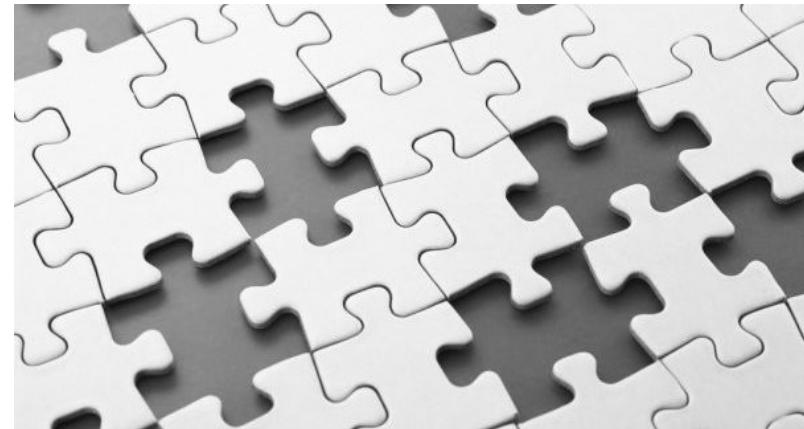
Una vez que tenemos los datos explorados....

Empezamos a detectar:

- **Nulos (missing)**
- **Valores atípicos (Outliers)**
- **Datos incompletos**
- **Datos erróneos**

Data preparation

Nulos



Debemos detectarlos y colocar una señal que identifiquemos en nuestro código como *nulos*.

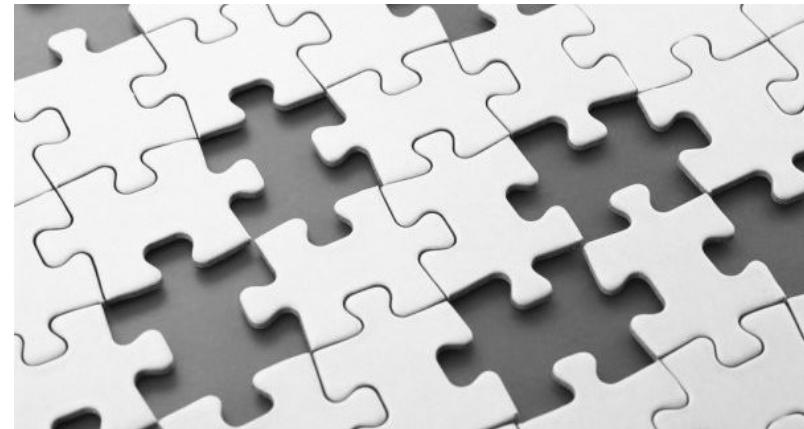
Unificar todos los nulos de la misma forma.

Student id	Marks in Maths (out of 100)	Marks in Maths (out of 100)	Remarks
19060641015	45	68	Good
19060641016	53	53	Bad
19060641017	68	78	Good
19060641018	75	75	Good
19060641019	80	45	Poor
19060641020	82	82	Good
19060641021	49	NULL	<u>NaN</u>
19060641022	76	80	Good
19060641023	79	79	Good
19060641024	55	55	Average
19060641025	80	52	Bad
19060641026	N/A	NULL	<u>NaN</u>
19060641027	N/A	87	<u>NaN</u>
19060641028	N/A	NULL	<u>NaN</u>

[NB: A dataset is a collection data points. Like age, weight etc.]

Data preparation

Nulos



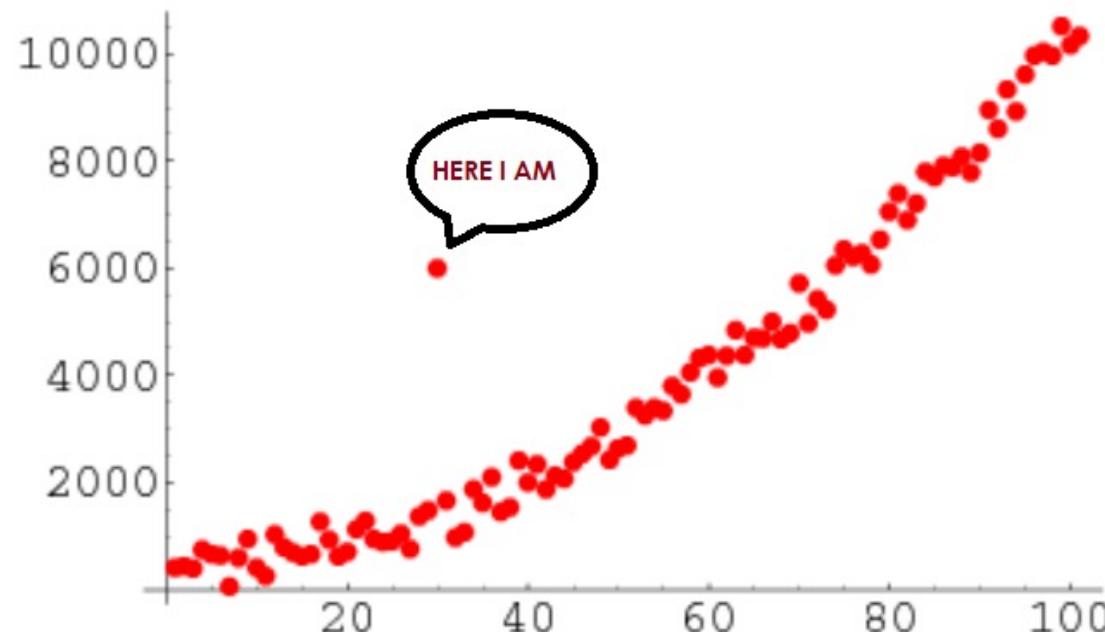
Opciones que podemos hacer:

- **Completar con una constante**
- **Completar con la media o la mediana**
- **Completar con el valor más frecuente**
- **Poner un flag (por ejemplo -999)**
- **Otro método de imputación**
- **Borrar las filas o columnas correspondientes (no recomendable!)**

Data preparation

Outliers

Valores fuera de la norma general / de la distribución de los datos.



Data preparation

Outliers

¿Cómo detectarlos?

$$Z\text{-score} = \frac{X - \bar{x}}{\sigma} = \text{stats.zscore()}$$

valores $> |3|$ están más lejos de μ que 99.7% del resto de los datos

Opciones que podemos hacer:

- Cambiarlos
- Eliminarlos

Data preparation

Incompletos

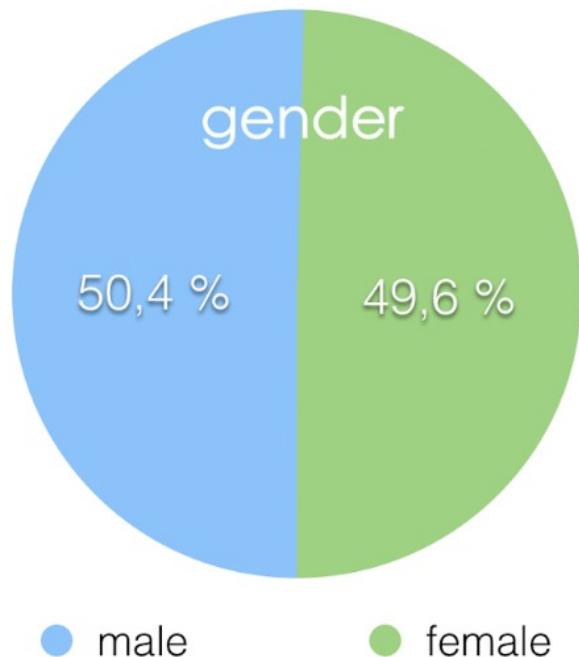
Hay veces que aún teniendo todo el histórico de los datos, no es suficientemente buena la calidad de los mismos y no es representativa la distribución que tenemos.

Datos no balanceados

Data preparation

Incompletos

Balanced Dataset



Unbalanced Dataset

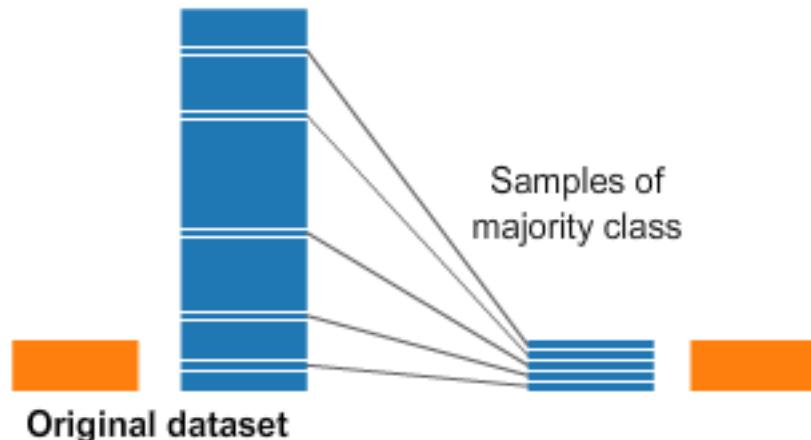


Data preparation

Incompletos

¿Qué hacemos?

Undersampling



Oversampling



Data preparation

Erróneos

Una mala calidad en los datos también se ve reflejada en errores en las bases de datos.

- ¿Cómo son los datos?
- ¿Cuál es el rango de valores posible?
- ¿Existe un orden?



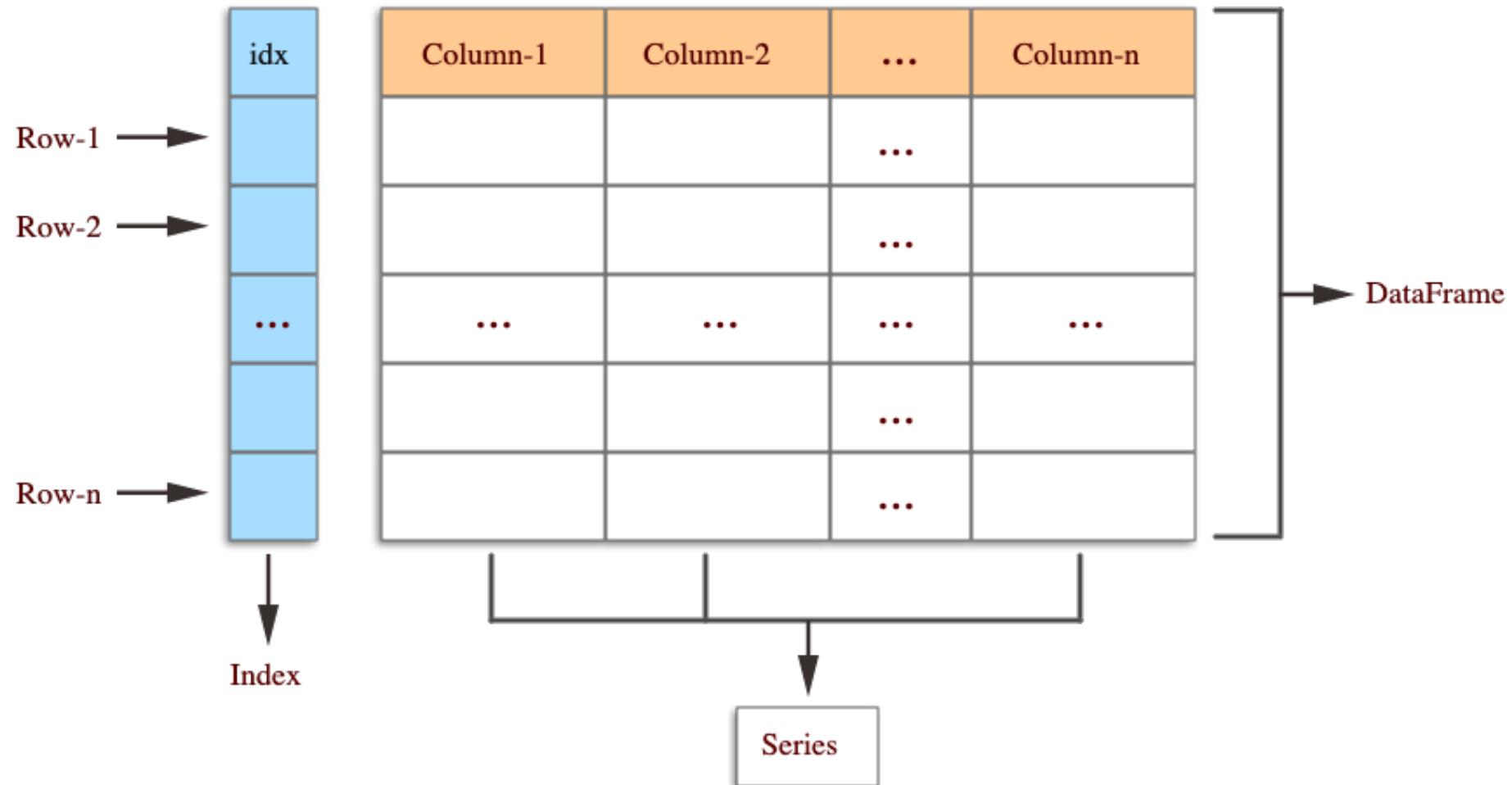
Data preparation

**BAD DATA IS NO
BETTER THAN NO
DATA.**



Pandas

Pandas Data structure



Práctica

Repasemos Pandas con Python e incorporemos algún concepto nuevo de ML?

Notebook:

2. Preprocesado de un dataset con Pandas

Duración estimada: 1h



Herramientas para el ML

Herramientas para el ML

Álgebra

Para el tratamiento de los datos vamos a trabajar con **vectores** (1D), **matrices** (2D) y **tensores** (mayores dimensiones) y operaciones entre ellos.

Cálculo

Es imprescindible en el paso de la optimización de una función de pérdidas para ajustar los parámetros del modelo.

Programación

Es la herramienta para analizar los datos, entrenar los modelos y visualizar los resultados. Utilizaremos **Python**.

Teoría de la probabilidad

Es la base teórica de un modelo de ML para **cuantificar la incertidumbre**.



Herramientas para el ML

Motivación

- Aplicación fundamental en el área de las matemáticas.
- Álgebra lineal, cálculo y probabilidad son los ‘lenguaje’ en los que el ML está escrito.
- El aprendizaje de estas materias nos ayudará a entender los algoritmos y a modificarlos o crear unos nuevos.
- Forma la columna vertebral de muchos algoritmos de ML.

Exploración y análisis de datos

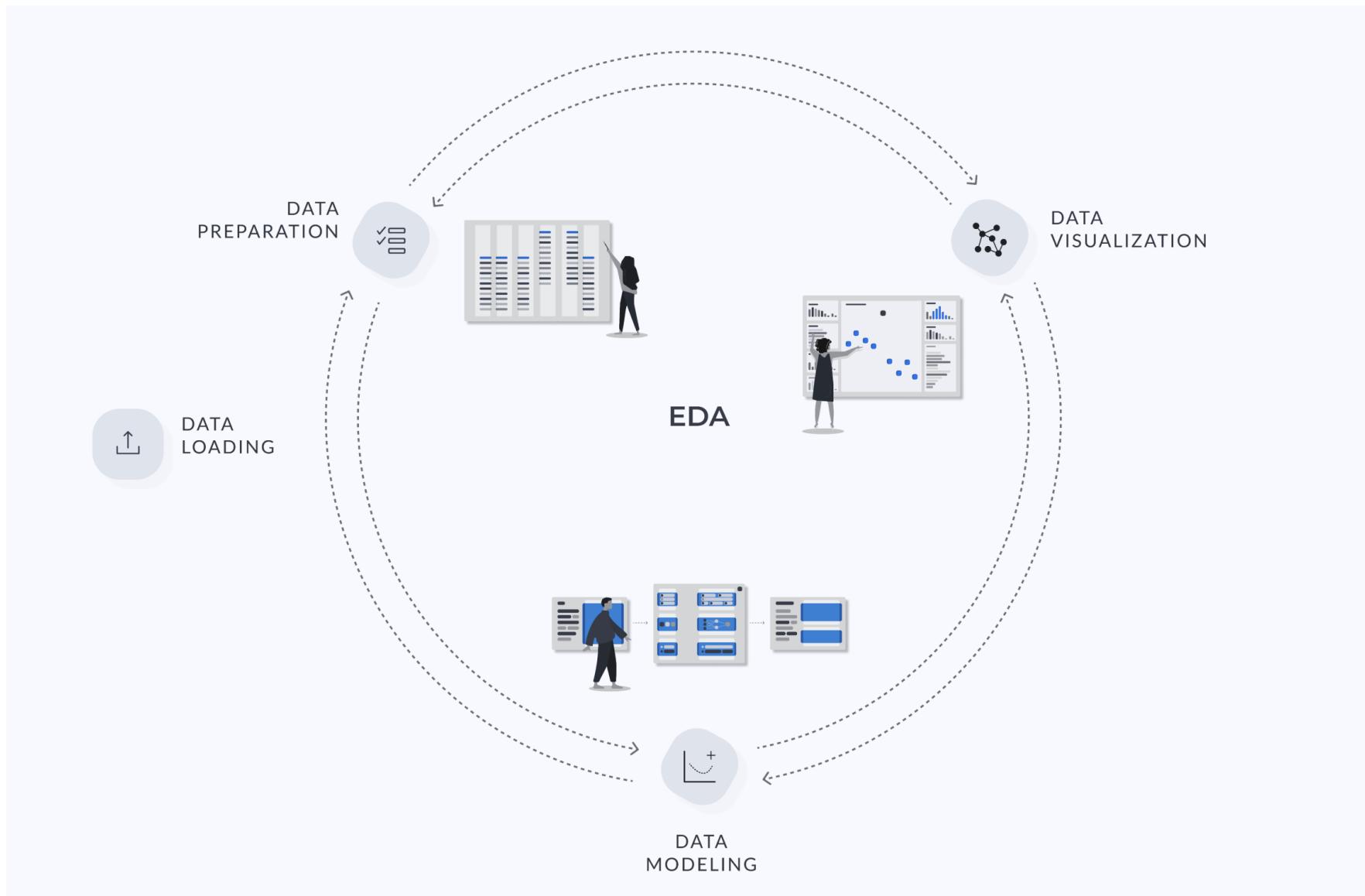
Programación

Programación

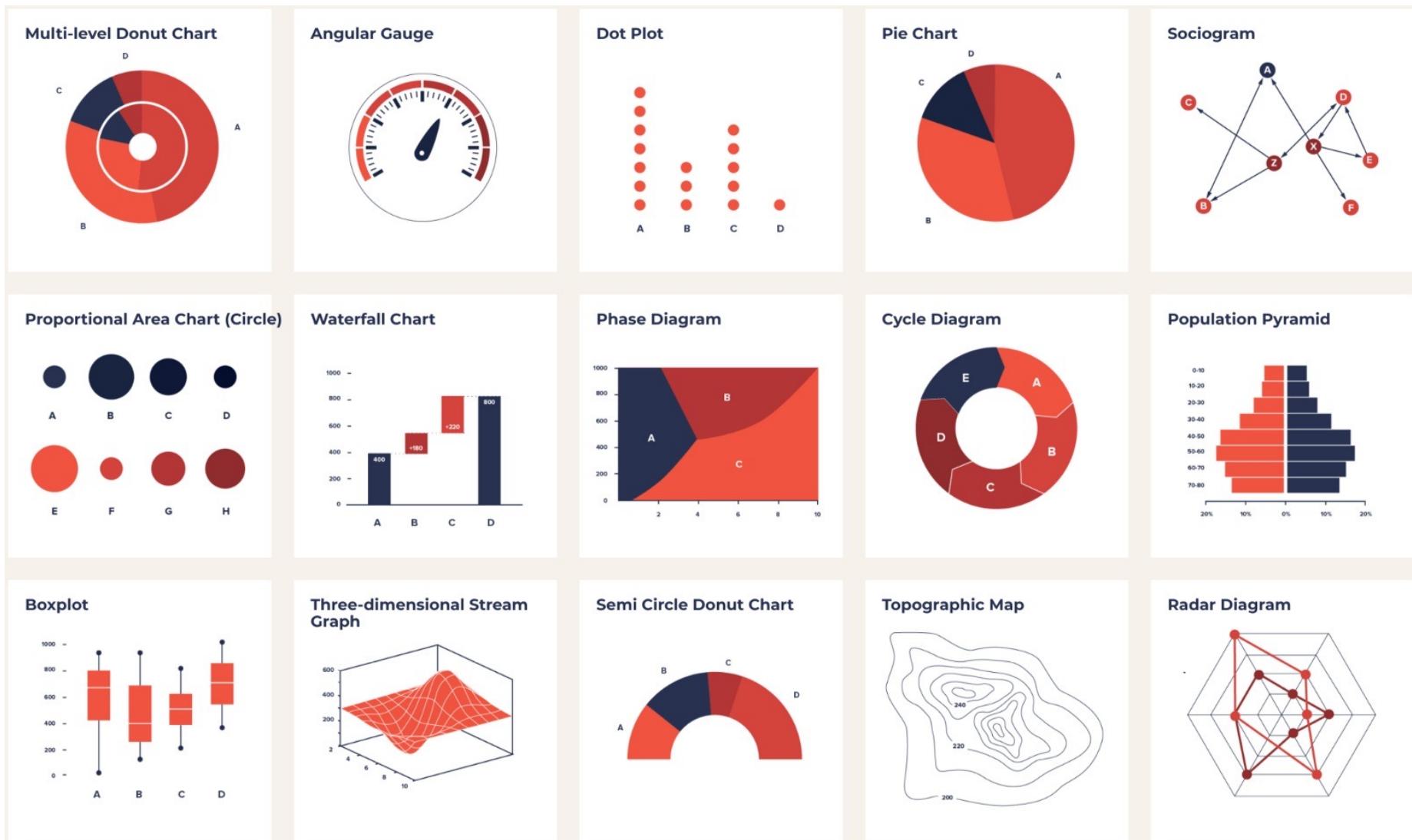
Python

- **Usaremos notebooks de python para estudiar todo el módulo, ver la teoría, procesado de datos y aplicación del modelo de ML con ejemplos prácticos.**
- **Ya los conocéis del módulo anterior. Google Colab.**

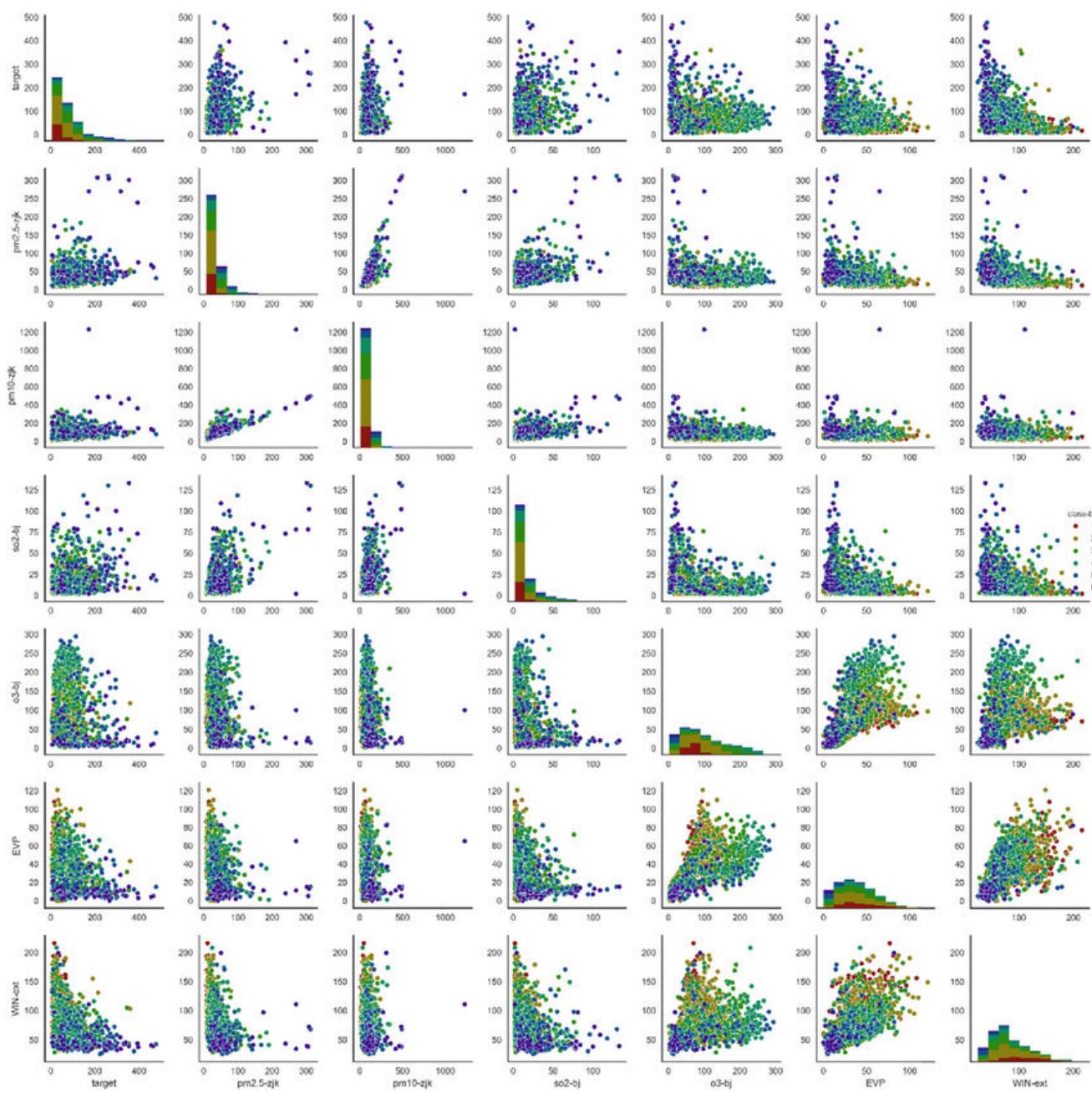
Visualización



Visualización



Visualización



Álgebra lineal en ML

Álgebra lineal en ML

Motivación

- Vectorización como una manera de paralelizar operaciones.
- Notación de bucle reformulada con ecuaciones matriciales ofreciendo ganancias en eficiencia computacional.
- Uso en librerías de Python como Numpy, SciPy, Scikit-Learn, Pandas, tensorflow o Pytorch.
- Las GPUs se han diseñado para realizar operaciones de álgebra lineal optimizadas.
- El crecimiento explosivo de *Deep Learning* se debe en parte de la naturaleza de paralelización en los algoritmos sobre hardware GPU.

Álgebra lineal en ML

Factorización de matrices

- Problemas: **overflow** (desbordamiento) y **underflow** son los límites de representar computacionalmente números extremadamente grandes o pequeños.
- Solución: Por ejemplo usando técnicas de factorización de matrices.
- Permiten representar matrices en otras más estructuradas y simples que tiene propiedades computacionales muy útiles.
- Las descomposiciones **LU** o **SVD** son componentes intrínsecos de algoritmos como *Linear Least Squares (LLS)* o *Principal Components Analysis (PCA)*, que veremos al final del módulo.

Álgebra lineal en ML

Vectores y matrices

Son las entidades primarias, y son ejemplos de una entidad más general conocida como un **tensor**.

- **Escalar:** Tensor de orden cero. Definimos el conjunto al que pertenece.
Ejemplo: $x \in \mathbb{R}$, $x \in \mathbb{N}$, $x \in \mathbb{Z}$

- **Vector:** Tensor de una dimensión. Miembros de espacios vectoriales.
Ejemplo: $x \in \mathbb{R}^3$

En ML, los vectores representan las características de los datos.

Ejemplo: la importancia de cada palabra que forma un documento, la intensidad de los pixels en una imagen o los valores de precios históricos para una muestra de instrumentos financieros.

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Álgebra lineal en ML

Vectores y matrices

- **Matriz:** Tensor de 2 dimensiones, $m \times n$, con m filas y n columnas.

Ejemplo: $A \in \mathbb{R}^{m \times n}$

Por defecto un vector es una matriz $1 \times n$. Definimos como vector columna a una matriz $m \times 1$

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \dots & a_{mn} \end{bmatrix}$$

- **Tensor:** es el término general para más de 2 dimensiones. Muy utilizado en Deep Learning, por ejemplo para los parámetros de las redes neuronales o para describir datos como una imagen con datos de intensidad en múltiples canales (colores RGB).

Operaciones con matrices

(Material de refuerzo)

Álgebra lineal en ML

Suma de matrices

- Posible entre matrices de mismas dimensiones.
- Propiedad conmutativa y asociativa
- $C = A + B$, donde $c_{ij} = a_{ij} + b_{ij}$

$$A + B = \begin{bmatrix} 1 & 4 & 17 \\ 18 & 3 & 2 \\ 5 & 19 & 1 \end{bmatrix} + \begin{bmatrix} 12 & 18 & 6 \\ 4 & 3 & 33 \\ 23 & 5 & 8 \end{bmatrix} = \begin{bmatrix} 13 & 22 & 23 \\ 22 & 6 & 35 \\ 28 & 24 & 9 \end{bmatrix} = C$$

- Posible entre matrices y escalares.
- $B = x + A$, donde $b_{ij} = x + a_{ij}$
- **Broadcasting:** Entre matriz $m \times n$ dimensional y vector m dimensional.
- $B = A + x$, donde $b_{ij} = a_{ij} + x_j$

Álgebra lineal en ML

Matrix transpuesta

- **Transpuesta** de una matriz, intercambio de filas por columnas.

$$\mathbf{A}^T = [a_{ji}]_{n \times m}$$

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}, \quad \mathbf{A}^T = \begin{bmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \\ a_{13} & a_{23} \end{bmatrix}$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad \mathbf{x}^T = [x_1 \quad x_2 \quad x_3]$$

Álgebra lineal en ML

Multiplicación de matrices

- Producto **entre matrices**.
- $\mathbf{A} = [a_{ij}]_{m \times n}$, $\mathbf{B} = [b_{ij}]_{n \times p}$, $\mathbf{C} = \mathbf{AB} = [c_{ij}]_{m \times p}$
- $\mathbf{AB} \neq \mathbf{BA}$ (no commutativa)

$$c_{ij} = \sum_{k=1}^n a_{ik} b_{kj}$$

$$\mathbf{A} = \begin{bmatrix} 2 & 5 & 1 \\ 7 & 3 & 6 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 8 \\ 9 & 4 \\ 3 & 5 \end{bmatrix}$$

$$\mathbf{AB} = \begin{bmatrix} 2 \cdot 1 + 5 \cdot 9 + 1 \cdot 3 & 2 \cdot 8 + 5 \cdot 4 + 1 \cdot 5 \\ 7 \cdot 1 + 3 \cdot 9 + 6 \cdot 3 & 7 \cdot 8 + 3 \cdot 4 + 6 \cdot 5 \end{bmatrix} = \begin{bmatrix} 50 & 41 \\ 52 & 98 \end{bmatrix}$$

$$\mathbf{BA} = \begin{bmatrix} 1 \cdot 2 + 8 \cdot 7 & 1 \cdot 5 + 8 \cdot 3 & 1 \cdot 1 + 8 \cdot 6 \\ 9 \cdot 2 + 4 \cdot 7 & 9 \cdot 5 + 4 \cdot 3 & 9 \cdot 1 + 4 \cdot 6 \\ 3 \cdot 2 + 5 \cdot 7 & 3 \cdot 5 + 5 \cdot 3 & 3 \cdot 1 + 5 \cdot 6 \end{bmatrix} = \begin{bmatrix} 58 & 29 & 49 \\ 46 & 57 & 33 \\ 41 & 30 & 33 \end{bmatrix}$$

Álgebra lineal en ML

Multiplicación de matrices

- Producto entre **matriz y escalar**.
- $\mathbf{A} = [a_{ij}]_{m \times n}, \lambda \in \mathbb{R}, \lambda\mathbf{A} = [\lambda a_{ij}]_{m \times n}$

$$\lambda(\mathbf{A} + \mathbf{B}) = \lambda\mathbf{A} + \lambda\mathbf{B}$$

$$(\lambda + \mu)\mathbf{A} = \lambda\mathbf{A} + \mu\mathbf{A}$$

$$\lambda(\mu\mathbf{A}) = (\lambda\mu)\mathbf{A}$$

Álgebra lineal en ML

Multiplicación de matrices

- **Producto Hadamard** (element-wise).
- $A = [a_{ij}]_{m \times n}$, $B = [b_{ij}]_{m \times n}$, $C = A \odot B = [a_{ij}b_{ij}]_{m \times n}$
- **Producto escalar** (entre dos vectores).

$$\mathbf{x}^T \mathbf{y} = \sum_{i=1}^n x_i y_i = \mathbf{y}^T \mathbf{x}$$

Álgebra lineal en ML

Ecuaciones lineales

$$\begin{aligned}x + 2y + 4z &= 10 \\3x + y - 5z &= -8 \\4x - 3y + 7z &= 4\end{aligned}$$

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 4 \\ 3 & 1 & -5 \\ 4 & -3 & 7 \end{bmatrix}$$

$$\mathbf{x} = [x, y, z]^T$$

$$\mathbf{b} = [10, -8, 4]^T$$

$$\mathbf{Ax} = \mathbf{b}$$

$$\begin{bmatrix} 1 & 2 & 4 \\ 3 & 1 & -5 \\ 4 & -3 & 7 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 10 \\ -8 \\ 4 \end{bmatrix}$$

$$\mathbf{x} = \frac{\mathbf{b}}{\mathbf{A}} = \frac{1}{\mathbf{A}}\mathbf{b} = \mathbf{A}^{-1}\mathbf{b}$$

Álgebra lineal en ML

Matriz identidad

$$\mathbf{I}_n = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}$$

$$\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}_n = \mathbf{AA}^{-1}$$

$$\mathbf{I}_n\mathbf{A} = \mathbf{A}\mathbf{I}_n = \mathbf{A}$$

$$\begin{aligned} \mathbf{Ax} &= \mathbf{b} \\ \mathbf{A}^{-1}\mathbf{Ax} &= \mathbf{A}^{-1}\mathbf{b} \\ \mathbf{I}_n\mathbf{x} &= \mathbf{A}^{-1}\mathbf{b} \\ \mathbf{x} &= \mathbf{A}^{-1}\mathbf{b} \end{aligned}$$

Método de Gauss-Jordan para calcular la matriz inversa...

Ejercicio

Conoces la librería **numpy** de Python?

Notebook:
3. Librería Numpy

Duración estimada: 1h



Cálculo en ML

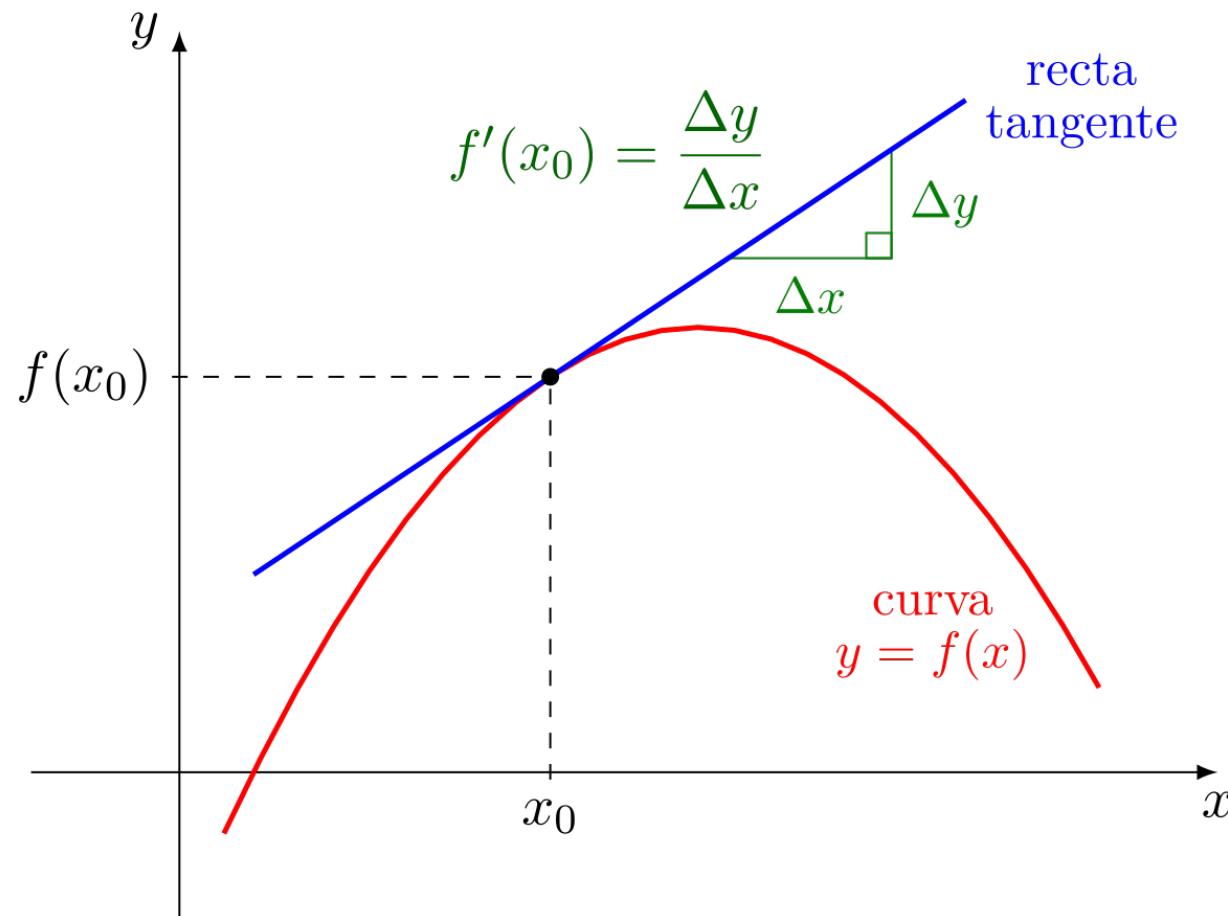
Cálculo en ML

Motivación

- **Optimización de una función de coste/error/pérdidas** para ajustar los parámetros del modelo.
- **Derivadas parciales**: cómo se altera la función de pérdidas con individuales cambios en cada parámetro.
- Las derivadas se agrupan en matrices para un cálculo más directo

Cálculo en ML

Derivadas



Cálculo en ML

Función de pérdidas/coste

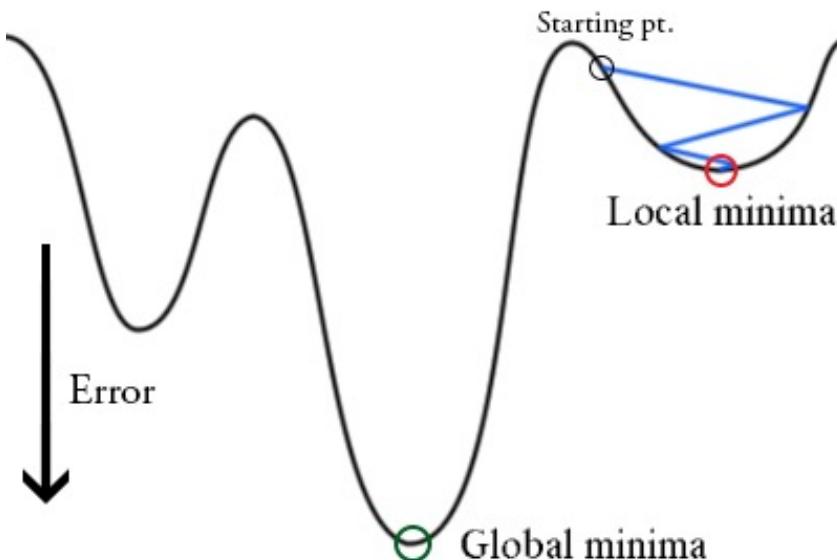
- Es una función que mapea el resultado de un modelo (predicción) en un **número real que representa el coste** asociado al modelo.
- El problema de **optimización** busca **minimizar** dicha función: calcular las derivadas e igualar a cero.
- Si en vez de función de pérdidas tenemos función de ganancias o de probabilidad, se buscará maximizar dicha función.
- Ejemplo: **MSE** (Mean Square Error o Error cuadrático medio)

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$$

Cálculo en ML

Optimización

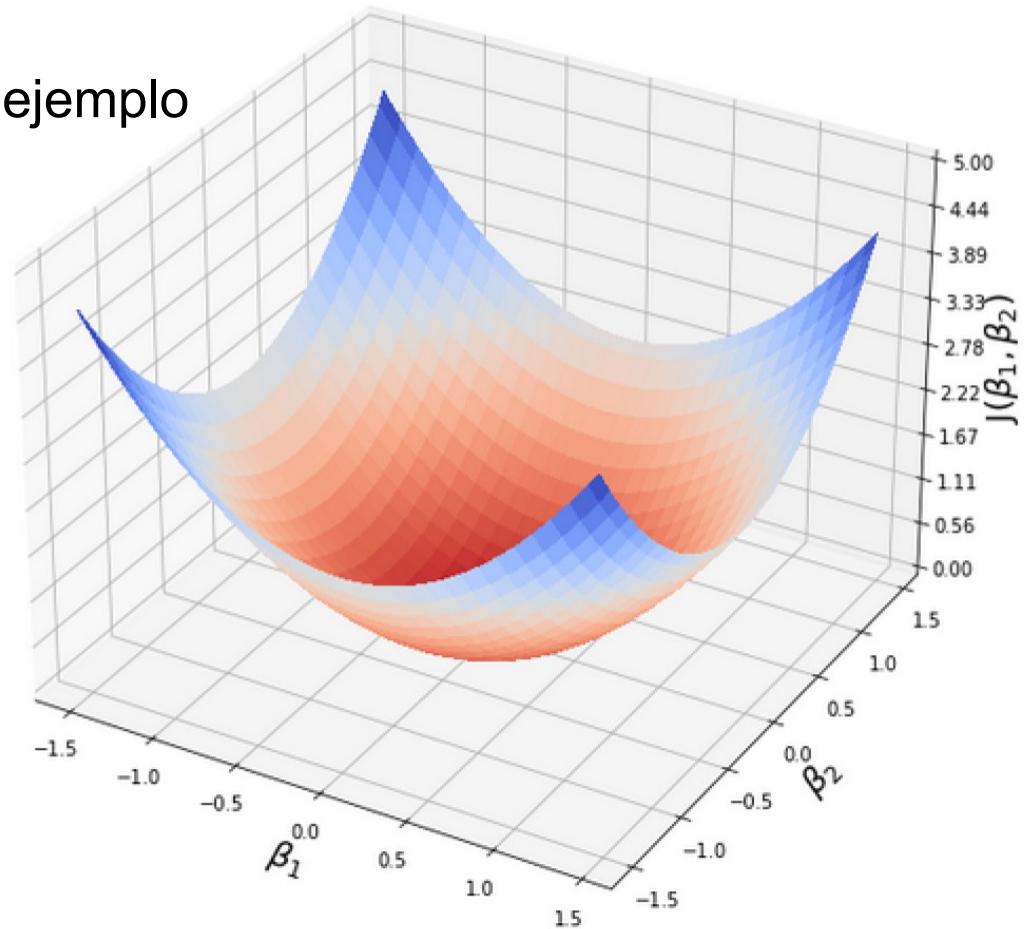
- Ejemplo: Buscar los parámetros que minimizan una función de pérdidas.



Cálculo en ML

Función de pérdidas

- En función de 2 parámetros, por ejemplo



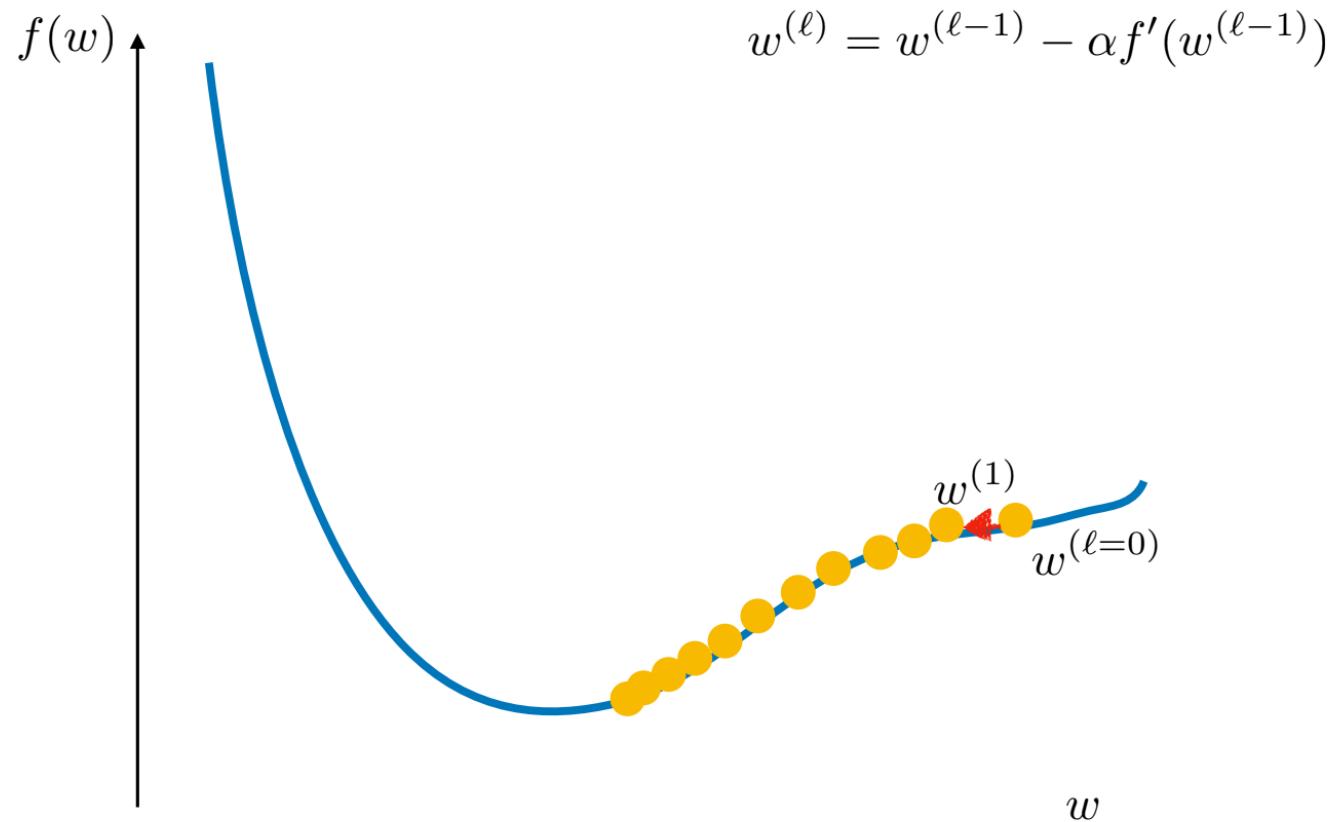
Cálculo en ML

Descenso por gradiente

- Cuando la optimización de una función **no tiene forma cerrada**: no podemos despejar los parámetros en la ecuación, hay que hacer uso de **algoritmos iterativos** que buscan esa solución del mínimo en la función.
- **Descenso por gradiente** es uno de ello, muy conocido y utilizado en ML.
- Tiene un parámetro muy importante que veremos en la siguiente transparencia, el '**learning rate**'.

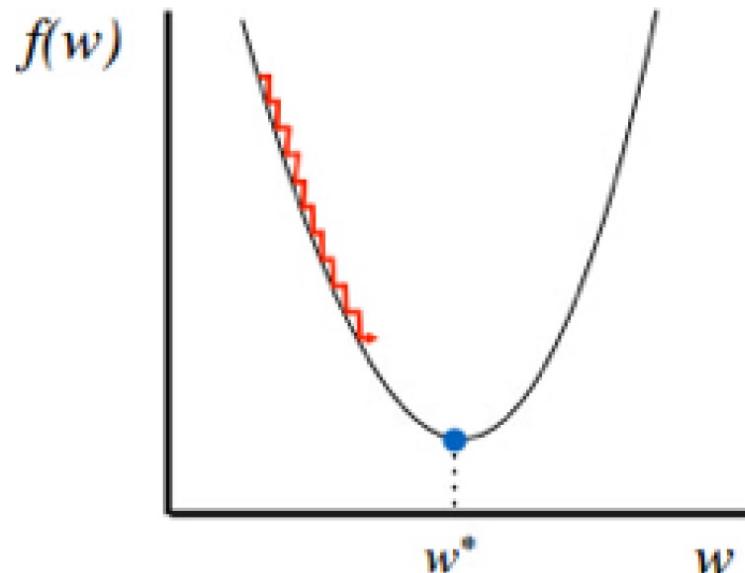
Cálculo en ML

Descenso por gradiente

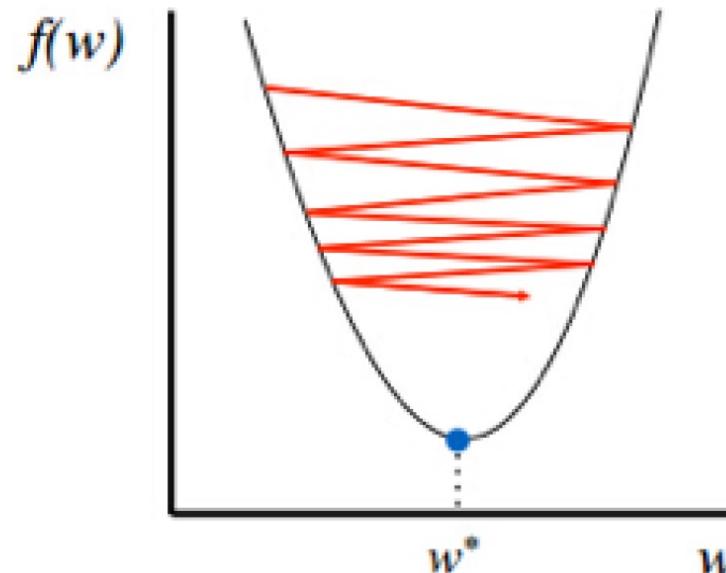


Cálculo en ML

Descenso por gradiente



Too small: converge
very slowly



Too big: overshoot and
even diverge

Ejemplo práctico

Ejemplo práctico

Ejemplo: Reconocimiento de dígitos manuscritos

Entrada del modelo

Imagen de 28x28 píxeles = vector x de 784 números reales

Salida del modelo

Identidad del dígito 0,...,9

Conjunto de entrenamiento: N dígitos:

$$\{x_1, \dots, x_N\}$$

Vector objetivo: Identidad de los dígitos

$$\{t_1, \dots, t_N\}$$

Resultado del algoritmo: expresado como...

$$y(x)$$



Ejemplo práctico

Ejemplo: Reconocimiento de dígitos manuscritos

Conjunto de test: El modelo debe ser capaz de determinar la identidad de nuevas imágenes de dígitos

Generalización: Capacidad del modelo para clasificar nuevos ejemplos diferentes de los usados en entrenamiento.

Preprocesado: Las imágenes de los dígitos de entrenamiento se trasladan y se escalan a una caja de tamaño fijo

Extracción de características / Feature extraction

Aplicar también a los datos de test



Ejercicio

Practiquemos con datasets reales el uso de las librerías más importantes. Leer datasets de ficheros, separar variables de entrada y salida, hacer gráficas e incluso entrenar el primer modelo.

Notebook:

4. Librerías para el análisis de datos

Duración estimada: 2h



Contacto

Correo: a.cobo.aguilera@gmail.com

LinkedIn: [Aurora Cobo Aguilera](#)

GitHub: [AuroraCoboAguilera](#)

Google Scholar: [Aurora Cobo Aguilera](#)





GOBIERNO
DE ESPAÑA

VICEPRESIDENCIA
PRIMERA DEL GOBIERNO

MINISTERIO
DE ASUNTOS ECONÓMICOS
Y TRANSFORMACIÓN DIGITAL

SECRETARÍA DE ESTADO
DE DIGITALIZACIÓN
E INTELIGENCIA ARTIFICIAL

red.es

Centro de
Referencia Nacional
en Comercio Electrónico
y Marketing
CRN
Digital



UNIÓN EUROPEA

“El FSE invierte en tu futuro”

Fondo Social Europeo


Barrabés

 The Valley