


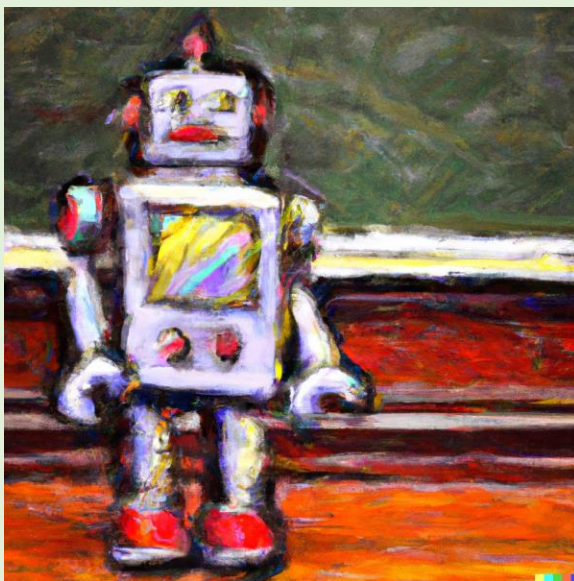


VI XORNADAS MONOGRÁFICAS  
APETEGA  
Asociación de Profesorado de Tecnología de Galicia

Juan David Rodríguez  
30 de Septiembre 2023

 @juandalibaba

# ¿Se puede enseñar Inteligencia Artificial en la Escuela?



Imágenes generadas con Dall-E

# ¿Qué es IA?

“la Inteligencia Artificial es todo aquello que tenga que ver con hacer que una máquina sea capaz de resolver problemas para cuya solución se necesita alguna característica de la inteligencia humana o animal, como el aprendizaje, el razonamiento o la adaptación”

Entender cómo pensamos para construir artefactos inteligentes

	<b>Humanidad</b>	<b>Racionalidad</b>
<b>Pensamiento</b>	Pensando como los humanos: Enfoque cognitivo, se apoya en la psicología	Pensando racionalmente: Las leyes del pensamiento, aproximación logicista
<b>Comportamiento</b>	Actuando como los humanos: Test de Turing, definición operativa de Inteligencia	Actuando racionalmente: El agente racional

1.

¿Qué es la Inteligencia Artificial?

# La IA es Multidisciplinar

## Filosofía

- ¿de dónde viene el conocimiento?
- ¿cómo surge la mente a partir de un cerebro físico?
- ¿se pueden usar reglas formales para llegar a conclusiones válidas?

## Matemáticas

- ¿cuáles son las reglas formales para sacar conclusiones válidas? LOGICA
- ¿qué se puede calcular? (COMPUTACIÓN)
- ¿cómo razonamos con información incierta? (ESTADÍSTICA, PROBABILIDAD)

## Teoría del control (cibernética)

- ¿cómo puede un artefacto operar bajo su propio control?

## Economía

- ¿cómo debemos tomar decisiones para maximizar la rentabilidad?
  - ¿cómo tomar decisiones cuando la recompensa tarda mucho en llegar?
- Teoría de juegos

# La IA es Multidisciplinar

## Lingüística

- ¿cómo se relaciona el lenguaje con el pensamiento?

## Ingeniería de computadores

- ¿cómo construir ordenadores más eficientes?
- ¿cómo construir software más eficiente?

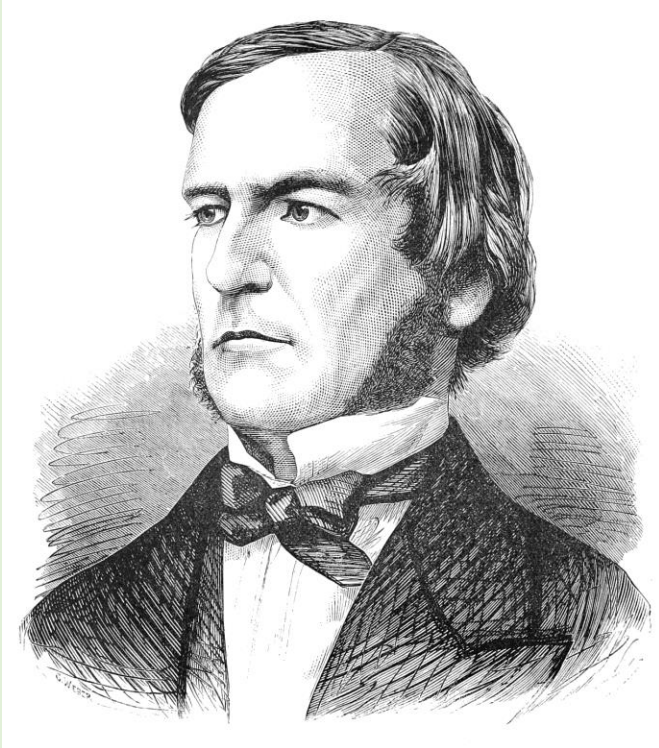
## Neurociencia

- ¿cómo procesa el cerebro la información?

## Psicología

- ¿cómo actúan y piensan los animales y las personas?

# Álgebra de boole y lógica binaria (1847)



George Boole  
1815 - 1864

En los cálculos solo se usan dos símbolos: 1 y 0, por ejemplo

Combinando esos dos símbolos podemos representar cualquier otro símbolo, basta con establecer un código de correspondencia

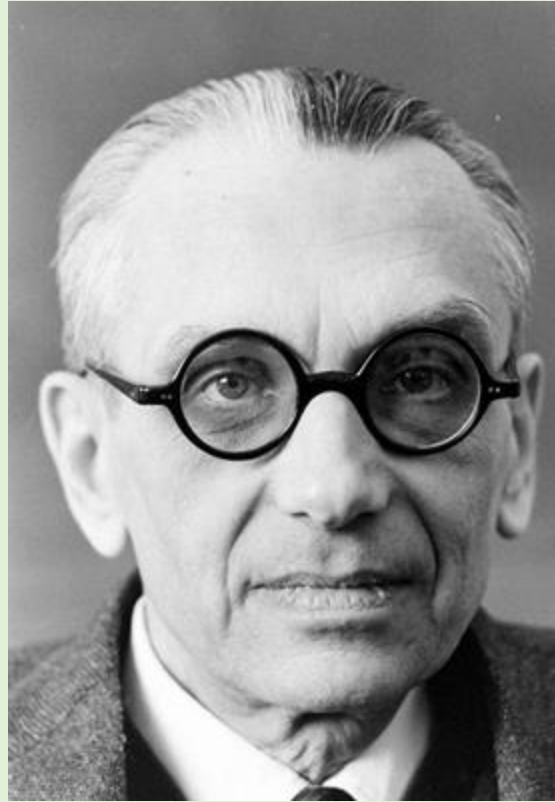
Las funciones lógicas transforman entradas especificadas como combinaciones de 1's y 0's en otras combinaciones de 1's y 0's

0	<b>AND</b>	0	=	0	0	<b>OR</b>	0	=	<b>NOT</b>	0	=	1
0	<b>AND</b>	1	=	0					<b>NOT</b>	1	=	0
1	<b>AND</b>	0	=	0	0	<b>OR</b>	1	=				
1	<b>AND</b>	1	=	1	1							
					1	<b>OR</b>	0	=				
					1							
					1	<b>OR</b>	1	=				
					1							

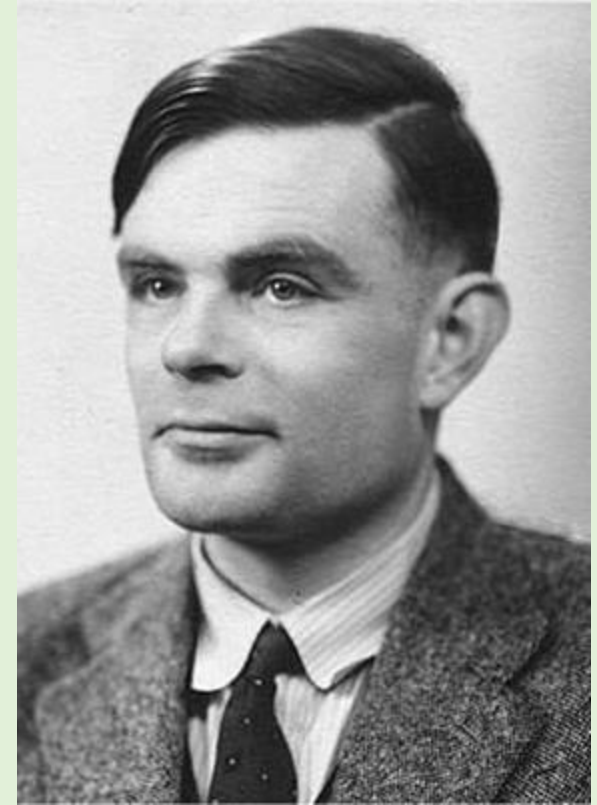
# Teoría de la computación (1930-40)



Alonzo Church  
1903 - 1995



Kurt Gödel  
1906 - 1978



Alan Turing  
1912 - 1954

# Máquina de Turing (1936)

230

A. M. TURING

[Nov. 12,

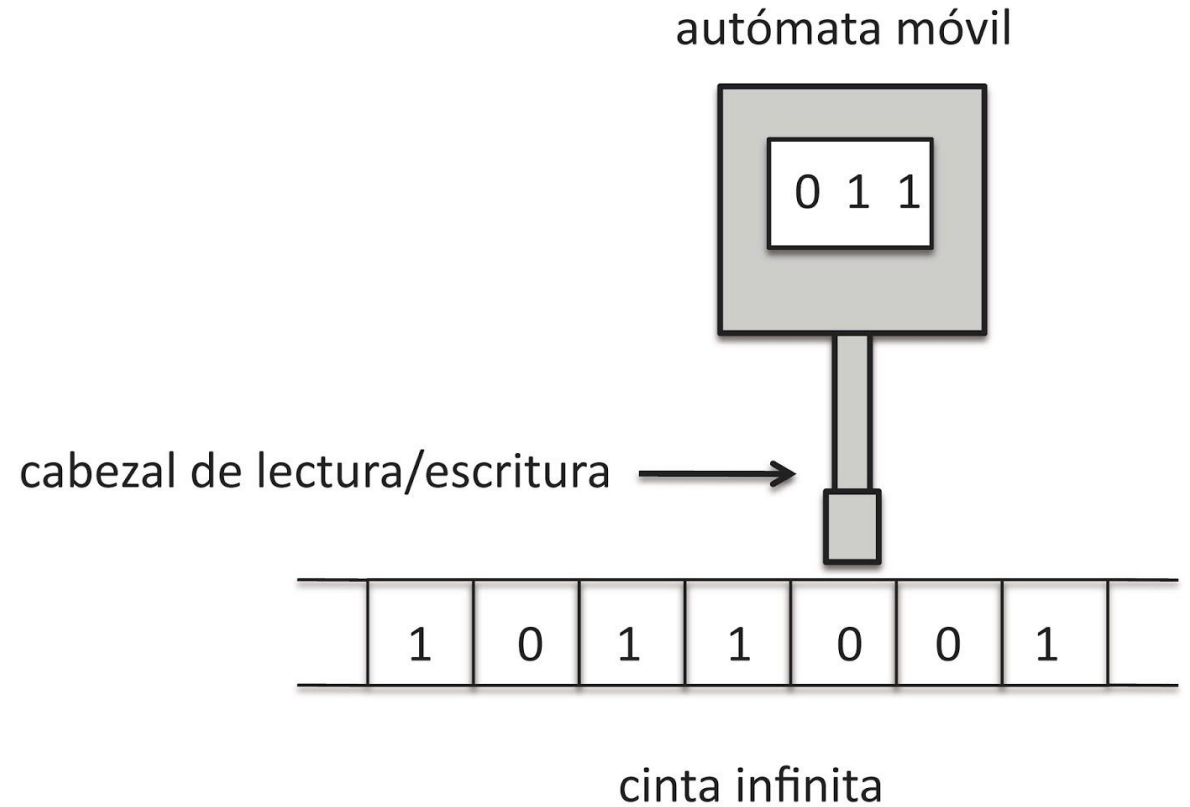
## ON COMPUTABLE NUMBERS, WITH AN APPLICATION TO THE ENTSCHEIDUNGSPROBLEM

By A. M. TURING.

[Received 28 May, 1936.—Read 12 November, 1936.]

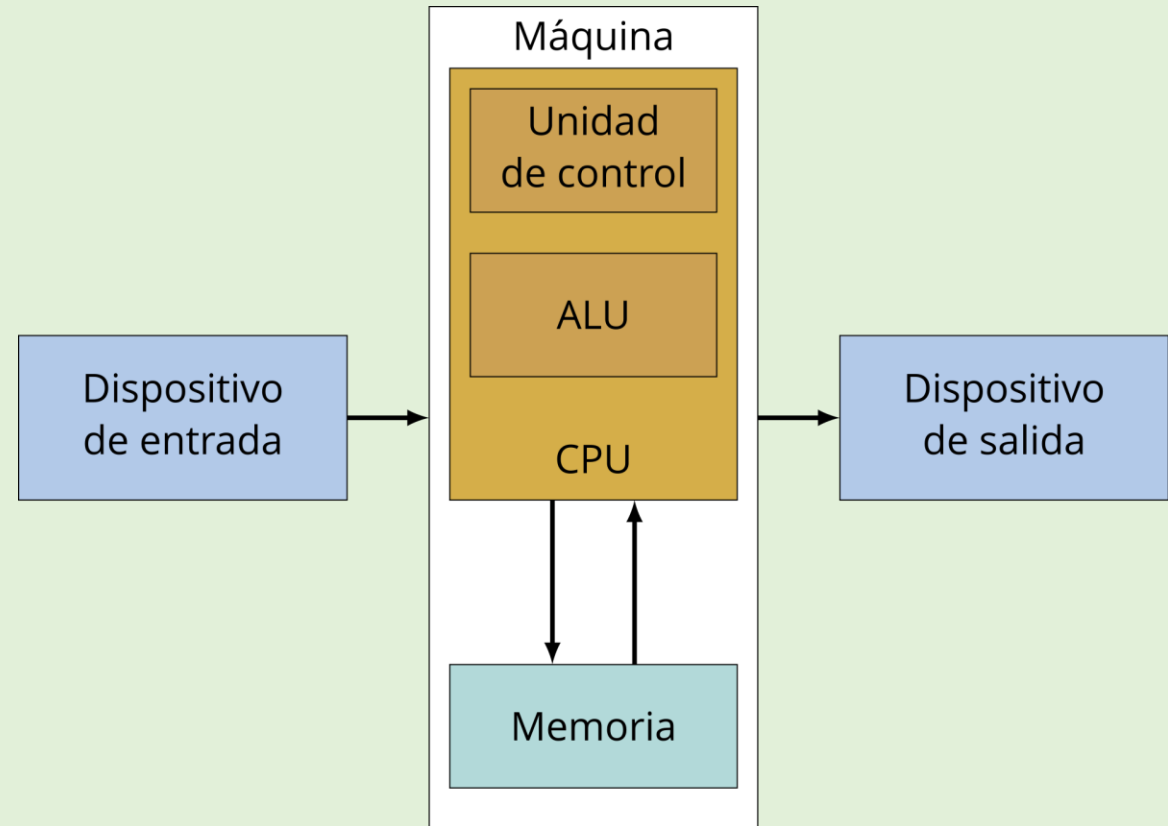
The “computable” numbers may be described briefly as the real numbers whose expressions as a decimal are calculable by finite means. Although the subject of this paper is ostensibly the computable *numbers*, it is almost equally easy to define and investigate computable functions of an integral variable or a real or computable variable, computable predicates, and so forth. The fundamental problems involved are, however, the same in each case, and I have chosen the computable numbers for explicit treatment as involving the least cumbersome technique. I hope shortly to give an account of the relations of the computable numbers, functions, and so forth to one another. This will include a development of the theory of functions of a real variable expressed in terms of computable numbers. According to my definition, a number is computable if its decimal can be written down by a machine.

In §§ 9, 10 I give some arguments with the intention of showing that the computable numbers include all numbers which could naturally be





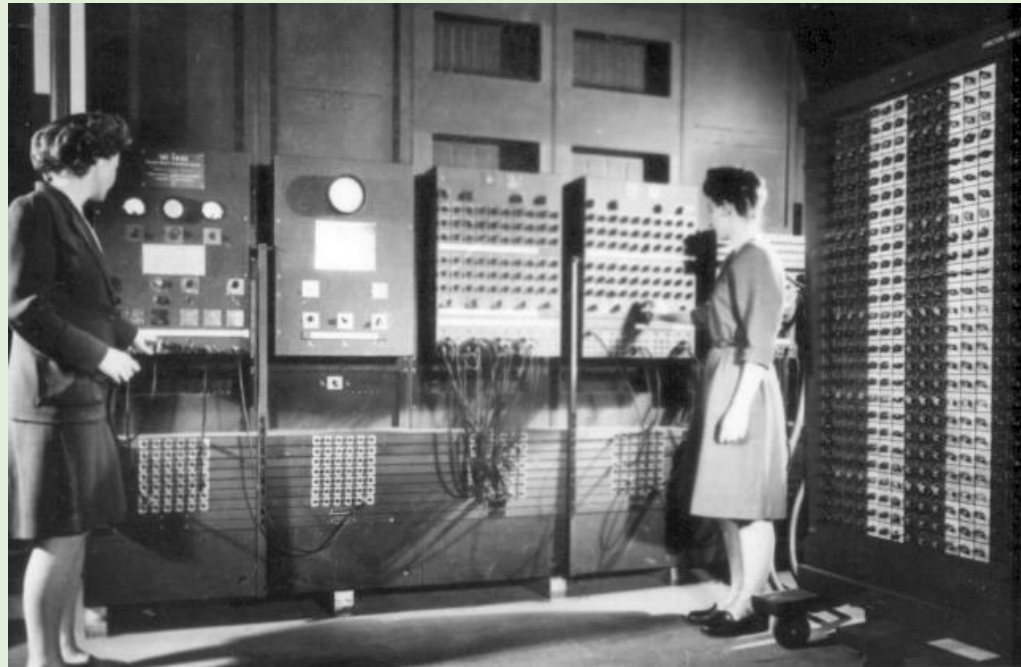
# Arquitectura de Von Newman. El Programa almacenado (1945)



# Válvulas de vacío y primeras computadoras



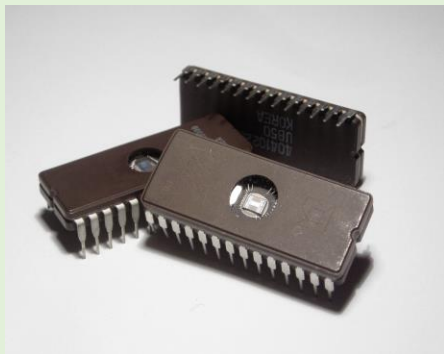
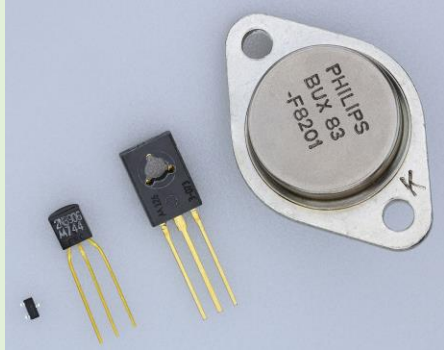
válvulas de vacío



ENIAC (1945, válvulas de vacío)

John Presper Eckert, John William Mauchly, Betty Snyder Holberton, Jean Jennings Bartik, Kathleen McNulty Mauchly Antonelli, Marlyn Wescoff Meltzer, Ruth Lichterman Teitelbaum y Frances Bilas Spence

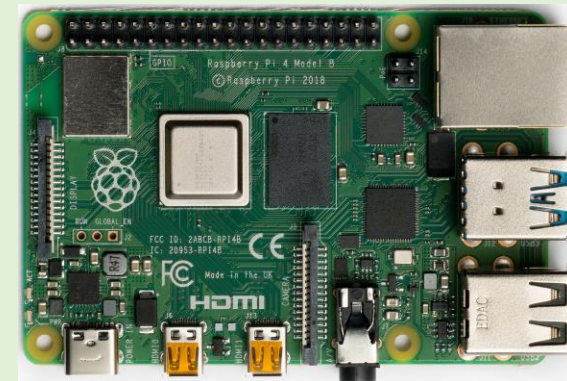
# El transistor y los chips



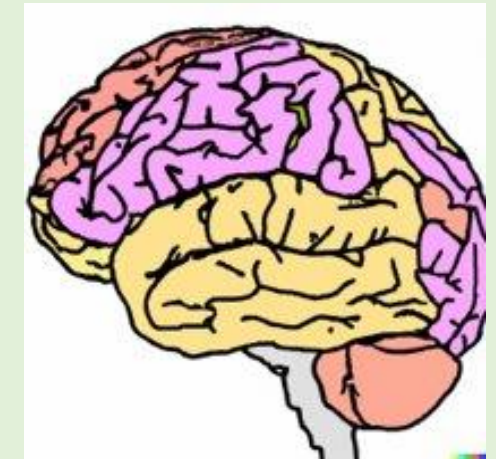
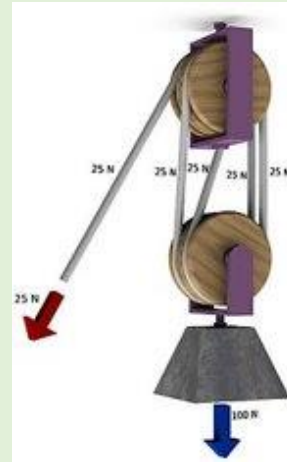
IBM 1620 (1959)



Raspberry Pi (2012)

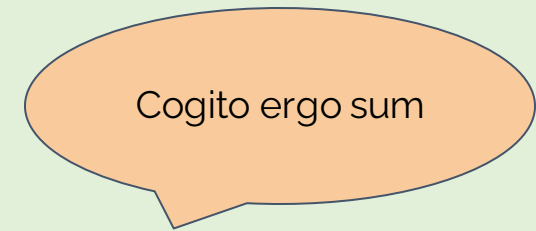
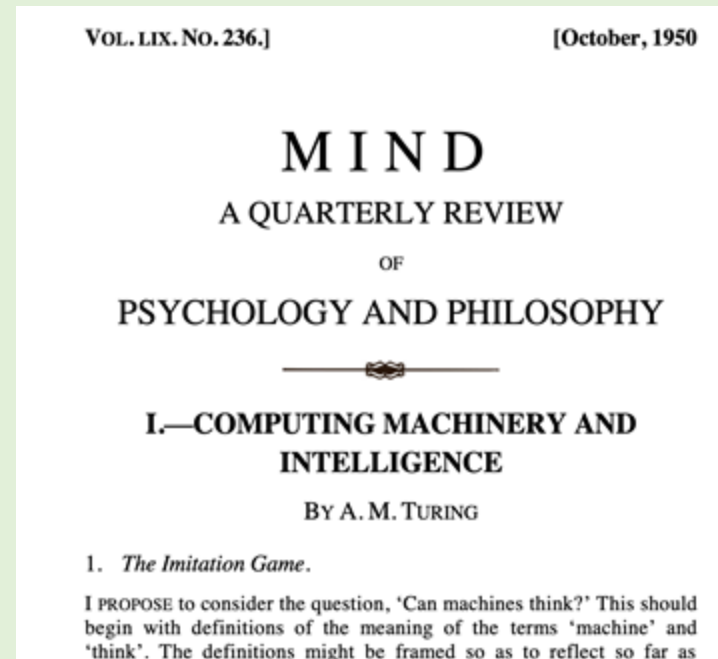
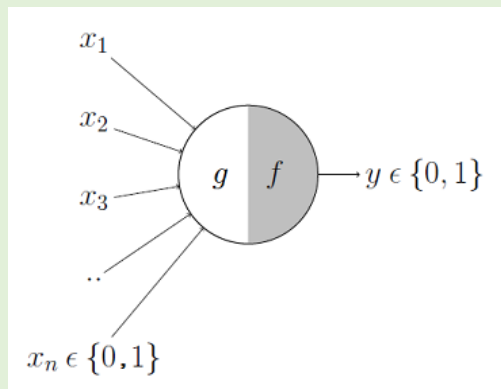
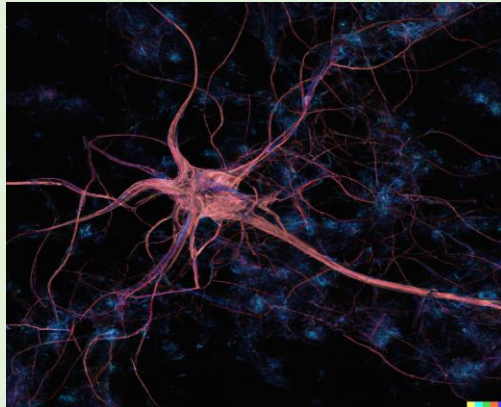


# Ha nacido un nuevo tipo de máquina: la computadora



## Gestación (1943 – 55)

- Neurona artificial, redes neuronales artificiales. W. McCulloch, W. Pitts (1943)
- Computing Machinery and Intelligence. Alan Turing (1950)



## Nacimiento (1956)

- Dartmouth workshop (1956)
- The Logic Theorist (Newell, Simon)



*¿Es posible describir cada aspecto del aprendizaje y otras características de la inteligencia de manera que se pueda construir una máquina que la simule?*

- J. McCarthy
- M.L. Minsky
- N. Rochester
- C.E. Shannon

## Entusiasmo temprano, grandes expectativas (1952 -1969)

- Época de muchos logros computacionales (mira mamá, sin manos!)
- Desarrollo de los lenguajes simbólicos (Lisp, Prolog)
- General Problem Solver (Newell, Simon)
- Geometry Theorem Prover
- Ajedrez y Damas
- Hipótesis physical symbol system -> es suficiente un sistema de representación simbólica para emular cualquier característica de la inteligencia
- Perceptron, Adaline

## Una dosis de realidad (1966-73)

- Incapacidad de resolver problemas más complejos y realistas. Explosión combinatoria (intratabilidad).
- La gramática + diccionario no es suficiente para traducir: the spirit is willing but the flesh is weak -> el vodka es bueno pero la carne está podrida.
- Limitación del perceptron (Minsky y Paper)



## Sistemas expertos: la clave para alcanzar la inteligencia (1969-79)

- Sistemas deductivos con cientos o miles de reglas extraídas de la experiencia y los conocimientos de expertos en distintos dominios (medicina, química, ingeniería de procesos, ...)
- Se desarrollan sistemas expertos capaces de "entender" el lenguaje natural, pero se encuentran de nuevo con el problema del contexto. "La sintaxis no existe".
- Limitación del perceptron (Minsky y Paper)

## La Inteligencia artificial se convierte en una industria (1980 – presente)

- Los sistemas expertos se comercializan.
- En Japón (1981) se anuncia el proyecto: "Fifth Generation"
- En EEUU se crea la MCC (Microelectronics and Computers Technology Corporation)
- A pesar de activarse el entusiasmo por la IA, los logros no alcanzaron lo deseado.

## El retorno de las redes neuronales (1986 – presente)

- Invención del algoritmo *backpropagation*.
- El enfoque conexionista (bottom-up) comienza a superar al enfoque simbólico (top-down)
- La hipótesis physical symbol system comienza a tambalearse.
- A pesar de activarse el entusiasmo por la IA, los logros no alcanzaron lo deseado.

## Disponibilidad de grandes conjuntos de datos (2001- presente)

- Internet + potencia de cómputo + almacenamiento barato -> proliferación de los datos
- El foco se desplaza desde el algoritmo hacia los datos. Más preocupación por los datos que por el algoritmo.
- Un algoritmo mediocre entrenado con 100 millones de datos produce mejores modelos que el mejor de los algoritmos con 1 millón de datos.

## Explosión de la IA basada en deep learning (2017- presente)

- Explosión del Machine Learning, especialmente del Deep Learning, basado en el uso de redes neuronales con muchas capas y distintas arquitecturas.
- Aparición de la arquitectura *transformer* y las redes generativas adversarias.
- Auge de la IA generativa.
- Aparición de los LLM (Large Language Models),
- OpenAI, Google, Deep Mind, y otras muchas empresas comienzan a entrenar modelos con ingentes cantidades de datos.
- **Aparición de las primeras herramientas educativas para enseñar Machine Learning a los niños y niñas.**

# Dos concepciones del término IA

## IA FUERTE (o GENERAL)

La IA fuerte se refiere a sistemas que tienen la capacidad de realizar cualquier tarea intelectual que un ser humano pueda hacer. Tienen la capacidad de razonar, aprender o adaptarse a nuevas situaciones de la misma manera como lo haría un ser humano.



## IA DÉBIL

La IA débil se refiere a sistemas que realizan tareas específicas. Pueden llegar a superar al humano, pero en una tarea concreta. No tienen la capacidad de razonar, aprender o adaptarse a nuevas situaciones de la misma manera como lo haría un ser humano.

# El fin: conseguir la IA General

**Singularidad Tecnológica:** Momento en el que se construya una superinteligencia que supere en todos los aspectos a la inteligencia humana



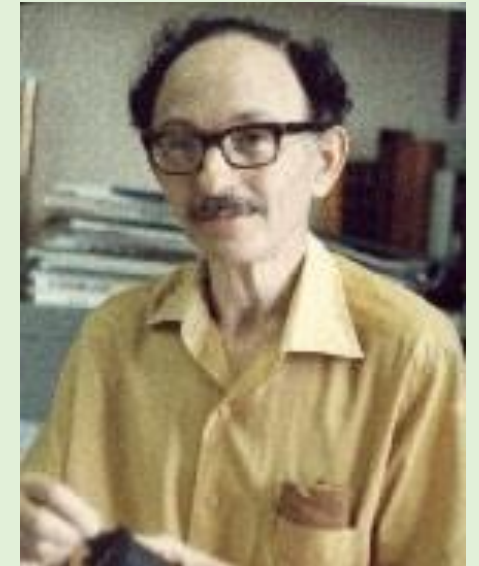
**Explosión de la Inteligencia.** Las IAs pueden construir nuevas y mejores IAs



Podría ser el último invento de la especie humana



¿Qué pasa si no se consigue introducir los **valores humanos** en el desarrollo de estas IAs generales?



I.J. Good

Fuente de la imagen: Wikipedia

# Dos enfoques para desarrollar la IA

## TOP DOWN A partir de reglas

Aplicando las leyes de la física (dinámica) se obtiene la siguiente fórmula para saber la velocidad que alcanza un cuerpo al caer de una altura h:

$$v = \sqrt{2gh}$$

Demostración de teoremas (Logic Theorist, 1956)

Sistemas de navegación: encontrar la ruta óptima

Diseño del layout de circuitos electrónicos y de circuitos integrados

Sistemas expertos

IBM Deep Blue

## BOTTON UP A partir de datos

Se suelta la pelota desde varias alturas, cuantas más mejor, y se mide la velocidad. Con el conjunto de datos obtenido se construye un **modelo** matemático que generalice el resultado para cualquier altura.

$$v = M(h)$$

Reconocimiento de textos

Reconocimiento de imágenes

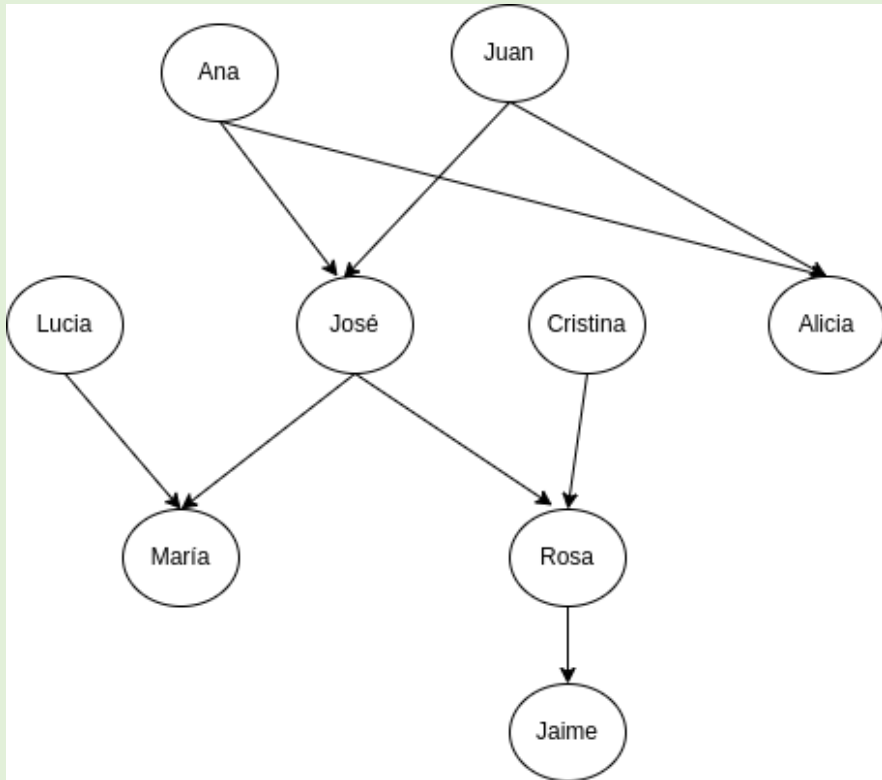
Reconocimiento de poses del cuerpo

Reconocimiento de sonidos

Reconocimiento de conjuntos numéricos



# Ejemplo top-down: Prolog

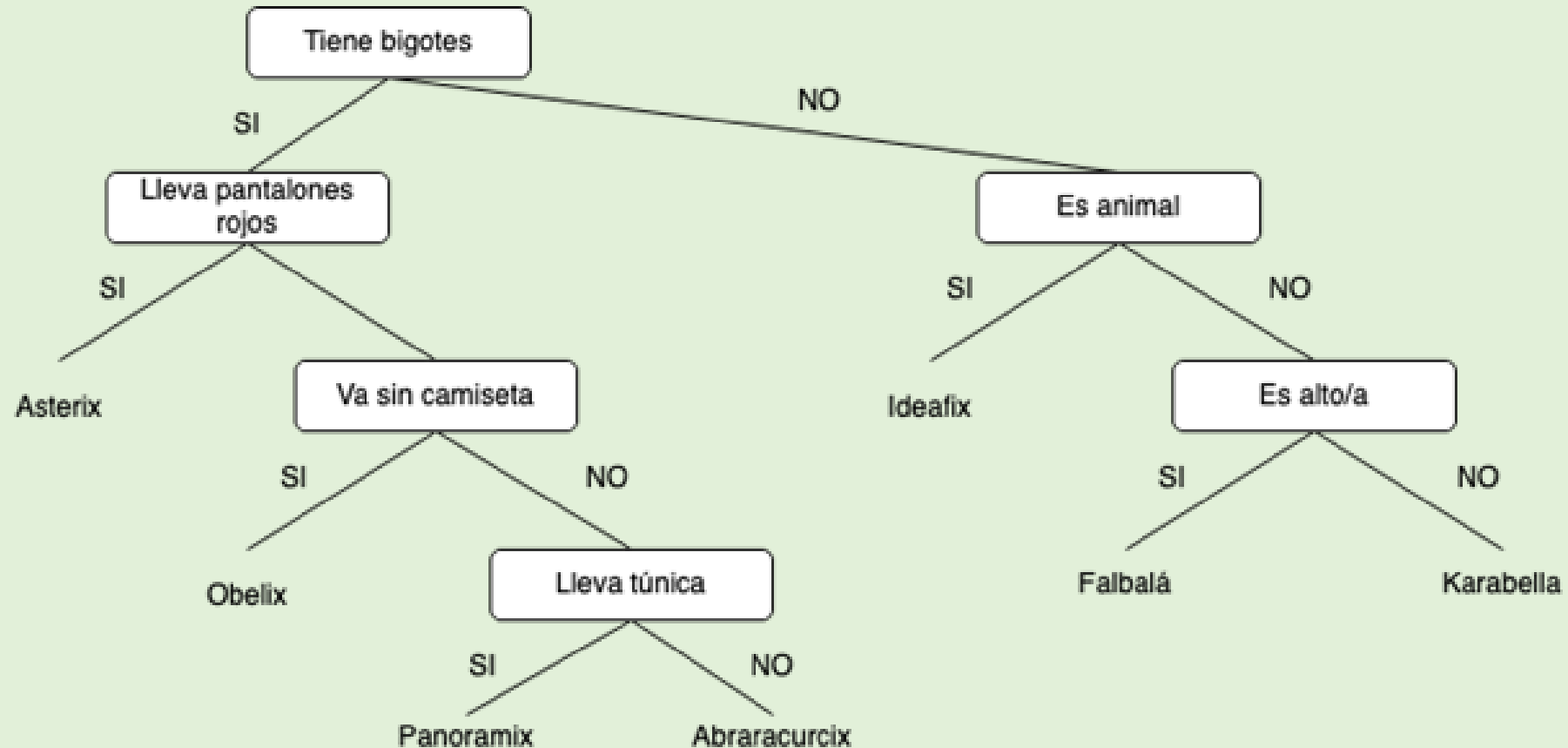


```
1 procreador(ana,jose).
2 procreador(juan,jose).
3 procreador(juan,alicia).
4 procreador(jose,maria).
5 procreador(lucia,maria).
6 procreador(jose,rosa).
7 procreador(cristina,rosa).
8 procreador(rosa,jaime).
9 mujer(ana).
10 mujer(maria).
11 mujer(alicia).
12 mujer(rosa).
13 mujer(cristina).
14 mujer(lucia).
15 mujer(rosa).
16 hombre(jose).
17 hombre(juan).
18 hombre(jaime).
19 madre(X,Y):-procreador(X,Y),mujer(X).
20 padre(X,Y):-procreador(X,Y),hombre(X).
21 hermana(X,Y):-procreador(Z,X),procreador(Z,X),mujer(X),X\==Y.
22 hermano(X,Y):-procreador(Z,X),procreador(Z,X),hombre(X),X\==Y.
23 abuelo(X,Y):-procreador(X,Z),procreador(Z,Y),hombre(X).
24 abuela(X,Y):-procreador(X,Z),procreador(Z,Y),mujer(X).
```



```
hermana(alicia,jose).
true ?
yes
| ?- hermano(jose,alicia).
true ?
```

# Otro ejemplo top-down: Árbol de decisión



<https://scratch.mit.edu/projects/461277594/>

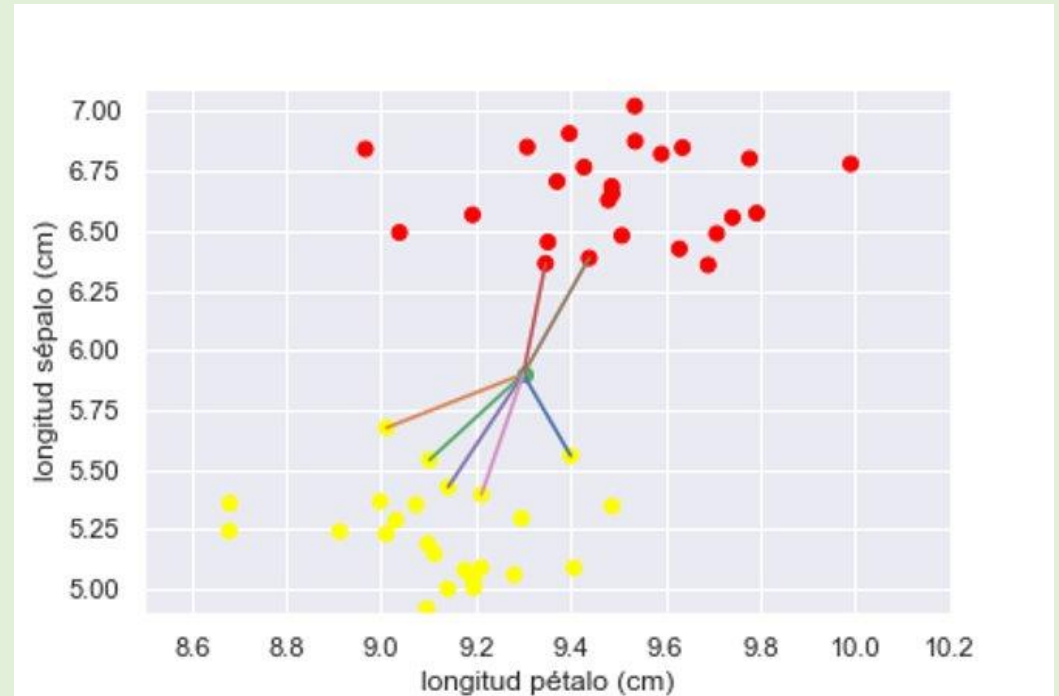
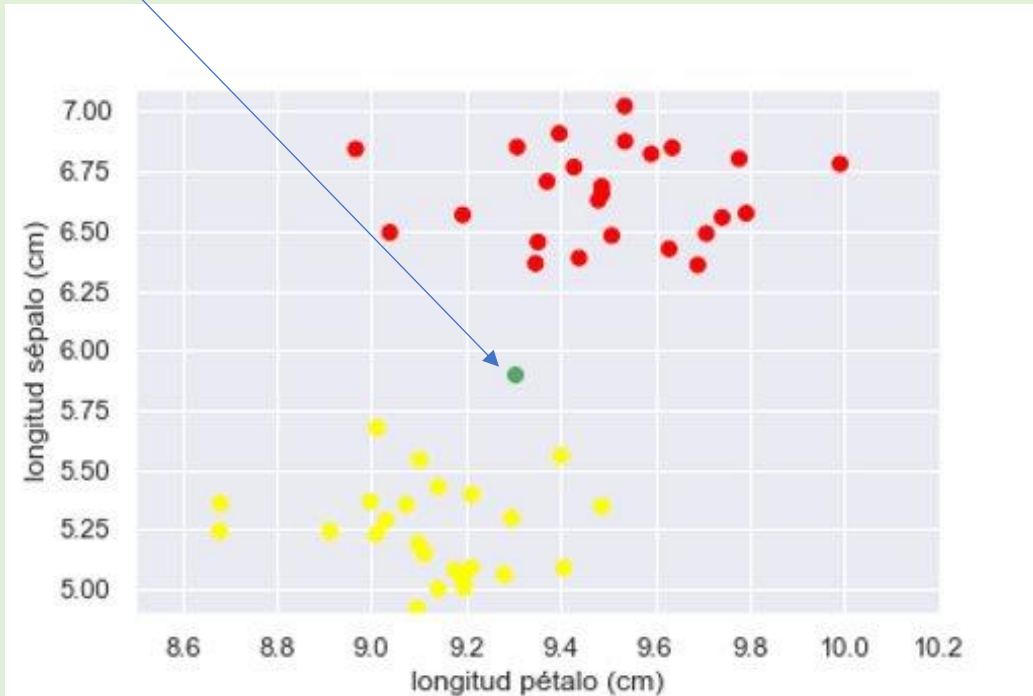
<https://es.akinator.com/>

# Ejemplo botton-up: K - nearest neighbors

Punto a clasificar

Confianza amarillo= 5/7 (70%)

Confianza rojo= 2/7 (29%)

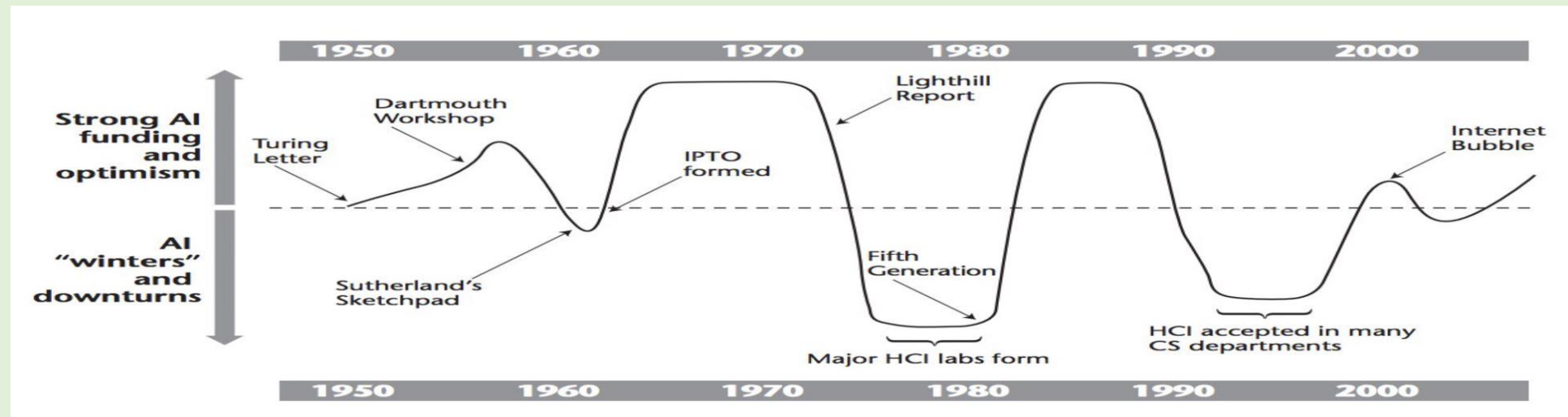


# El efecto IA (o la maldición del término IA)

*“As soon as it works, no one calls it AI anymore”* (John McCarthy)

*“AI is whatever hasn’t been done yet”* (Larry Tesler)

*“Inteligencia Artificial es el término que se usa para todas las cosas guays que un ordenador aún no puede hacer”* (Definición chistosa popular)



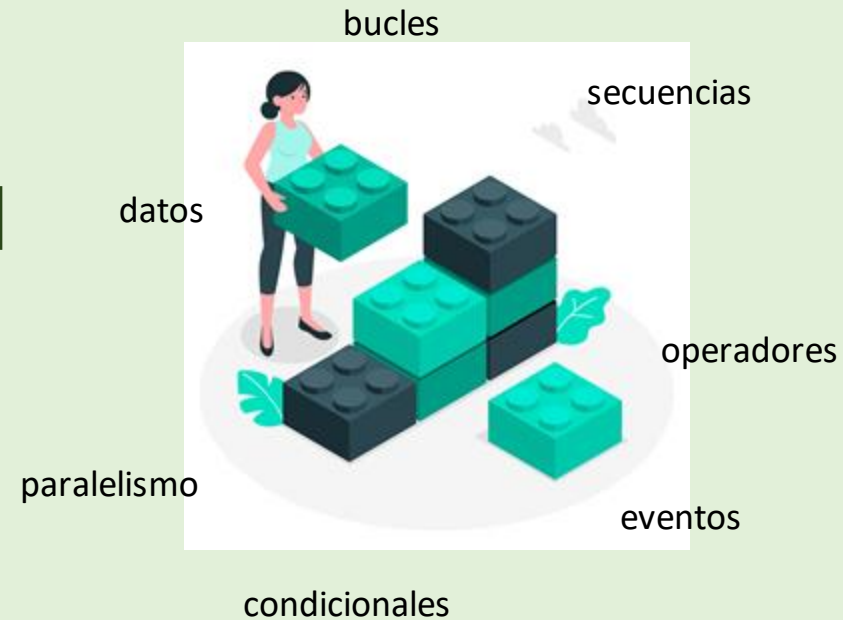
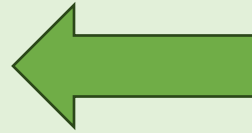
2.

¿Por qué enseña Inteligencia Artificial?

1. Para (sobre) vivir en una sociedad digital
2. Preocupación institucional
3. Para fomentar el Pensamiento Computacional

# Aprender IA para (sobre) vivir en una sociedad digital

Como ciudadanos, debemos conocer las tecnologías digitales lo mejor posible y exigir el desarrollo y uso de herramientas digitales responsables, transparentes y justas.



Conocer los fundamentos de las ciencias de la computación (informática) será de gran ayuda.

# Aprender IA para (sobre) vivir en una sociedad digital

Muchas aplicaciones de uso común usan IA





# Aprender IA para (sobre) vivir en una sociedad digital



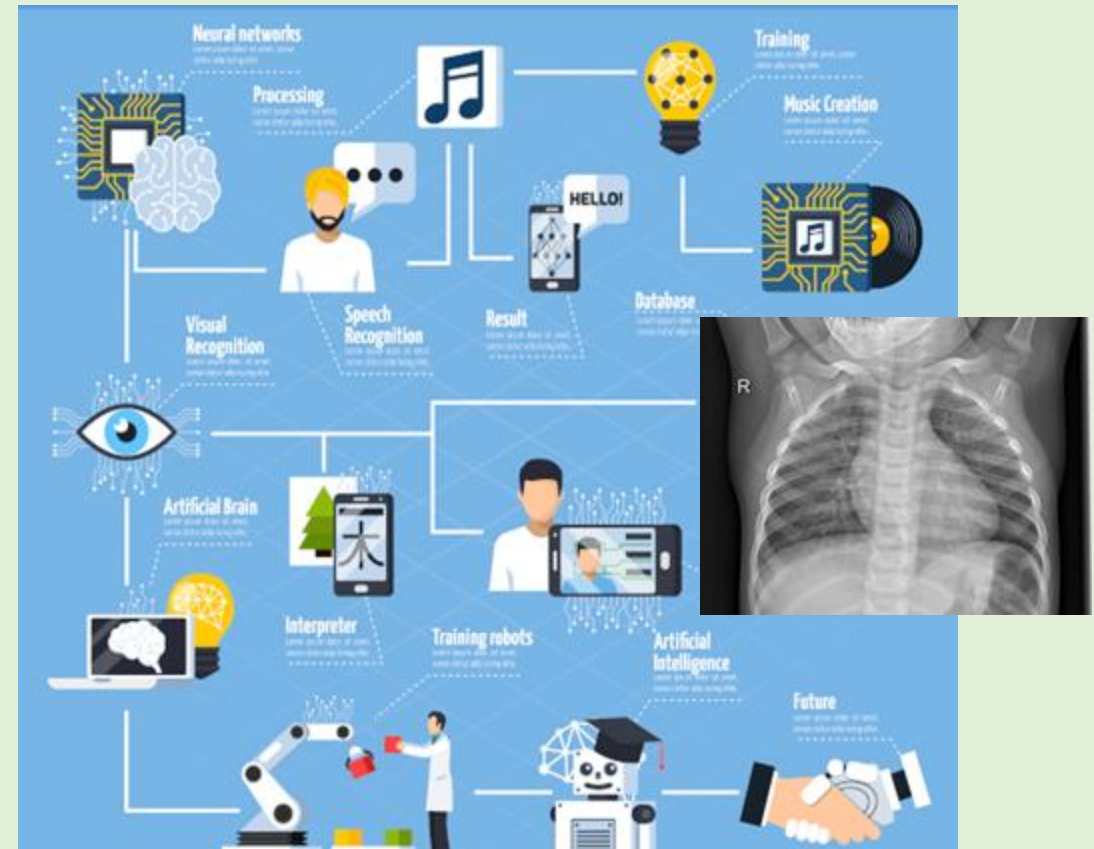
# Aprender IA para (sobre) vivir en una sociedad digital

La IA tiene un fuerte impacto en todos los aspectos de la sociedad, y por lo que se está viendo, aún tendrá más en un futuro.

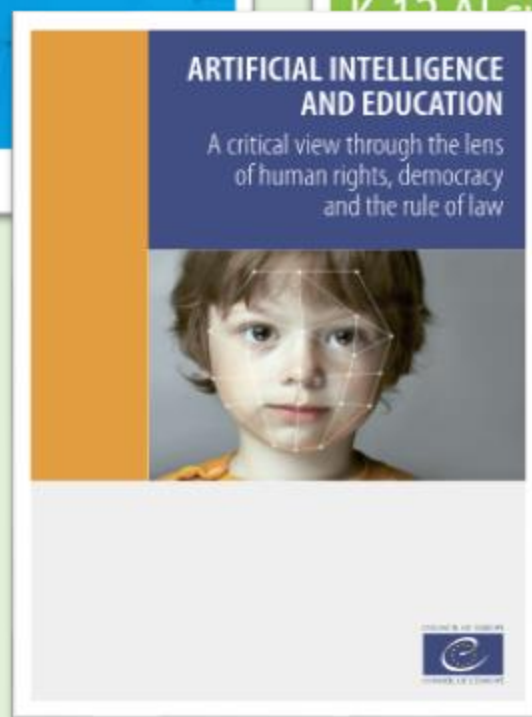
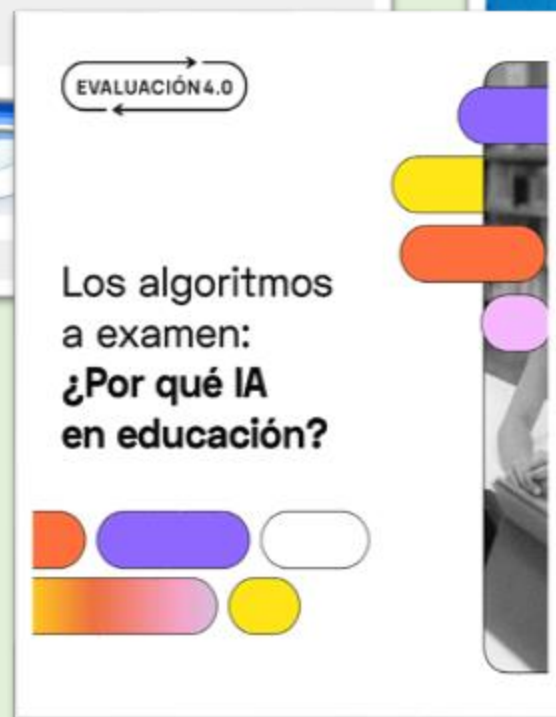
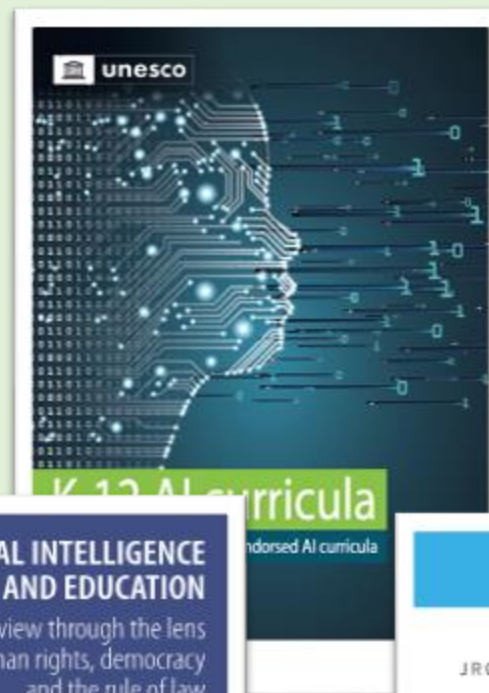
Justicia



Aplicaciones médicas



# Aprender IA, una preocupación institucional



# Aprender IA, una preocupación institucional



## PENSAMIENTO COMPUTACIONAL EN LA LOMLOE ENSEÑANZAS MÍNIMAS

### Al completar este nivel educativo el alumno o la alumna...

### Áreas o materias

#### INFANTIL

Desarrolla, de manera progresiva, las destrezas del pensamiento computacional a través de procesos de observación y manipulación de objetos para iniciarse en la interpretación del entorno y responder de forma creativa a las situaciones y retos que se plantean.

Área 2. Descubrimiento y exploración del entorno (en el 2º ciclo)

#### PRIMARIA

Se inicia en el desarrollo de soluciones digitales sencillas y sostenibles (reutilización de materiales tecnológicos, programación informática por bloques, robótica educativa...) para resolver problemas concretos o retos propuestos de manera creativa, solicitando ayuda en caso necesario.

Conocimiento del medio natural, social y cultural (en los 3 ciclos) y Matemáticas (en los 3 ciclos)

#### E.S.O.

Desarrolla aplicaciones informáticas sencillas y soluciones tecnológicas creativas y sostenibles para resolver problemas concretos o responder a retos propuestos, mostrando interés y curiosidad por la evolución de las tecnologías digitales y por su desarrollo sostenible y uso ético.

Tecnología y Digitalización (1º, 2º y 3º), Tecnología (4º), Digitalización (4º), Biología y Geología (1º, 2º, 3º y 4º), Matemáticas (1º, 2º y 3º), Matemáticas A y B (4º), Ámbito Ciencias Aplicadas (CFGB)

#### BACHI-LLERATO

Desarrolla soluciones tecnológicas innovadoras y sostenibles para dar respuesta a necesidades concretas, mostrando interés y curiosidad por la evolución de las tecnologías digitales y por su desarrollo sostenible y uso ético

Tecnología e Ingeniería I y II, Biología, Geología y Ciencias Ambientales (1º), Geología y Ciencias Ambientales (2º), Matemáticas I y II, Matemáticas Aplicadas a las Ciencias Sociales I y II, Matemáticas Generales

Para saber más: <https://programamos.es/pensamiento-computacional-en-todas-las-etapas-no-universitarias-analisis-de-la-lomloe/>

# Aprender IA para desarrollar el Pensamiento Computacional

Competencia instrumental

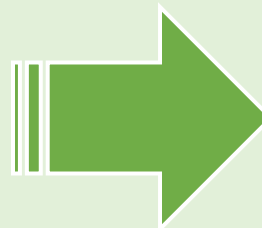
Habilidad cognitiva



Programación



Actividades desenchufadas



## Computational Thinking



IA y Ciencia de Datos

3.

¿Qué y cómo enseñar IA en la escuela?

# Qué enseñar sobre IA en la escuela

## Currículo competencial de Long y Magerko

1. Recognizing AI	Distinguish between technological artifacts that use and do not use AI.
2. Understanding intelligence	Critically analyse and discuss features that make an entity 'intelligent'. Discuss differences between human, animal, and machine intelligence.
3. Interdisciplinarity	Recognize that there are many ways to think about and develop 'intelligent' machines. Identify a variety of technologies that use AI, including technology spanning cognitive systems, robotics and ML.
4. General vs narrow AI	Distinguish between general and narrow AI.
5. AI strengths and weaknesses	Identify problem types that AI does/does not excel at. Determine when it is appropriate to use AI and when to leverage human skills.
6. Imagine future AI	Imagine possible future applications of AI and consider the effects of such applications on the world.
7. Representations	Understand what a knowledge representation is and describe some examples of knowledge representations.
8. Decision-making	Recognize and describe examples of how computers reason and make decisions.
9. ML steps	Understand the steps involved in machine learning and the practices and challenges that each step entails.

10. Human role in AI	Recognize that humans play an important role in programming, choosing models, and fine-tuning AI systems.
11. Data literacy	Understand basic data literacy concepts.
12. Learning from data	Recognize that computers often learn from data (including one's own data).
13. Critically interpreting data	Understand that data requires interpretation. Describe how the training examples provided in an initial dataset can affect the results of an algorithm.
14. Action and reaction	Understand that some AI systems have the ability to physically act on the world. This action can be directed by higher-level reasoning (e.g. walking along a planned path) or reactive impulses (e.g. jumping backwards to avoid a sensed obstacle).
15. Sensors	Understand what sensors are and that computers perceive the world using sensors. Identify sensors on a variety of devices. Recognize that different sensors support different types of representation and reasoning about the world.
16. Ethics	Identify and describe different perspectives on the key ethical issues surrounding AI: privacy, employment, misinformation, 'singularity', <sup>11</sup> decision-making, diversity, bias, transparency and accountability.
17. Programmability	Understand that agents are programmable.

# Qué enseñar sobre IA en la escuela

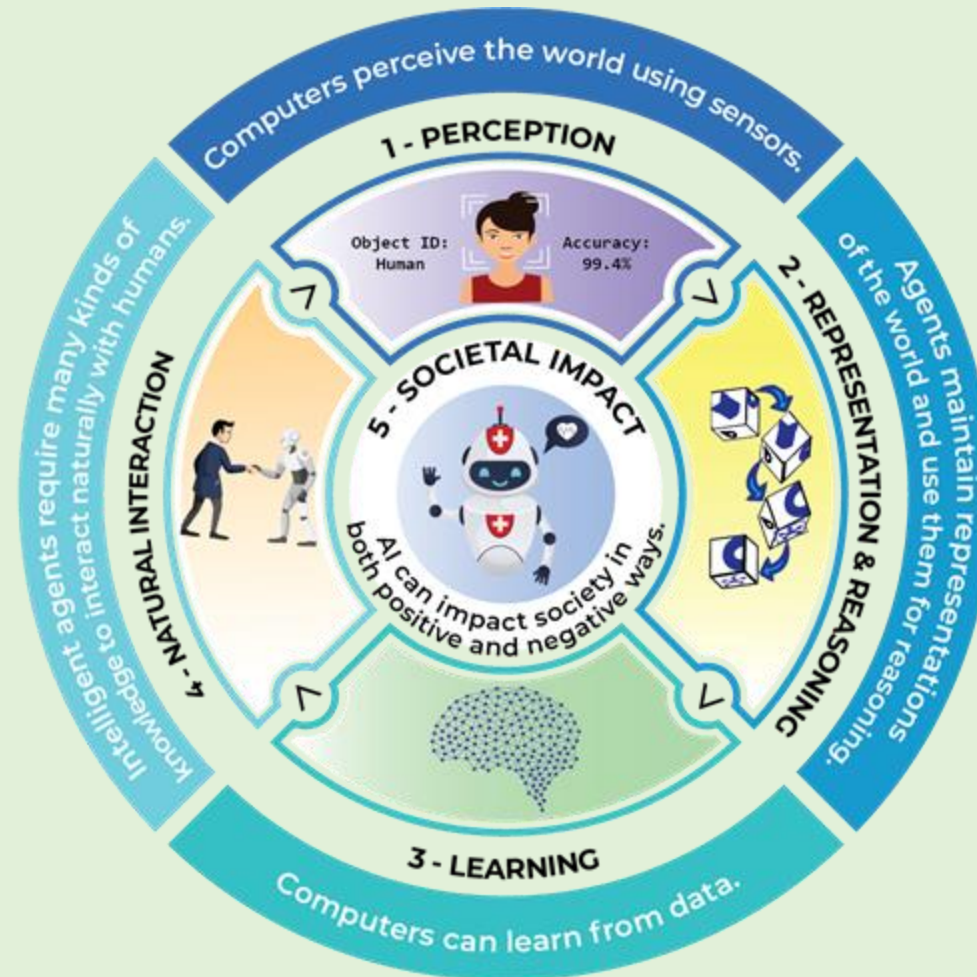
## Framework de Machine Learning de Lao

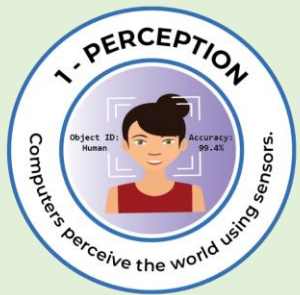
Knowledge	
1. General ML knowledge*	Know what machine learning is (and is not). Understand the entire pipeline of the creation of ML systems.
2. Knowledge of ML methods	Identify when to use a range of ML methods across the breadth of the field (e.g. k-nearest neighbours, CARTs or decision trees, neural networks, ensemble methods). Understand how different methods work.
3. Bias in ML systems*	Understand that systems can be biased, and the different levels and ways in which bias can be introduced.
4. Societal implications of AI*	Understand that ML systems can have widespread positive and negative impacts. Consider the ethical, cultural and social implications of what they do.
Skills	
1. ML problem scoping	Determine which problems can and should be solved by ML.
2. ML project planning	Plan a solution which is sensitive to both technical and contextual considerations.
3. Creating ML artefacts	Use tools to create appropriate artefacts.
4. Analysis of ML design interactions and results*	Describe the explicit and implicit design intentions of an ML system. Critically analyse the intentions against how the system can and should be used.
5. ML advocacy*	Critically discuss ML policies, products and education.
6. Independent out-of-class learning	Students seek learning experiences outside the classroom.
Attitudes	
1. Interest	Students are engaged and motivated to study the topic.
2. Identity and community	Students contribute to and learn from a community of peers and/or broader online communities who are interested in ML.
3. Self-efficacy	Students are empowered to build new, meaningful things.
4. Persistence	Students continue and expand their engagement with ML.



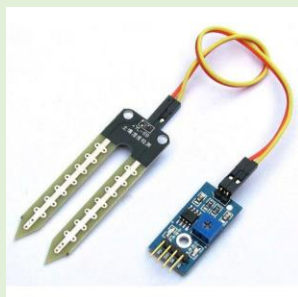
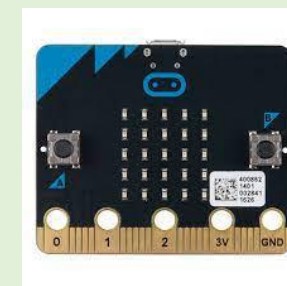
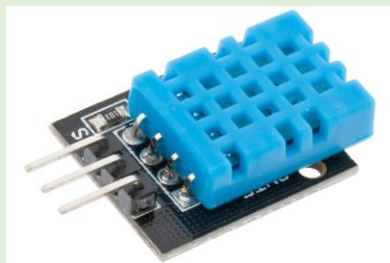
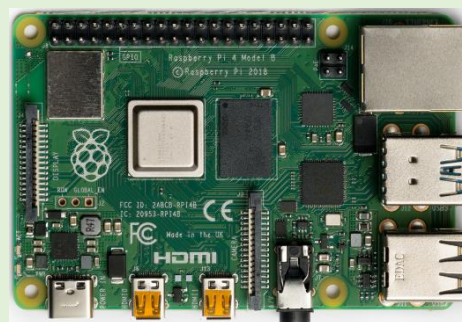
# Qué enseñar sobre IA en la escuela

5 grandes ideas sobre IA



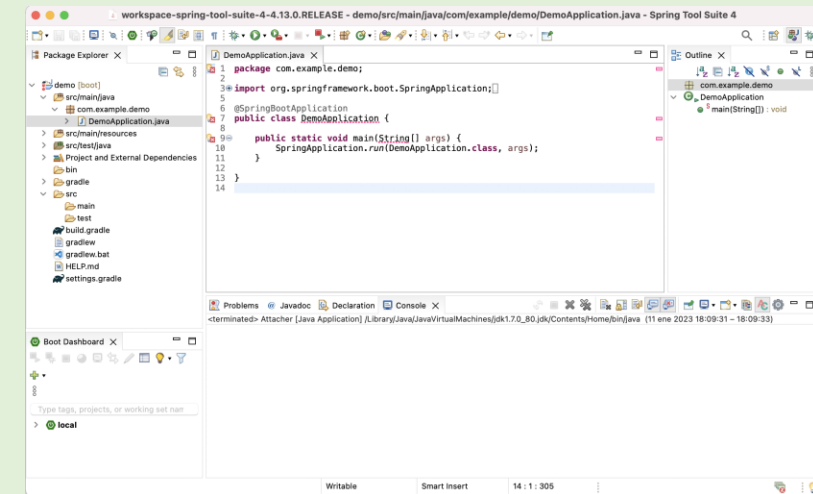
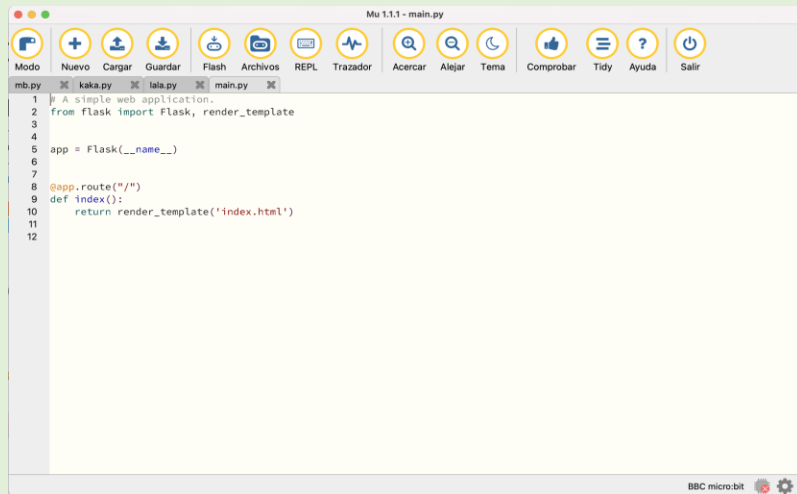
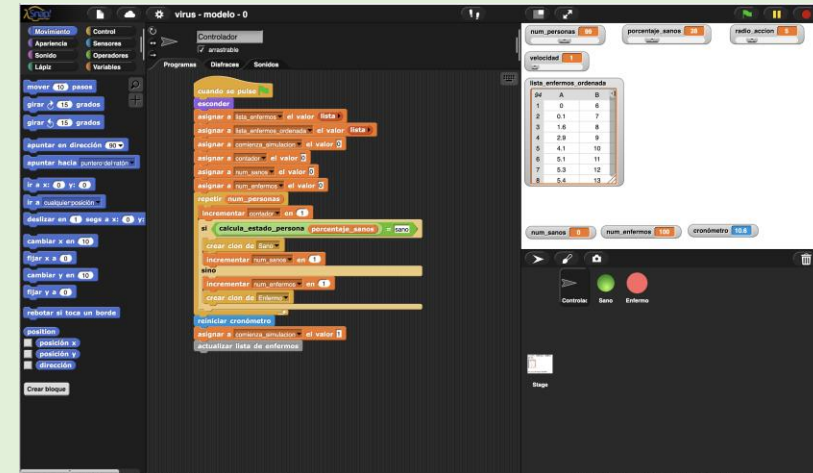
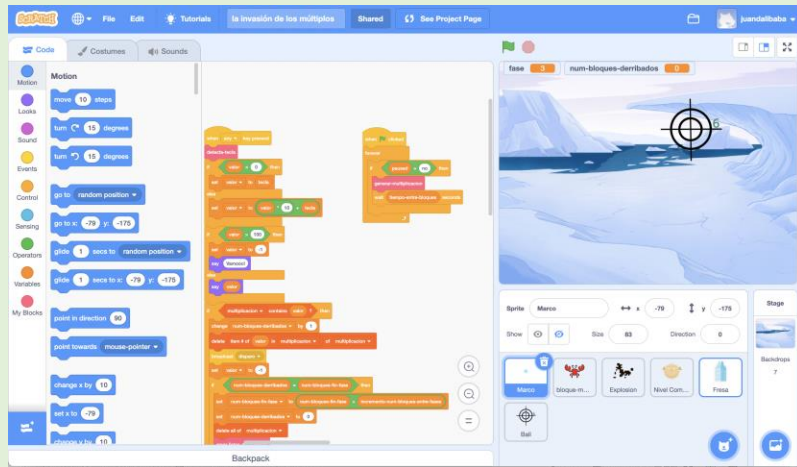


# Cómo enseñar sobre IA en la escuela





# Cómo enseñar sobre IA en la escuela





# Cómo enseñar sobre IA en la escuela

## Machine Learning for Kids

Acerca de Ejercicios Pretrained Stories Book Novedades Ayuda Ingresar Language

### Enséñale a una computadora a jugar un juego

[Empezar](#) [Aprender más](#)

- 1 Agrega ejemplos de cosas que quieres que la computadora reconozca/identifique automáticamente
- 2 Usa los ejemplos para entrenar una computadora y así aprende a reconocer este tipo de objetos (texto/imágenes/números) automáticamente
- 3 Crea un juego en Scratch que use la habilidad de la computadora para reconocerlos

## Cognimates

COGNIMATES Get Started Projects Tutorials Events Research Blog Team

An AI education platform for building games, programming robots & training AI models

[Train Models](#) [Code & Play](#)

## LearningML

LearningML - AI made easy

QUÉ ES POR QUÉ APRENDER DESARROLLO BLOG DESCARGAS ACERCA DE CONTACTO

Elige tu versión. La beta tiene más funcionalidades, pero aún no está suficientemente probada.

[Versión 1.2 \(estable\)](#) [Versión 1.3 \(beta\)](#)

### Recopila datos

Recopila textos o imágenes sobre algo que quieras clasificar de forma automática y añádelos a LearningML indicando a qué clase pertenece cada uno de ellos. Estos datos constituyen el conjunto de entrenamiento.

### Crema un modelo

Construye con LearningML un modelo capaz de clasificar correctamente otros datos distintos, aunque similares, a los del conjunto de entrenamiento.

### Construye una aplicación

Exporta tu modelo de Machine Learning a Scratch y programa una aplicación con capacidad para clasificar datos sobre el tema que hayas elegido. ¡Enhorabuena! ¡has incorporado Inteligencia Artificial a tu programa Scratch!

## Teachable Machine

Acerca de Preguntas frecuentes [Primeros pasos](#)

### Teachable Machine

Prepara a un ordenador para que reconozca tus imágenes, sonidos y posturas.

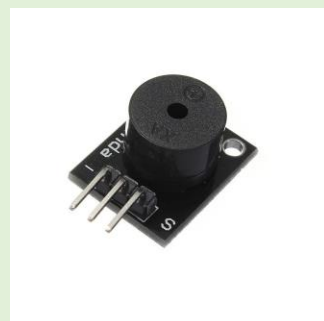
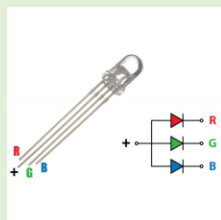
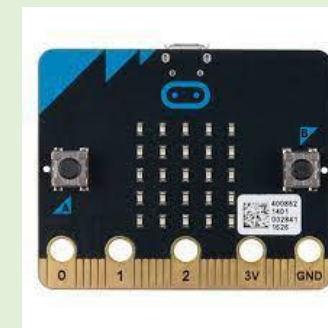
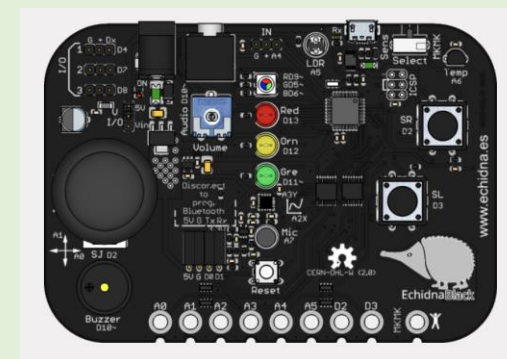
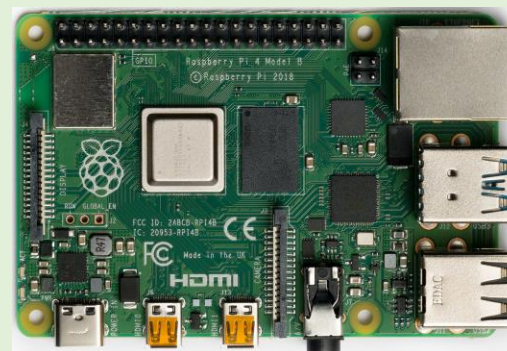
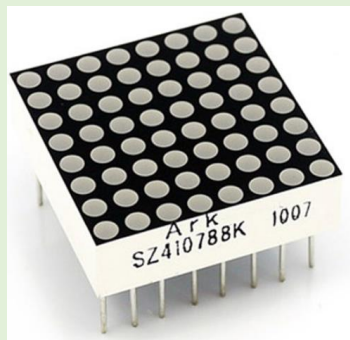
Una forma rápida y sencilla de crear modelos de aprendizaje automático para tus sitios web, aplicaciones y mucho más, sin necesidad de conocimientos especializados ni de programar.

[Primeros pasos](#)

Tree	0%
Wings	100%



# Cómo enseñar sobre IA en la escuela





# Cómo enseñar sobre IA en la escuela

## The moral machine

¿Qué debe hacer el coche autónomo?

1 / 13

Mostrar descripción

Mostrar descripción

Imprint | Privacy Policy

## Dilemas éticos UNESCO

English Français Español العربية 中文

ENHANCED BY Google

Acceso personal Intranet

unesco

"Construir la paz en la mente de los hombres y de las mujeres"

EN BREVE QUÉ HACEMOS DÓNDE TRABAJAMOS SOCIOS ÚNETE RECURSOS

Inicio > Inteligencia Artificial > Recomendación sobre la Ética de la Inteligencia Artificial > Inteligencia artificial: ejemplos de dilemas éticos

### Inteligencia artificial: ejemplos de dilemas éticos

IA sesgada Coche autónomo La IA crea arte AI en el Tribunal de Justicia

### IA sesgada

colegial

colegiala

DETENGA-MOS EL SESO DE GÉNERO EN INTERNET!

4.

Machine Learning: la base de la IA actual

# Machine Learning

Cuando no disponemos de un algoritmo o procedimiento para resolver un problema, pero tenemos una gran cantidad de datos relacionados con sus soluciones, podemos usarlos para extraer patrones con los que **extraer las reglas de manera automática.**

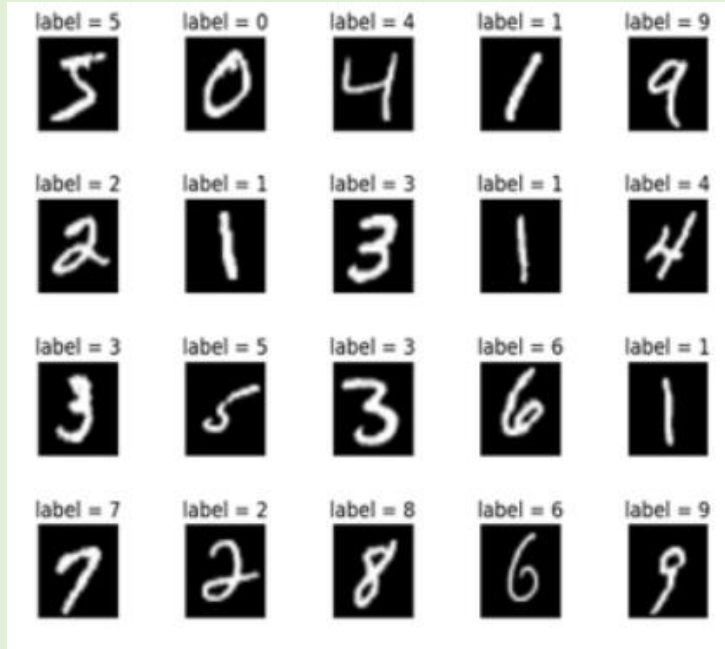
Ya no somos nosotros quienes proponemos las reglas, sino que es la computadora la que ejecutando un **algoritmo** de Machine Learning las **induce** a partir de los datos de ejemplo.

El conjunto de reglas inducidas por el algoritmo de Machine Learning se denomina **modelo** de Machine Learning.



# Los ingredientes del Machine Learning

1



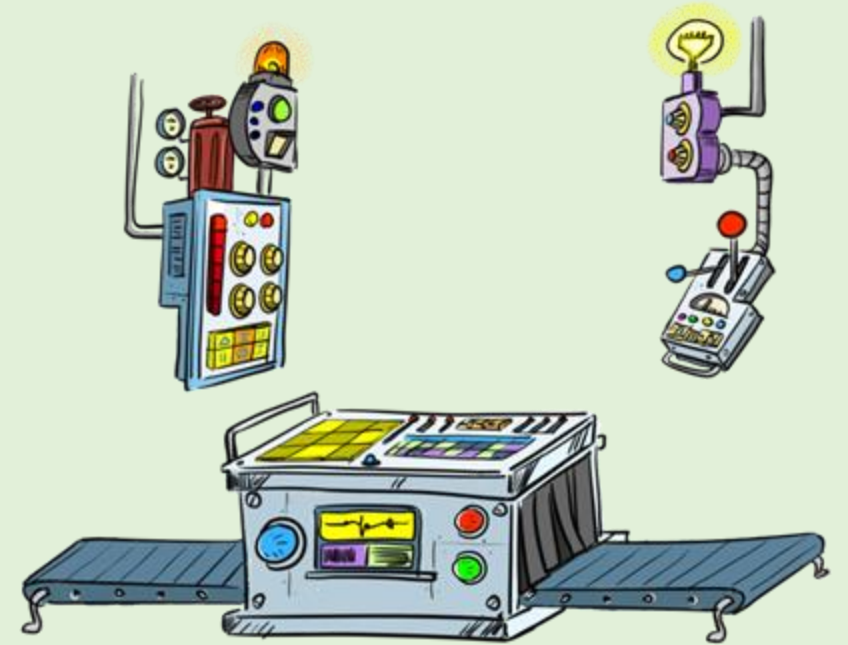
Datos

2



Algoritmo

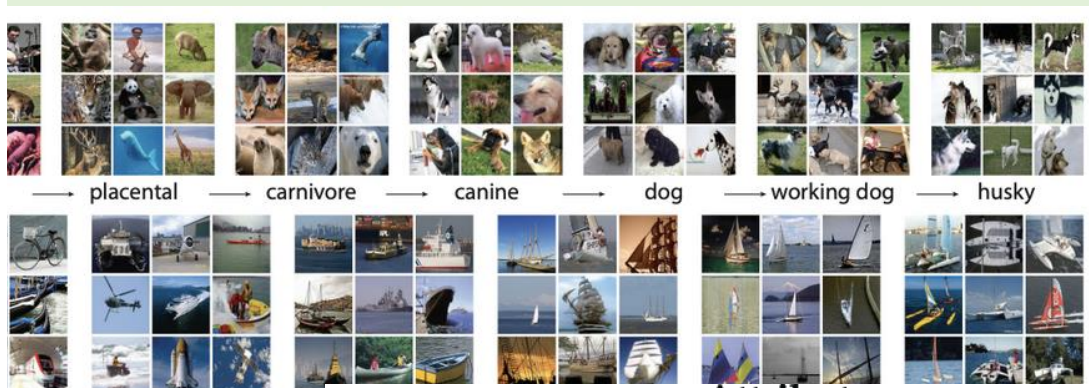
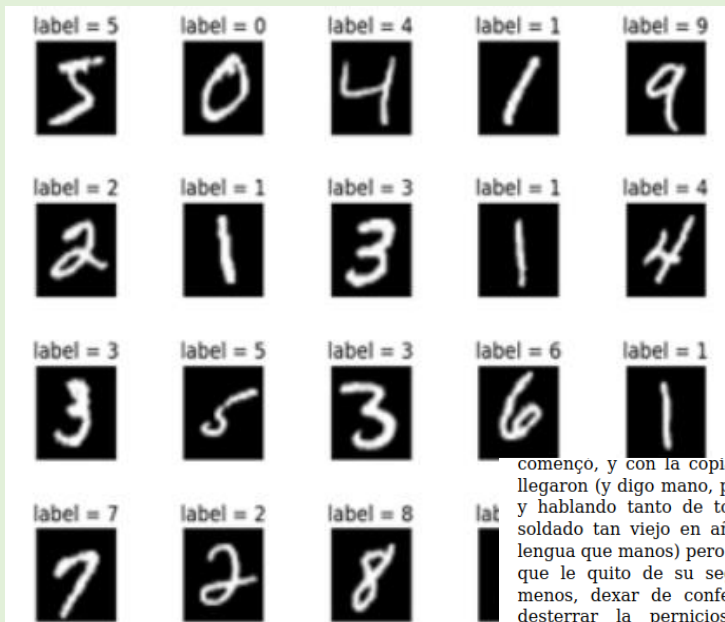
3



Modelo

# 1

## Los Datos



començo, y con la copia de fieles relaciones que a su mano llegaron (y digo mano, pues confiesa de sí que tiene sola una; y hablando tanto de todos, hemos de dezir del que, como soldado tan viejo en años quanto moço en brios, tiene más lengua que manos) pero quexese de mi trabajo por la ganancia que le quito de su segunda parte; pues no podrá, por lo menos, dexar de confesar tenemos ambos un fin, que es desterrar la perniciosa lición de los vanos libros de caballerias, tan ordinaria en gente rustica y ociosa; si bien en los medios diferenciamos; pues él tomó por tales el ofender á mí; y particularmente á quien tan justamente celebran las naciones más extrangeras, y la nuestra debe tanto, por haber entretenido honestissima y fecundamente tantos años los teatros de España con estupendas é innumerables comedias, con el rigor del arte que pide el mundo, y con la seguridad y limpieza que de un ministro del Santo Ofizio se debe esperar.

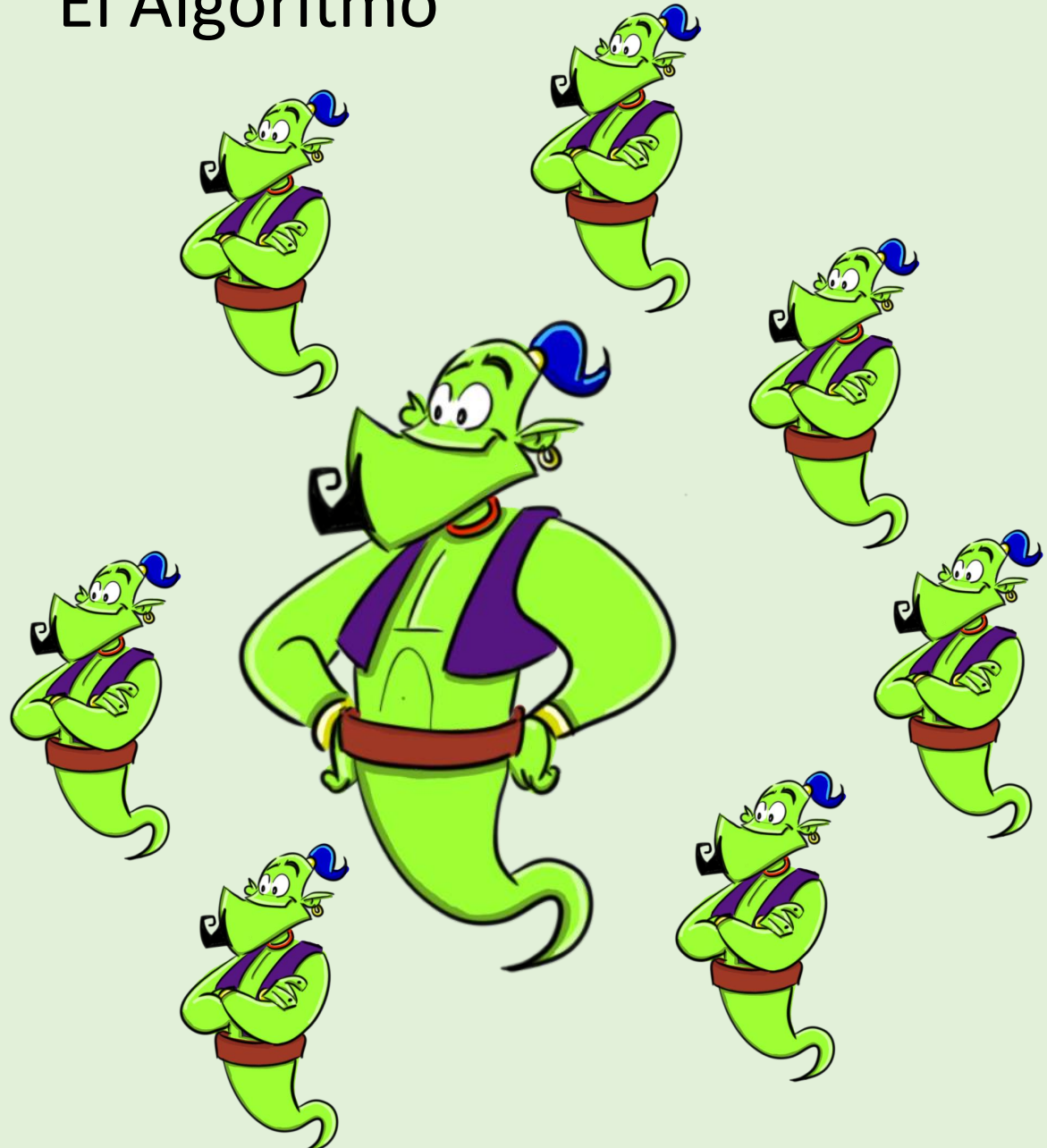
No solo he tomado por medio entremesar la presente comedia con las simplicidades de Sancho Pança, huyendo de ofender á nadie ni de hazer ostentacion de sinonimos voluntarios, si bien supiera hazer lo segundo, y mal lo primero; solo digo que nadie se espante de que salga de diferente autor esta segunda parte, pues no es nuevo el proseguir una historia diferentes sugetos. ¿Cuantos han hablado de los amores de Angelica y de sus sucesos? Las Arcadias, diferentes las han escrito, la Diana no es toda de una mano. Y pues Miguel de Cervantes es ya de viejo como el castillo de San Cervantes, y por los años tan mal contentadizo, que todo y todos le enfadan, y por ello está tan falto de

Case	Attributes				Decision
	Length	Height	Width	Weight	Quality
	4.7	1.8	1.7	1.7	high
	4.5	1.4	1.8	0.9	high
	4.7	1.8	1.9	1.3	high
	4.5	1.8	1.7	1.3	medium
	4.3	1.6	1.9	1.7	medium
	4.3	1.4	1.7	0.9	low
	4.5	1.6	1.9	0.9	very-low
	4.5	1.4	1.8	1.3	very-low

## 2

# El Algoritmo

- K-nearest neighbors
- Naïve Bayes
- Decision trees
- Support Vector Machine
- K-means
- Linear regression
- Logistic Regression
- **Neural networks**
  - **Perceptrons**
  - **FeedForward**
  - **RRN/LSTM**
  - **Convolutional**
  - **Transformers**

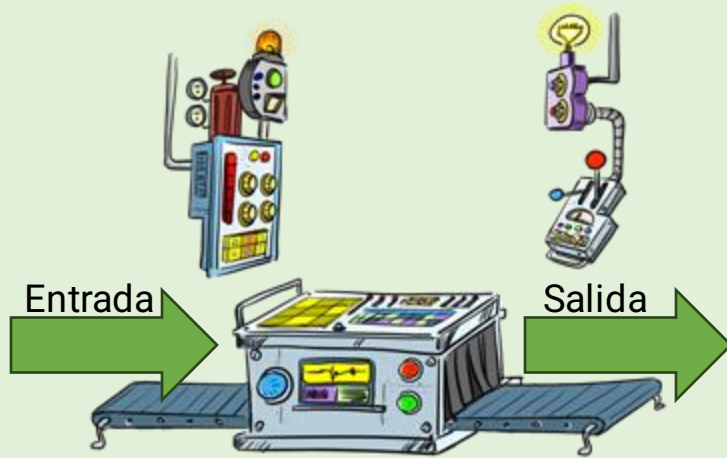


# 3

## El Modelo

El MODELO de Machine Learning contiene todas las reglas inducidas automáticamente

Podemos imaginarlo como una máquina, una función o un operador



Máquina



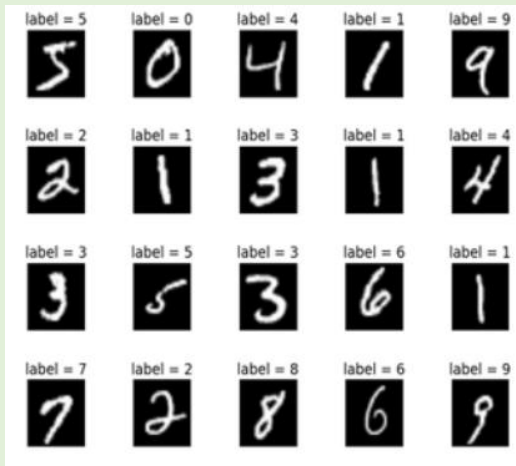
Función



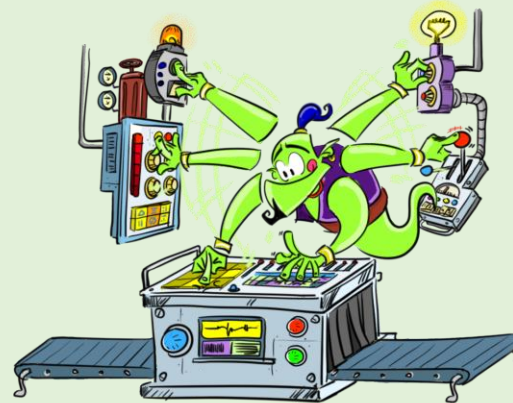
Operador

# El Proceso completo

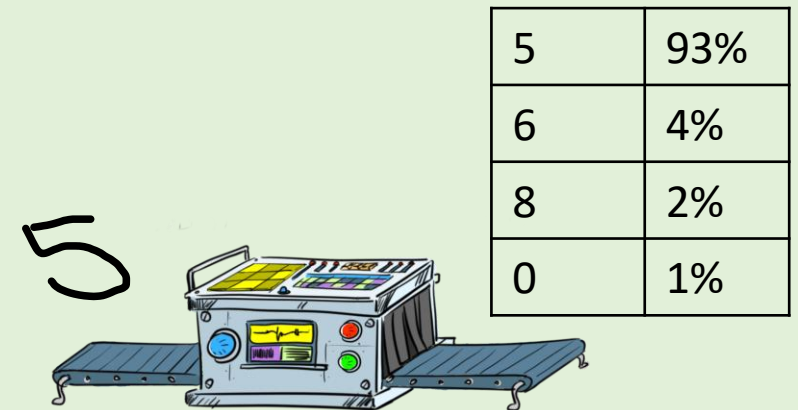
Construcción del dataset  
Fase de Entrenamiento



Construcción del modelo  
Fase de Aprendizaje



Evaluar el modelo  
Fase de Evaluación



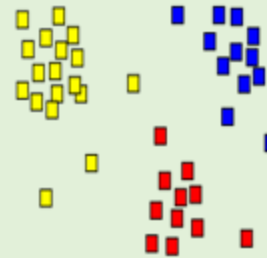
Poder de generalización del modelo  
Naturaleza probabilística

# Tipos de Machine Learning

Por estrategia de entrenamiento



Supervisado

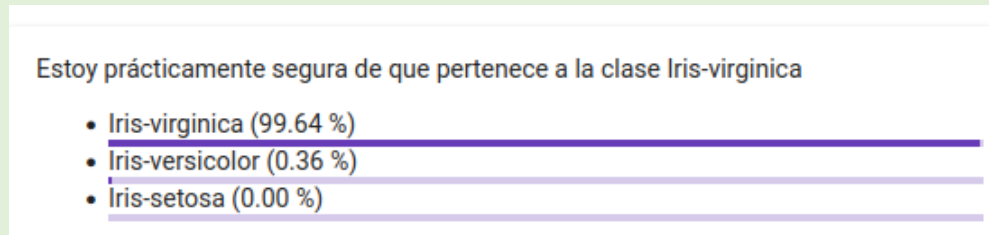


No supervisado

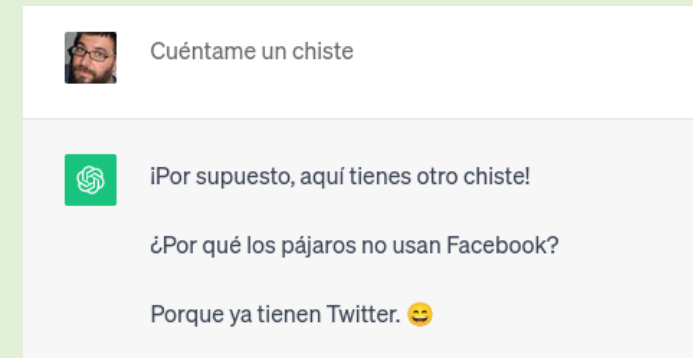


Reforzado

Por el tipo de salida



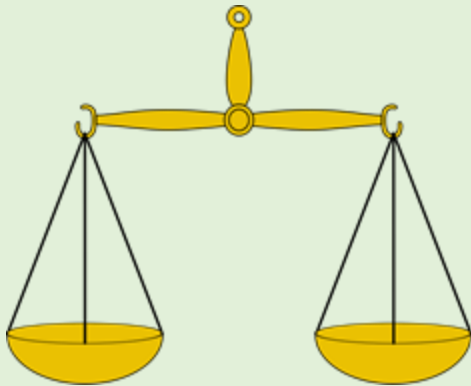
**Discriminativo:** clasifican o reconocen datos similares a los de entrenamiento



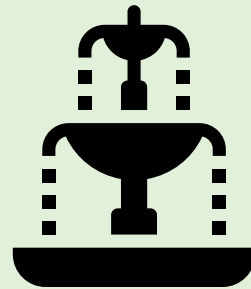
**Generativo:** generan datos similares a los de entrenamiento

# Recolección de Datos

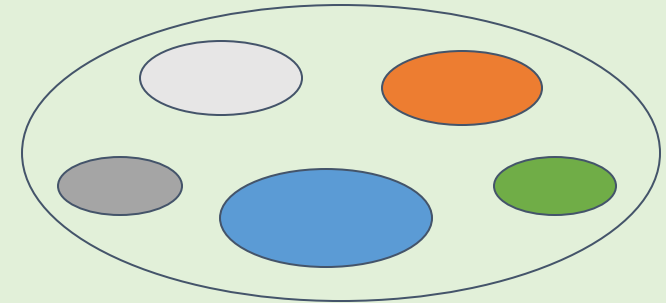
La **precisión** y **eficacia** de los modelos construidos por los algoritmos de ML depende en gran medida de **lo bueno que sea el conjunto de datos de entrenamiento**.



**Balanceado**, número parecido de ejemplos en cada clase (en aprendizaje supervisado)



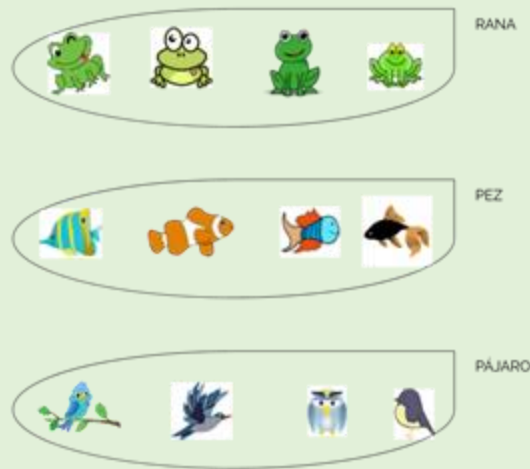
Las fuentes del sesgo



**Representativo** del problema, ejemplos que abarquen todo los tipos de cosas que deseamos clasificar o reconocer

# ¿Cómo aprende la máquina? (supervisado)

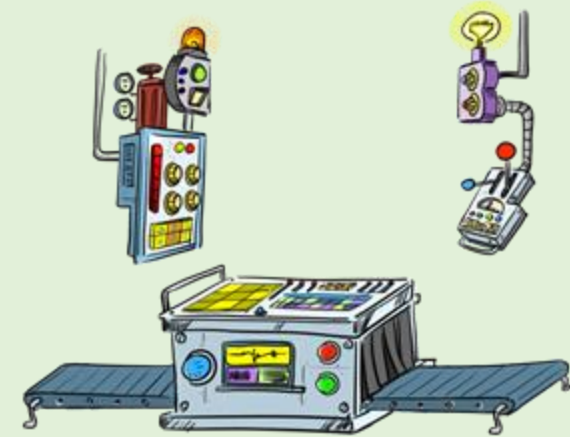
- Es la fase computacionalmente más compleja
- Existen muchos algoritmos de Machine Learning
- Todos construyen los modelos siguiendo una estrategia parecida



Dataset



Algoritmo de ML



Modelo



# Fase de aprendizaje (supervisado)

- El algoritmo construye una primera versión del modelo que, aunque clasifica los datos, lo hace mal.
- Entonces presentamos un primer ejemplo del dataset y, seguramente, el modelo fallará. De esto se da cuenta el algoritmo, pues los datos están etiquetados.



# Fase de aprendizaje (supervisado)

El algoritmo de ML se pone a trabajar para conseguir que, al menos, el primer dato que se le ha presentado, se clasifique bien



# Fase de aprendizaje (supervisado)

¡Y lo CONSIGUE!



# Fase de aprendizaje (supervisado)

Entonces presentamos otro ejemplo del dataset y, seguramente, el modelo fallará. De esto, de nuevo, se da cuenta el algoritmo, pues los datos están etiquetados.



# Fase de aprendizaje (supervisado)

El algoritmo de ML se pone a trabajar para conseguir que, al menos, los dos primeros datos que se le ha presentado, se clasifique bien



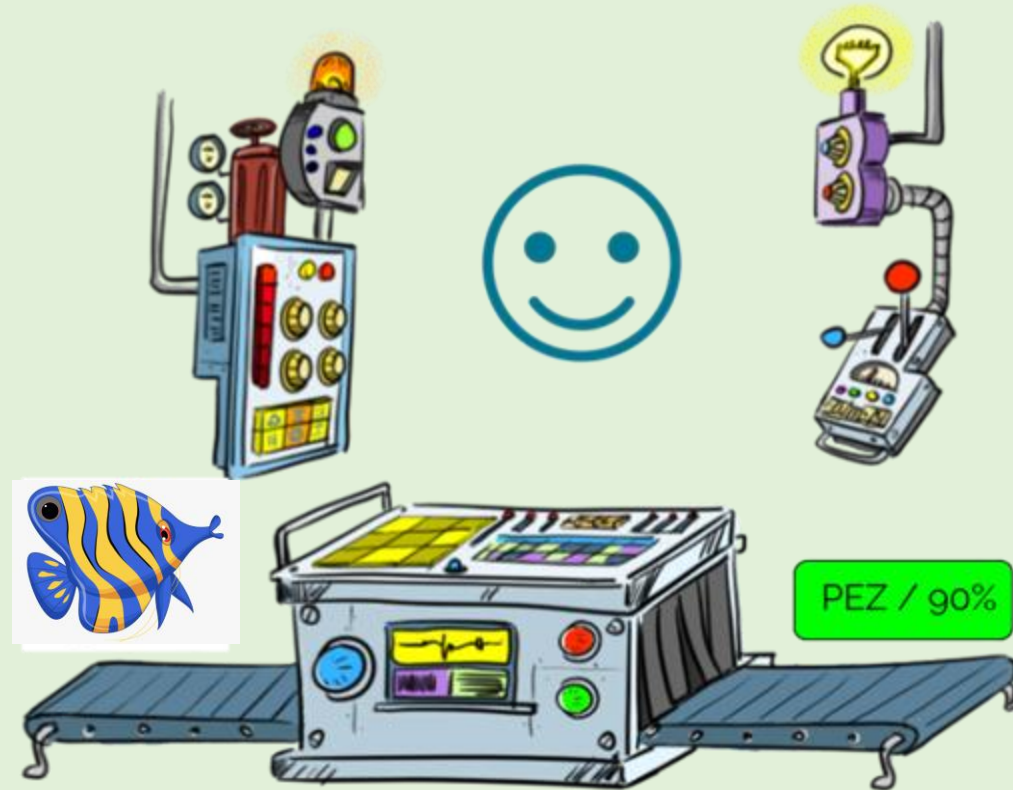
# Fase de aprendizaje (supervisado)

¡Y Lo CONSIGUE de nuevo!



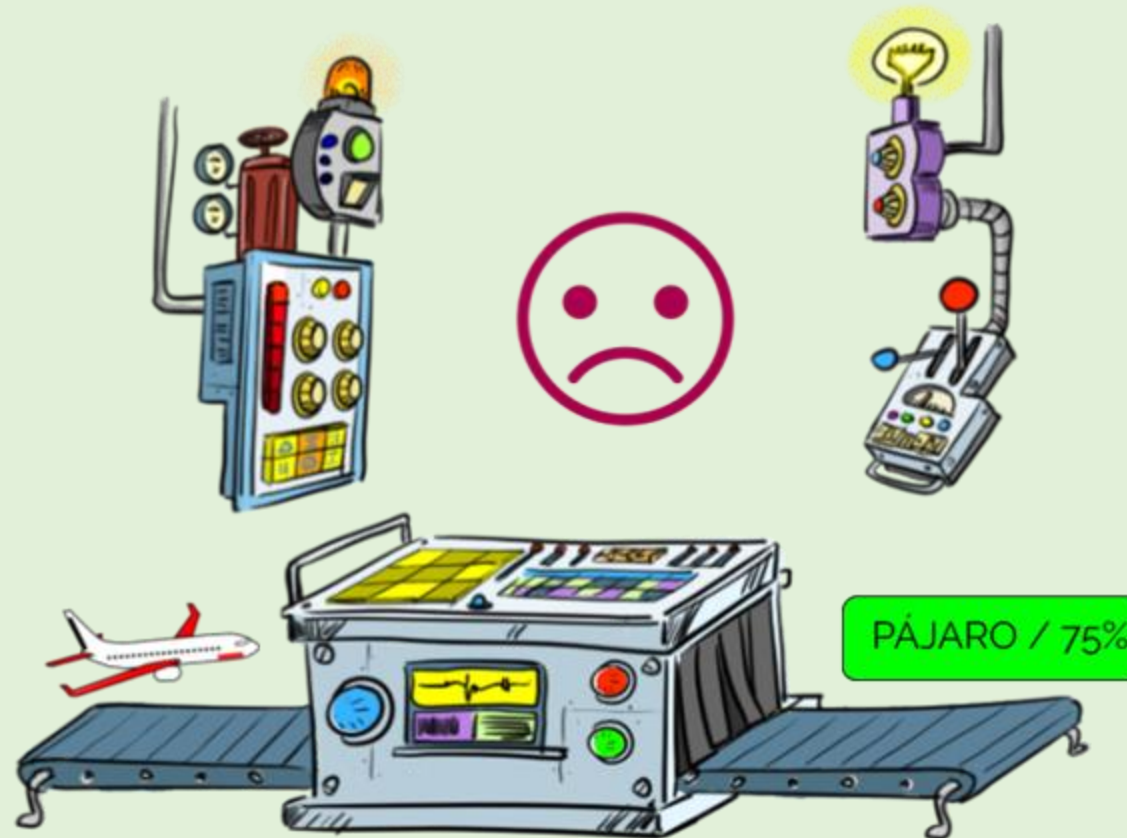
# Fase de aprendizaje (supervisado)

- Después de repetir este proceso con todos los datos del conjunto de entrenamiento (o dataset), el modelo es capaz de reconocer, no solo a los ejemplos, si no también a datos nuevos que no están entre los ejemplos.
- Ya no necesitamos al algoritmo de ML, pues el modelo funciona y generaliza lo suficientemente bien.



# Fase de aprendizaje (supervisado)

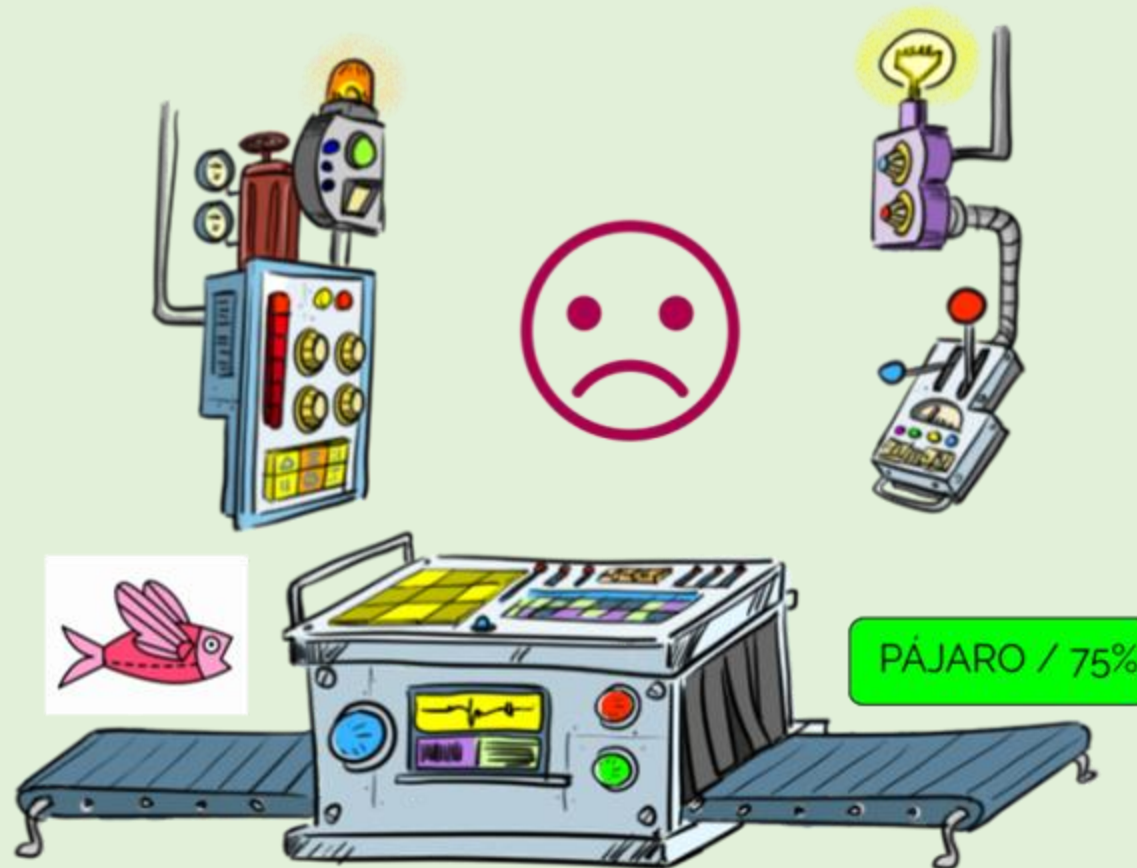
Pero, aunque sea capaz de generalizar y funciona bastante bien, no podemos asegurar que siempre funcione correctamente.



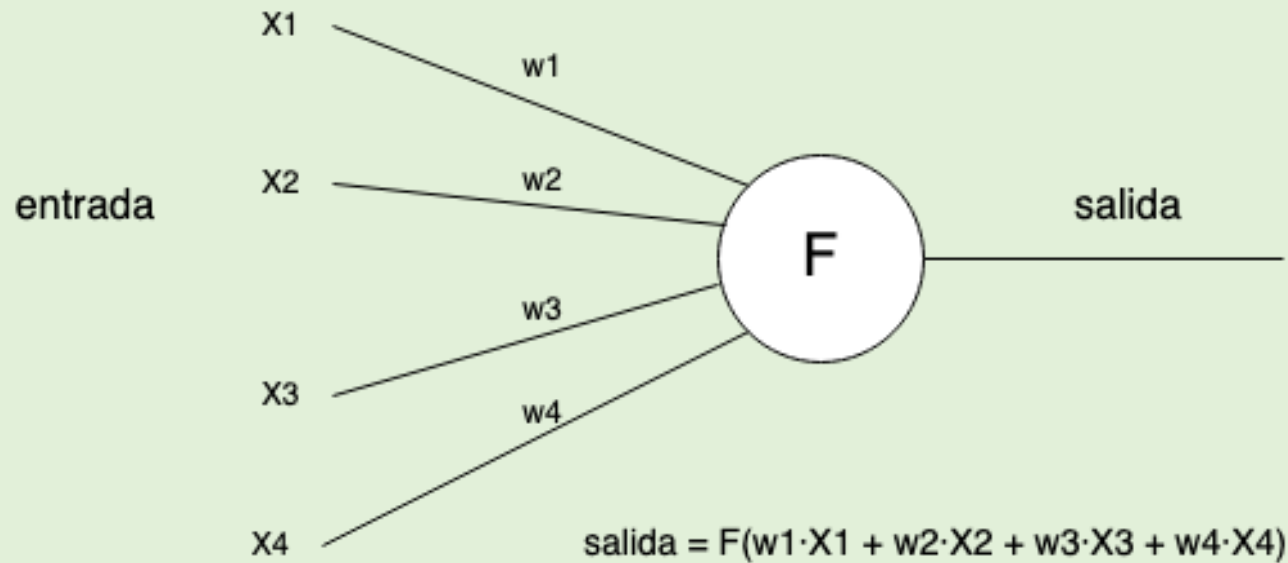


# Fase de aprendizaje (supervisado)

Incluso puede ocurrir que presentemos un ejemplar de una clase que sea muy distinto a los ejemplos que se usaron en esa clase y el modelo, de nuevo, falle.



# Redes neuronales: La neurona

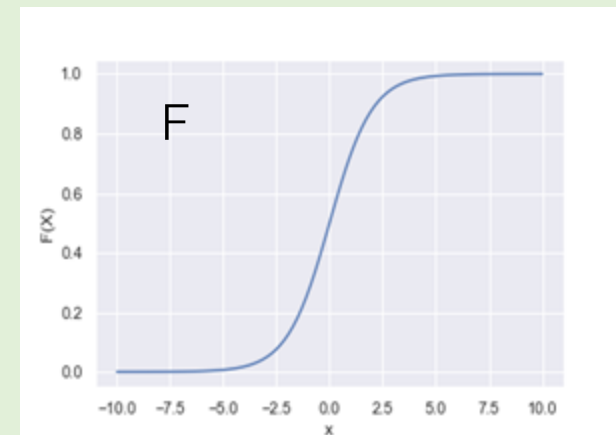


## Elementos de la neurona

Entrada:  $[X_1, X_2, X_3, X_4]$

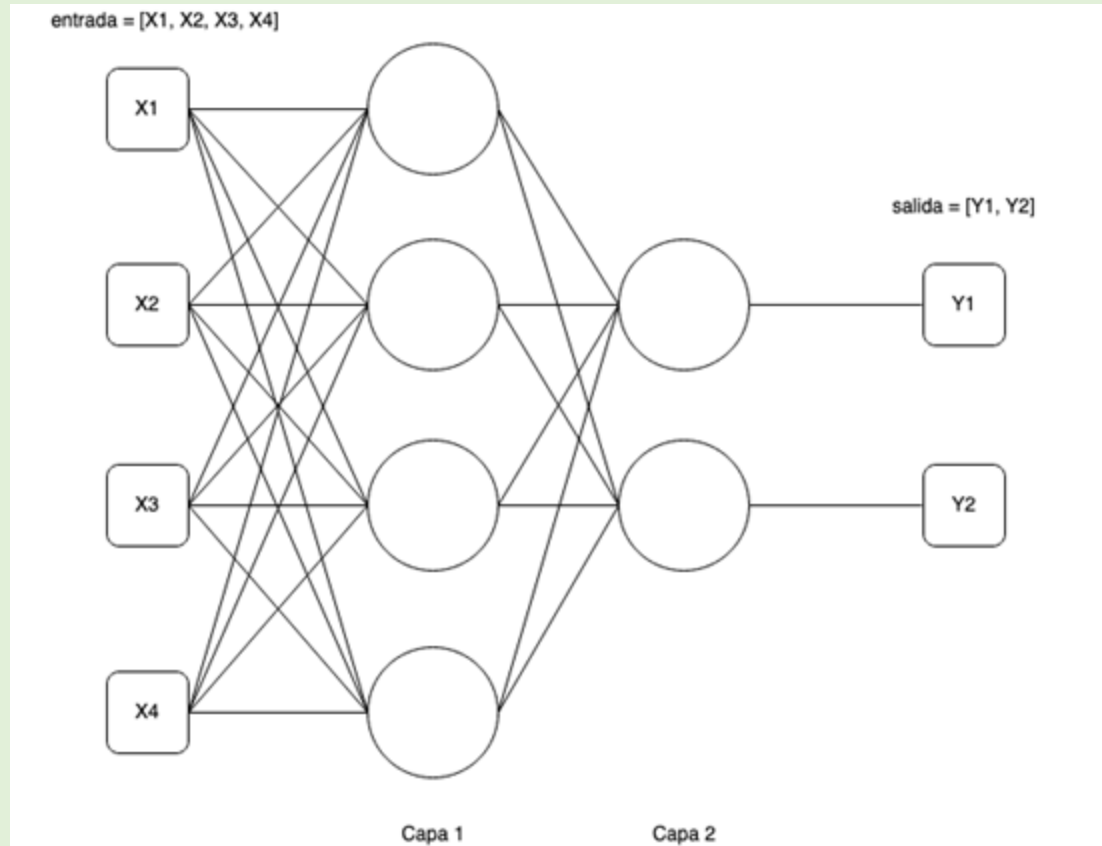
Pesos:  $w_1, w_2, w_3$  y  $w_4$

Salida = Función de activación:  $F(s)$   
 $s = w_1 \cdot X_1 + w_2 \cdot X_2 + w_3 \cdot X_3 + w_4 \cdot X_4$



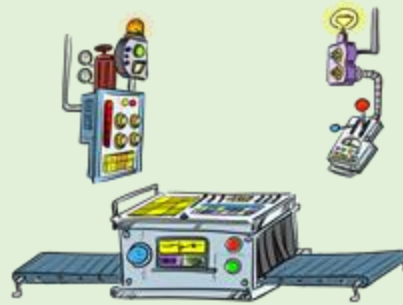
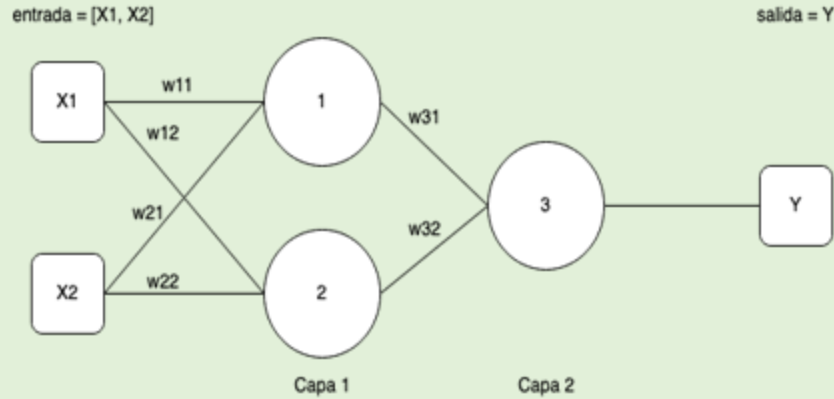
# Redes neuronales: FeedForward

No se muestran los pesos, en este ejemplo habría  
 $4 \times 4 + 2 \times 4 = 24$  pesos

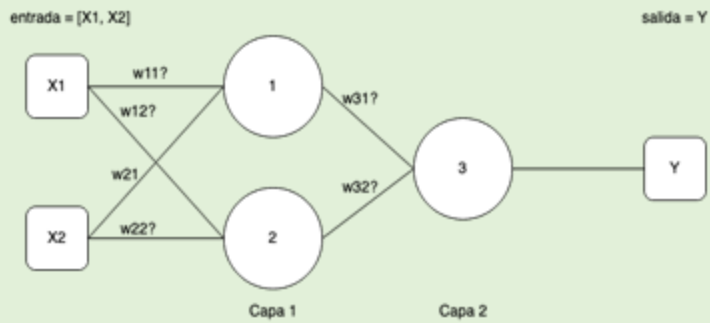


- Las neuronas se conectan entre sí para formar una red
- Una de las formas más típicas y que mejores resultados da es organizar la red por capas. Las salidas de cada neurona de una capa se conecta con las entradas de todas las neuronas de la capa siguiente.
- A esta configuración (arquitectura) se le denomina FeedForward Network

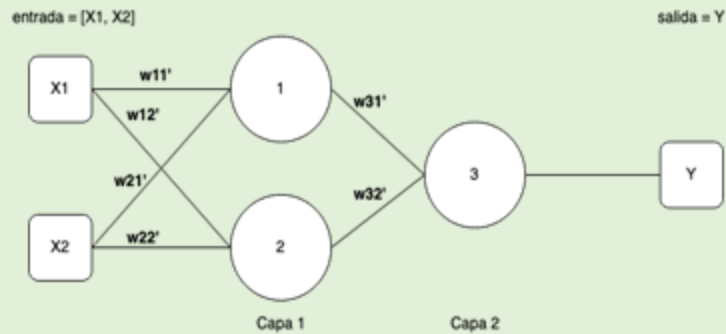
# Redes neuronales: aprendizaje



1. Pesos iniciales



2. Actualización de pesos



3. Nuevos pesos que mejoran el modelo

# Redes neuronales: El algoritmo (backpropagation)

- La red neuronal con el valor de sus pesos es el modelo de ML
- El algoritmo de ML lo que hace es buscar el valor de los pesos óptimo para que la red sea capaz de clasificar correctamente los datos de entrenamiento y también nuevos datos parecidos pero distintos a aquellos (generalización)

## Algoritmo de aprendizaje

1. Se dan valores aleatorios a los pesos de la red. Así ya se tiene un primer modelo, que no clasifica muy bien, pero funciona.
2. Los datos de entrenamiento se reparten en varios grupos (bunches) y para cada grupo:
  - a. Se calcula la salida de la red para cada dato del grupo (predicciones del modelo)
  - b. Se comparan las predicciones anteriores con la clasificación real (etiquetas)
  - c. Se usa el algoritmo **backpropagation** para actualizar el valor de los pesos, de manera que el error cometido en las predicciones se haga mínimo

Se repiten de forma iterativa estos pasos hasta que el error cometido sea aceptable.

# Codificación

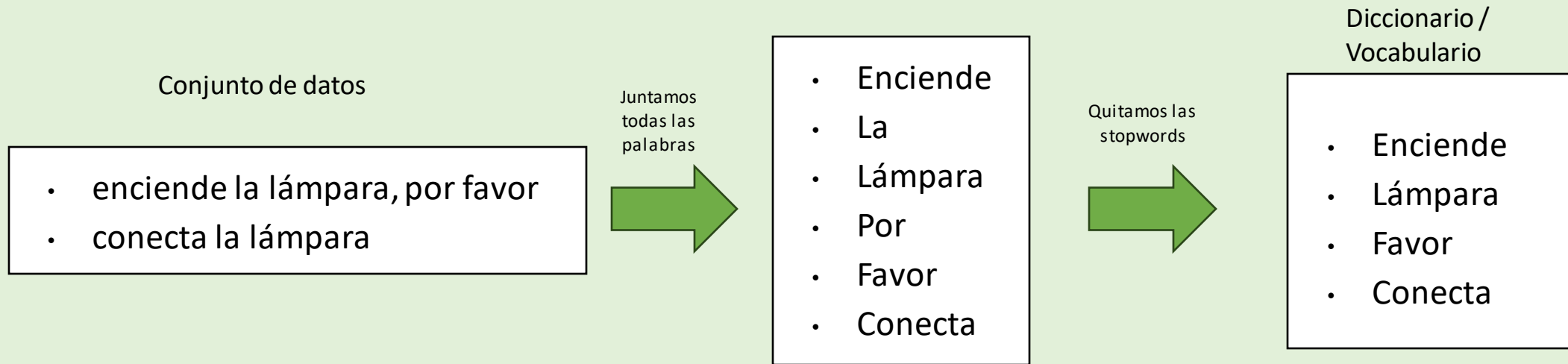
No hay magia en el ML, solo matemáticas (o mejor dicho, nada más y nada menos que matemáticas)

Por eso, antes de construir un modelo, tenemos que convertir nuestros datos en conjuntos de números, y estos últimos serán los que el algoritmo de ML analice para construir el modelo.

Texto	Representación numérica del texto	Representación numérica de la clase
enciende el ventilador, por favor	[1 0 0 1 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1]	0
apaga el ventilador, por favor	[0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1]	1

# Codificación: one hot encoding

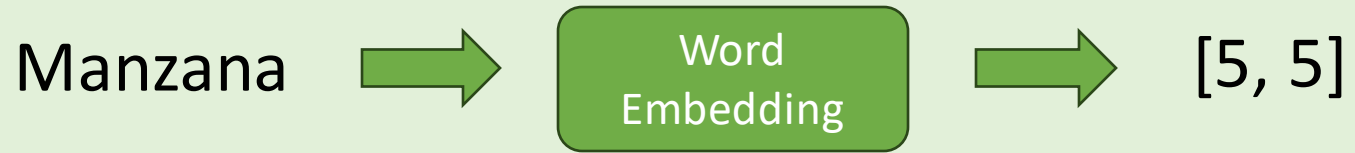
## 1. Construcción del diccionario



## 2. Codificación one-hot-encoding

enciende la lámpara, por favor	[1, 1, 1, 0]
conecta la lámpara	[0, 1, 0, 1]

# Codificación: word embedding

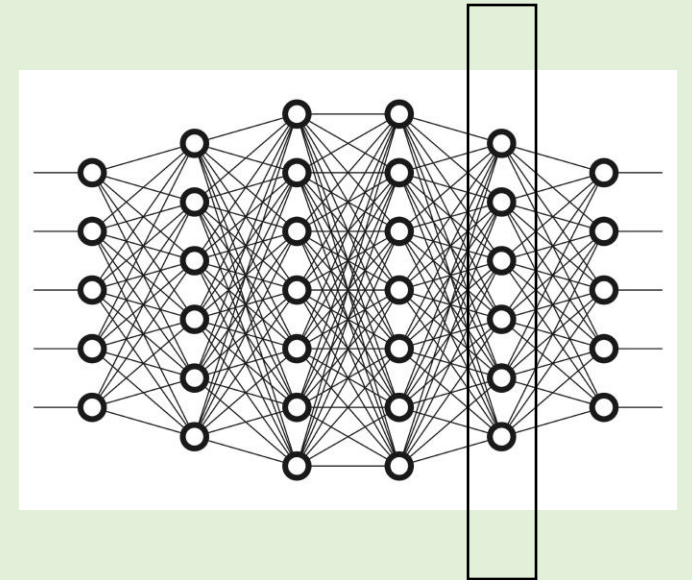


## Embeddings Quiz 1:

Where would you put the word "apple"?

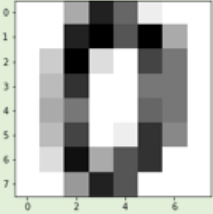
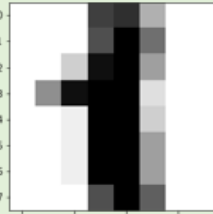


Embedding

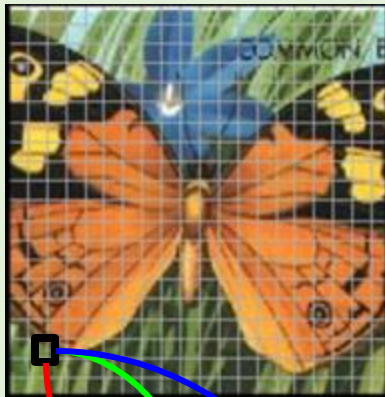




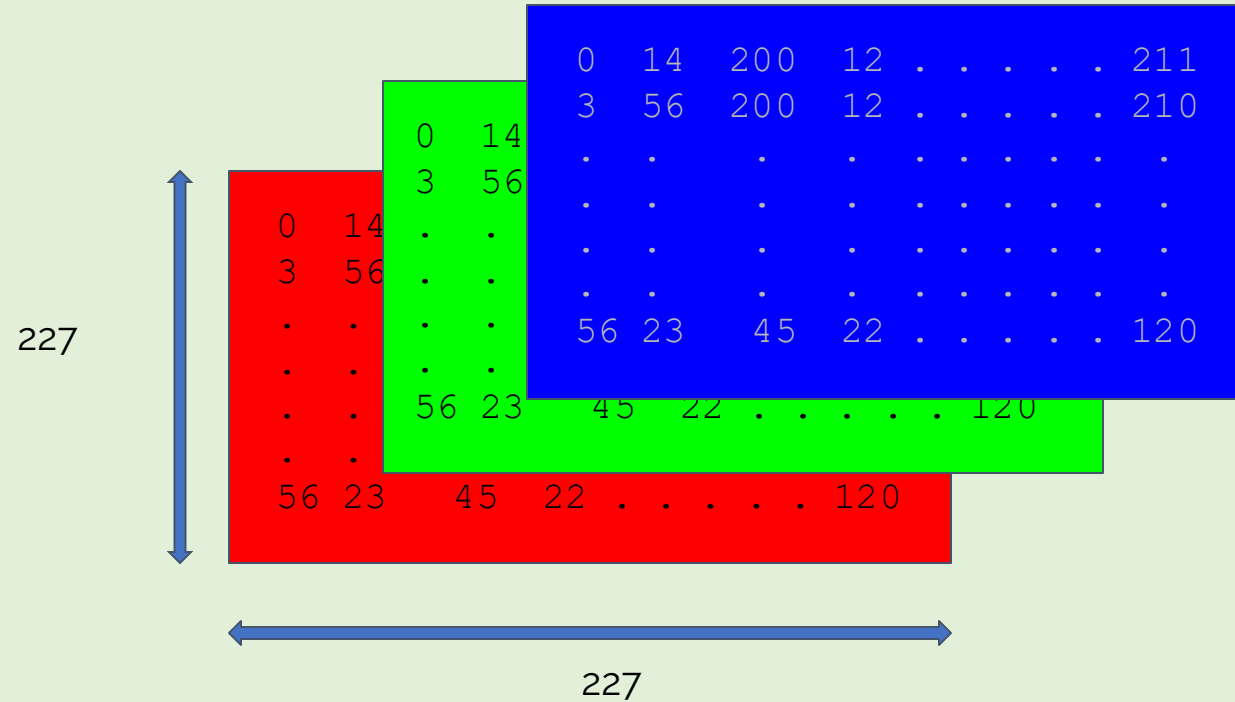
# Codificación: imagen escala de grises

Imágen del número	Representación numérica de la imágen	Representación numérica de la clase
	<pre>[ 0.,  0.,  5., 13.,  9.,  1.,  0.,  0.], [ 0.,  0., 13., 15., 10., 15.,  5.,  0.], [ 0.,  3., 15.,  2.,  0., 11.,  8.,  0.], [ 0.,  4., 12.,  0.,  0.,  8.,  8.,  0.], [ 0.,  5.,  8.,  0.,  0.,  9.,  8.,  0.], [ 0.,  4., 11.,  0.,  1., 12.,  7.,  0.], [ 0.,  2., 14.,  5., 10., 12.,  0.,  0.], [ 0.,  0.,  6., 13., 10.,  0.,  0.,  0.]</pre>	<p>0</p>
	<pre>[ 0.,  0.,  0., 12., 13.,  5.,  0.,  0.], [ 0.,  0.,  0., 11., 16.,  9.,  0.,  0.], [ 0.,  0.,  3., 15., 16.,  6.,  0.,  0.], [ 0.,  7., 15., 16., 16.,  2.,  0.,  0.], [ 0.,  0.,  1., 16., 16.,  3.,  0.,  0.], [ 0.,  0.,  1., 16., 16.,  6.,  0.,  0.], [ 0.,  0.,  1., 16., 16.,  6.,  0.,  0.], [ 0.,  0.,  0., 11., 16., 10.,  0.,  0.]</pre>	<p>1</p>

# Codificación: imagen a color

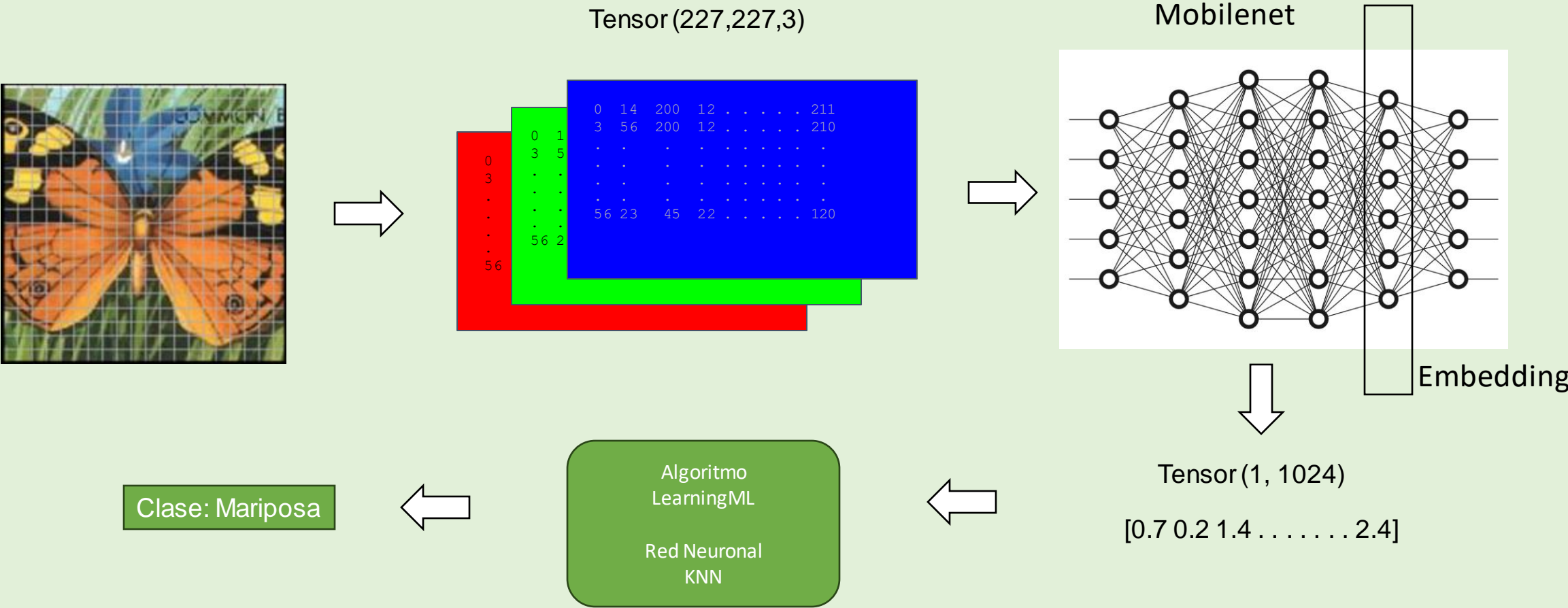


0-255 0-255 0-255



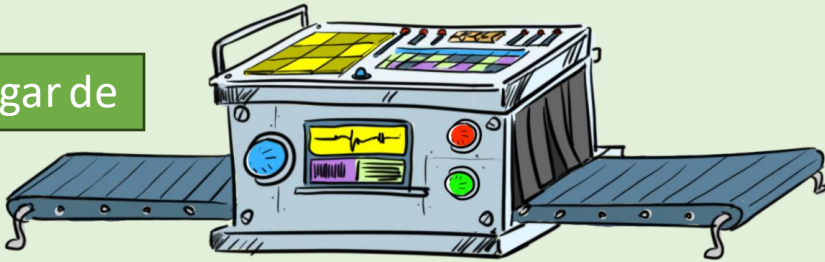
$227 \times 227 \times 3 = 154587$  números

# Codificación: imagen a color con embedding

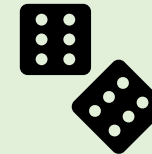


# LLM – Large Language Models (Grandes modelos del lenguaje)

En un lugar de

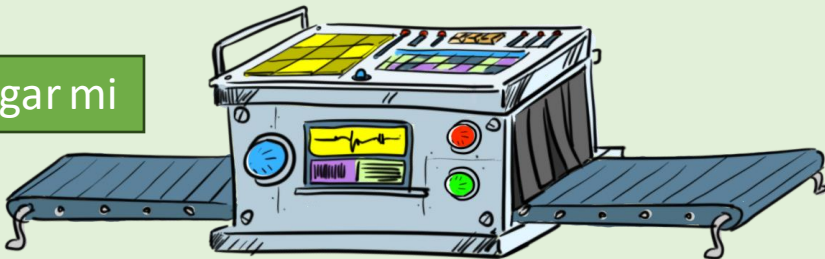


la	30%
mi	15%
su	10%
Madrid	1%

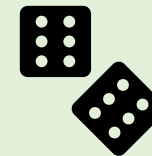


mi

En un lugar mi



corazón	23%
ciudad	15%
cuerpo	8%
memoria	3%



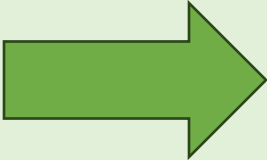
ciudad

# Cómo se entrenan los LLMs

## Corpus

En un lugar de la Mancha, de cuyo nombre no quiero acordarme, no ha mucho tiempo que vivía un hidalgo de los de lanza en astillero, adarga antigua, rocín flaco y galgo corredor. Una olla de algo más vaca que carnero, salpicón las más noches, duelos y quebrantos los sábados, lantejas los viernes, algún palomino de añadidura los domingos, consumían las tres partes de su hacienda. El resto della concluían sayo de velarte, calzas de velludo para las fiestas, con sus pantuflos de lo mismo, y los días de entresemana se honraba con su vellorí de lo más fino. Tenía en su casa una ama que pasaba de los cuarenta, y una sobrina que no llegaba a los veinte, y un mozo de campo y plaza, que así ensillaba el rocín como tomaba la podadera. Frisaba la edad de nuestro hidalgo con los cincuenta años; era de complexión recia, seco de carnes, enjuto de rostro, gran madrugador y amigo de la caza. Quieren decir que tenía el sobrenombre de Quijada, o Quesada, que en esto hay alguna diferencia en los autores que deste caso escriben; aunque, por conjeturas verosímiles, se deja entender que se llamaba Quejana. Pero esto importa poco a nuestro cuento; basta que en la narración dél no se salga un punto de la verdad.

Extracción  
Del dataset



## Texto

## Etiqueta

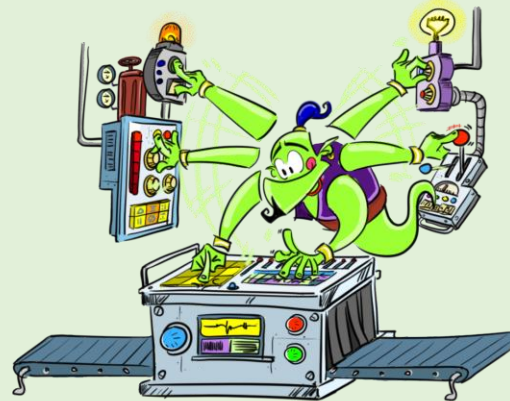
En un lugar de	la
un lugar de la	Mancha
lugar de la Mancha	,
de la Mancha,	de
La Mancha, de	cuyo
Mancha, de cuyo	nombre
, de cuyo nombre	no
de cuyo nombre no	quiero
...	...

# El Proceso completo

## Entrenamiento

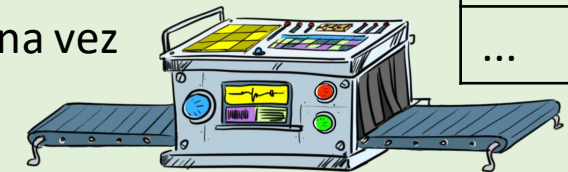
En un lugar de	la
un lugar de la	Mancha
lugar de la Mancha	,
de la Mancha,	de
La Mancha, de	cuyo
Mancha, de cuyo	nombre
, de cuyo nombre	no
de cuyo nombre no	quiero
...	...

## Construcción del LLM Aprendizaje



## Evaluar / usar el modelo

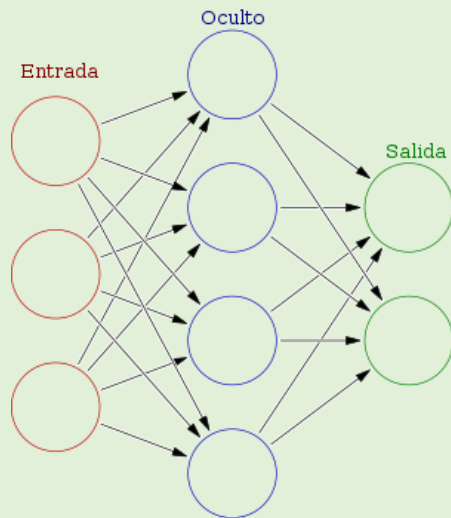
Erase una vez



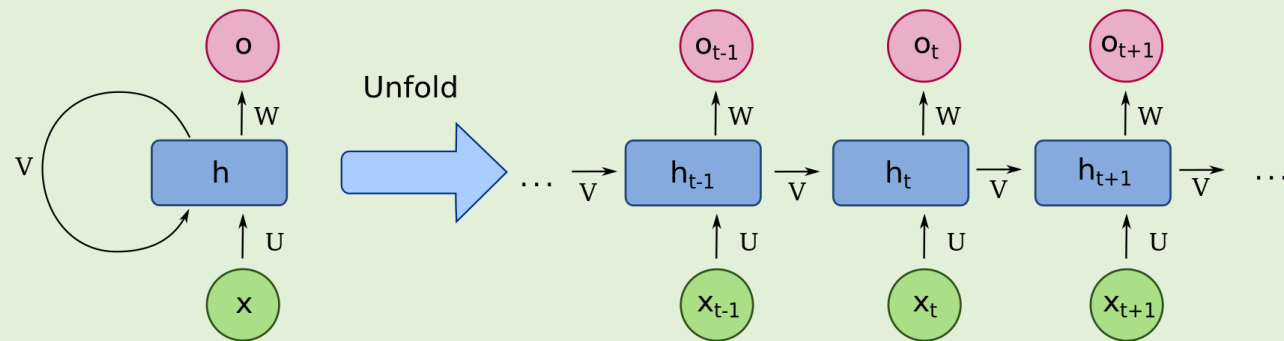
un	30%
una	15%
hace	10%
...	...

# Qué algoritmo usar para construir LLMs

- Lo primero: con una ingente cantidad de textos, millones y millones de palabras
- Con técnicas basadas en deep learning: redes neuronales con muchas capas
- Se puede usar cualquier arquitectura:
  - Feedforward Networks (muy ineficaz para estas cosas)
  - RNN/LSTM Networks (da buenos resultados pero, debido a su naturaleza temporal, requiere muchísimo tiempo de entrenamiento)
  - Transformers (la buena del tó)



Feedforward



RNN: [Recurrent Neural Network](#)

# La arquitectura transformer (2017)

Provided proper attribution is provided, Google hereby grants permission to reproduce the tables and figures in this paper solely for use in journalistic or scholarly works.

## Attention Is All You Need

Ashish Vaswani<sup>\*</sup>  
Google Brain  
avaswani@google.com

Noam Shazeer<sup>\*</sup>  
Google Brain  
noam@google.com

Niki Parmar<sup>\*</sup>  
Google Research  
nikip@google.com

Jakob Uszkoreit<sup>\*</sup>  
Google Research  
usz@google.com

Llion Jones<sup>\*</sup>  
Google Research  
llion@google.com

Aidan N. Gomez<sup>\* †</sup>  
University of Toronto  
aidan@cs.toronto.edu

Lukasz Kaiser<sup>\*</sup>  
Google Brain  
lukasz@kaiser@google.com

Illia Polosukhin<sup>\* ‡</sup>  
illia.polosukhin@gmail.com

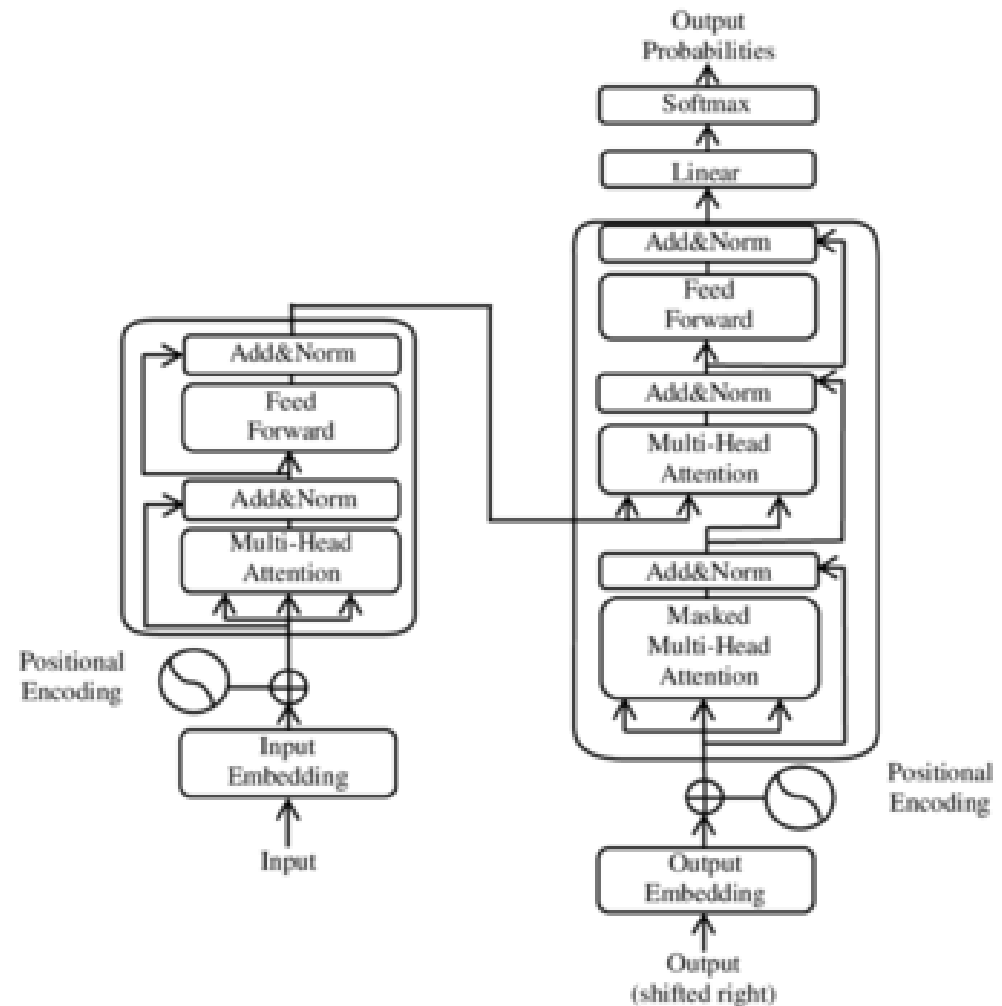
### Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

<sup>\*</sup>Equal contribution. Listing order is random. Jakob proposed replacing RNNs with self-attention and started the effort to evaluate this idea. Ashish, with Illia, designed and implemented the first Transformer models and has been crucially involved in every aspect of this work. Noam proposed scaled dot-product attention, multi-head attention and the parameter-free position representation and became the other person involved in nearly every detail. Niki designed, implemented, tuned and evaluated countless model variants in our original codebase and tensor2tensor. Llion also experimented with novel model variants, was responsible for our initial codebase, and efficient inference and visualizations. Lukasz and Aidan spent countless long days designing various parts of and implementing tensor2tensor, replacing our earlier codebase, greatly improving results and massively accelerating our research.

<sup>†</sup>Work performed while at Google Brain.

<sup>‡</sup>Work performed while at Google Research.





# Timeline de los LLMs con nº de parámetros



# The Hugging Face: El punto de encuentro de los ML geeks

The screenshot displays the Hugging Face website interface. At the top, there is a search bar with the text "Search models, datasets, users..." and navigation links for "Models", "Datasets", "Spaces", "Docs", "Solutions", and "Pricing". The main content area is titled "Models 345,116" and includes a "Filter by name" search bar and a "Sort: Trending" dropdown. The models are listed in a grid format, each with a card containing the model name, category, and statistics. The left sidebar contains navigation tabs for "Tasks", "Libraries", "Datasets", "Languages", "Licenses", and "Other", along with a "Filter Tasks by name" search bar. The sidebar is organized into categories: "Multimodal" (Feature Extraction, Text-to-Image, Image-to-Text, Text-to-Video, Visual Question Answering, Document Question Answering, Graph Machine Learning), "Computer Vision" (Depth Estimation, Image Classification, Object Detection, Image Segmentation, Image-to-Image, Unconditional Image Generation, Video Classification, Zero-Shot Image Classification), "Natural Language Processing" (Text Classification, Token Classification, Table Question Answering, Question Answering, Zero-Shot Classification, Translation, Summarization, Conversational, Text Generation, Text2Text Generation, Fill-Mask, Sentence Similarity), and "Audio" (Text-to-Speech, Automatic Speech Recognition).

**Tasks** Libraries Datasets Languages Licenses Other

Filter Tasks by name

Multimodal

- Feature Extraction
- Text-to-Image
- Image-to-Text
- Text-to-Video
- Visual Question Answering
- Document Question Answering
- Graph Machine Learning

Computer Vision

- Depth Estimation
- Image Classification
- Object Detection
- Image Segmentation
- Image-to-Image
- Unconditional Image Generation
- Video Classification
- Zero-Shot Image Classification

Natural Language Processing

- Text Classification
- Token Classification
- Table Question Answering
- Question Answering
- Zero-Shot Classification
- Translation
- Summarization
- Conversational
- Text Generation
- Text2Text Generation
- Fill-Mask
- Sentence Similarity

Audio

- Text-to-Speech
- Automatic Speech Recognition

**Models** 345,116 Filter by name

new Full-text search Sort: Trending

- mistralai/Mistral-7B-v0.1**  
Text Generation • Updated about 12 hours ago • ↓ 7.36k • ♥ 356
- mistralai/Mistral-7B-Instruct-v0.1**  
Text Generation • Updated about 15 hours ago • ↓ 7.46k • ♥ 243
- monster-labs/control\_v1p\_sd15\_qrcode\_monster**  
Updated Jul 21 • ↓ 423k • ♥ 813
- stabilityai/stable-diffusion-xl-base-1.0**  
Text-to-Image • Updated 24 days ago • ↓ 3.26M • ♥ 2.88k
- microsoft/phi-1\_5**  
Text Generation • Updated 1 day ago • ↓ 79.6k • ♥ 837
- Qwen/Qwen-14B-Chat**  
Text Generation • Updated about 23 hours ago • ↓ 3.07k • ♥ 117
- Xwin-LM/Xwin-LM-70B-V0.1**  
Text Generation • Updated 8 days ago • ↓ 3.54k • ♥ 137
- meta-llama/Llama-2-7b**  
Text Generation • Updated Jul 19 • ♥ 2.62k
- llyyasviel/sd\_control\_collection**  
Updated 20 days ago • ♥ 475
- Qwen/Qwen-14B**  
Text Generation • Updated about 23 hours ago • ↓ 1.11k • ♥ 110
- meta-llama/Llama-2-7b-chat-hf**  
Text Generation • Updated Aug 9 • ↓ 1.02M • ♥ 1.28k
- tiiuae/falcon-180B**  
Text Generation • Updated 23 days ago • ↓ 60k • ♥ 791
- TheBloke/Mistral-7B-Instruct-v0.1-GGUF**  
Text Generation • Updated about 10 hours ago • ↓ 193 • ♥ 60
- internlm/internlm-chat-20b**  
Text Generation • Updated 7 days ago • ↓ 4.52k • ♥ 102
- coqui/XTTS-v1**  
Text-to-Speech • Updated 1 day ago • ↓ 54 • ♥ 217
- LingxinAI/CharacterGLM-6b**  
Updated 7 days ago • ♥ 44
- runwayml/stable-diffusion-v1-5**  
Text-to-Image • Updated Aug 23 • ↓ 9.19M • ♥ 9.3k
- pfnet/plamo-13b**  
Text Generation • Updated 1 day ago • ↓ 1.77k • ♥ 42

# Un ejemplo de juguete de Modelo del Lenguaje

```
▶ start = time.time()
  states = None
  next_char = tf.constant(['con la iglesia'])
  result = [next_char]

  for n in range(1000):
    next_char, states = one_step_model.generate_one_step(next_char, states=states)
    result.append(next_char)

  result = tf.strings.join(result)
  end = time.time()
  print(result[0].numpy().decode('utf-8'), '\n\n' + '_'*80)
  print('\nRun time:', end - start)
```

↳ con la iglesia, que en las obligaciones de Herajó de tu jurisdicción al grito y médico de su malandante alcalde.

Y, aunque ya no venís, que, aunque no eran malos.

Había de reír y dejar de encomer la insula pidió a mi señora, pensarán el secretario. Otro lícitos no fueran en tres ánimo, ni metador, sino en semejantes libros de la Lusceansada vida y afinco de sus amores, que sin mocedad siquiera el camino que vivo? Cuanto más, que ya sé yo que iba bien podridad y escantorniciere, y pueda sobrina ni imaginar cuán neciamente se lo cayeron; y así, le dijo:

– Cuerpo de mis compañeros, y uno de los que el bolso grado. Otro Bellanesto día oyó entre sí al ventero, don Quijote y su escudero, de cuyo nombre se quieren estar a media noche, y muy por entonces no sé con la visera, si le huelen, me esforcéis, florquine; y, así como los que somos barbados a caballo, y seando yos pensamientos de invidias, éspera y tenía, puesto que les sería descuido; y, como si fuera blanda y amarse de Cervantes,

---

Run time: 3.2226364612579346

<https://colab.research.google.com/drive/1yfHyJkozvV5g3fxAROfNiXHR8VNAXpix?usp=sharing>



# Loros estocásticos

## On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?

Emily M. Bender\*  
ebender@uw.edu  
University of Washington  
Seattle, WA, USA

Angelina McMillan-Major  
aymm@uw.edu  
University of Washington  
Seattle, WA, USA

Timnit Gebru\*  
timnit@blackinai.org  
Black in AI  
Palo Alto, CA, USA

Shmargaret Shmitchell  
shmargaret.shmitchell@gmail.com  
The Aether


### ABSTRACT

The past 3 years of work in NLP have been characterized by the development and deployment of ever larger language models, especially for English. BERT, its variants, GPT-2/3, and others, most recently Switch-C, have pushed the boundaries of the possible both through architectural innovations and through sheer size. Using these pretrained models and the methodology of fine-tuning them for specific tasks, researchers have extended the state of the art on a wide array of tasks as measured by leaderboards on specific benchmarks for English. In this paper, we take a step back and ask: How big is too big? What are the possible risks associated with this technology and what paths are available for mitigating those risks? We provide recommendations including weighing the environmental and financial costs first, investing resources into curating and carefully documenting datasets rather than ingesting everything on the web, carrying out pre-development exercises evaluating how the planned approach fits into research and development goals and supports stakeholder values, and encouraging research directions beyond ever larger language models.

### CCS CONCEPTS

• Computing methodologies → Natural language processing.

### ACM Reference Format:

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? . In *Conference on Fairness, Accountability, and Transparency (FAccT '21)*, March 3–10, 2021, Virtual Event, Canada. ACM, New York, NY, USA, 14 pages. <https://doi.org/10.1145/3442188.3445922>

### 1 INTRODUCTION

One of the biggest trends in natural language processing (NLP) has been the increasing size of language models (LMs) as measured by the number of parameters and size of training data. Since 2018

\*joint first authors



This work is licensed under a Creative Commons Attribution International 4.0 License.

FAccT '21, March 3–10, 2021, Virtual Event, Canada  
ACM ISBN 978-1-4503-8309-7/21/03.  
<https://doi.org/10.1145/3442188.3445922>

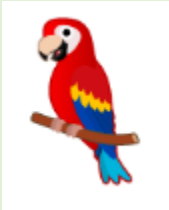
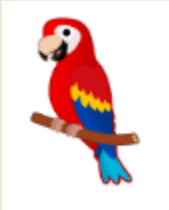
alone, we have seen the emergence of BERT and its variants [39, 70, 74, 113, 146], GPT-2 [106], T-NLG [112], GPT-3 [25], and most recently Switch-C [43], with institutions seemingly competing to produce ever larger LMs. While investigating properties of LMs and how they change with size holds scientific interest, and large LMs have shown improvements on various tasks (§2), we ask whether enough thought has been put into the potential risks associated with developing them and strategies to mitigate these risks.

We first consider environmental risks. Echoing a line of recent work outlining the environmental and financial costs of deep learning systems [129], we encourage the research community to prioritize these impacts. One way this can be done is by reporting costs and evaluating works based on the amount of resources they consume [37]. As we outline in §3, increasing the environmental and financial costs of these models doubly punishes marginalized communities that are least likely to benefit from the progress achieved by large LMs and most likely to be harmed by negative environmental consequences of its resource consumption. At the scale we are discussing (outlined in §2), the first consideration should be the environmental cost.

Just as environmental impact scales with model size, so does the difficulty of understanding what is in the training data. In §4, we discuss how large datasets based on texts from the Internet overrepresent hegemonic viewpoints and encode biases potentially damaging to marginalized populations. In collecting ever larger datasets we risk incurring documentation debt. We recommend mitigating these risks by budgeting for curation and documentation at the start of a project and only creating datasets as large as can be sufficiently documented.

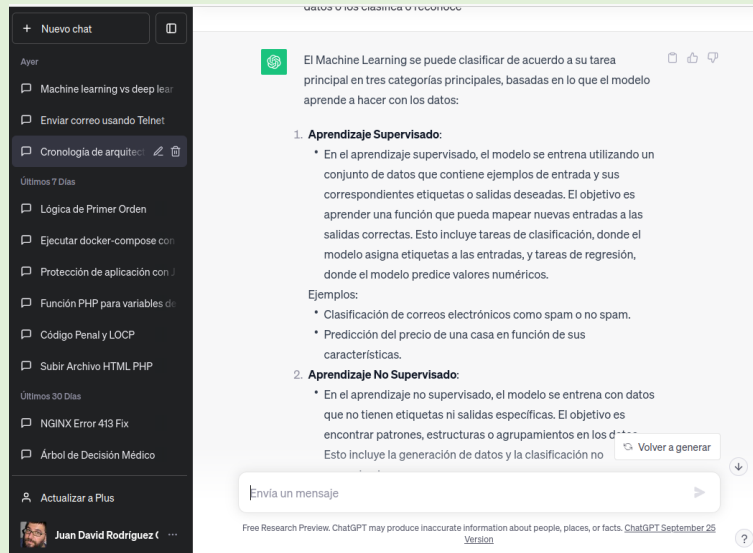
As argued by Bender and Koller [14], it is important to understand the limitations of LMs and put their success in context. This not only helps reduce hype which can mislead the public and researchers themselves regarding the capabilities of these LMs, but might encourage new research directions that do not necessarily depend on having larger LMs. As we discuss in §5, LMs are not performing natural language understanding (NLU), and only have success in tasks that can be approached by manipulating linguistic form [14]. Focusing on state-of-the-art results on leaderboards without encouraging deeper understanding of the mechanism by which they are achieved can cause misleading results as shown

- Costes medioambientales y de financiación
  - Entrenar un modelo BERT requiere tanta energía como un vuelo transoceánico
  - Humano medio origina 5t CO2 al año, entrenar un LLM Transformer 284t CO2
- Datos de entrenamiento insondables
  - Se ha demostrado que los texto de entrenamiento extraídos de Internet presentan estereotipos y asociaciones despectivas de género, raza, etnia y discapacidades
  - LLM entrenados en grandes conjuntos de datos estáticos, no curados de La web codifica puntos de vista hegemónicos que son perjudiciales para las poblaciones marginadas.
- Alucinaciones: Frases perfectamente construidas pero que son incorrectas, sin sentidos o irreales.



# Herramientas IA para la generación de textos

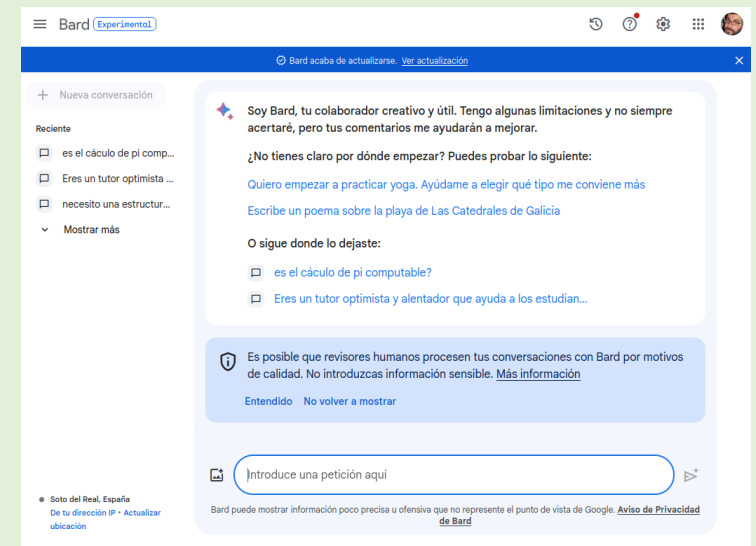
## Chat GPT



## Microsoft Bing chat



## Google Bard



La clave: saber preguntar



"Ingeniería" de PROMPTS



# ChatGPT y similares como herramientas de asistencia al aprendizaje

Las propuestas educativas de uso de ChatGPT y similares se reducen a usar trucos de prompts con los que el estudiante o el profesor pueden consultar a la IA aspectos relacionados con la tarea que están realizando

Ejemplo de estructura de prompt:

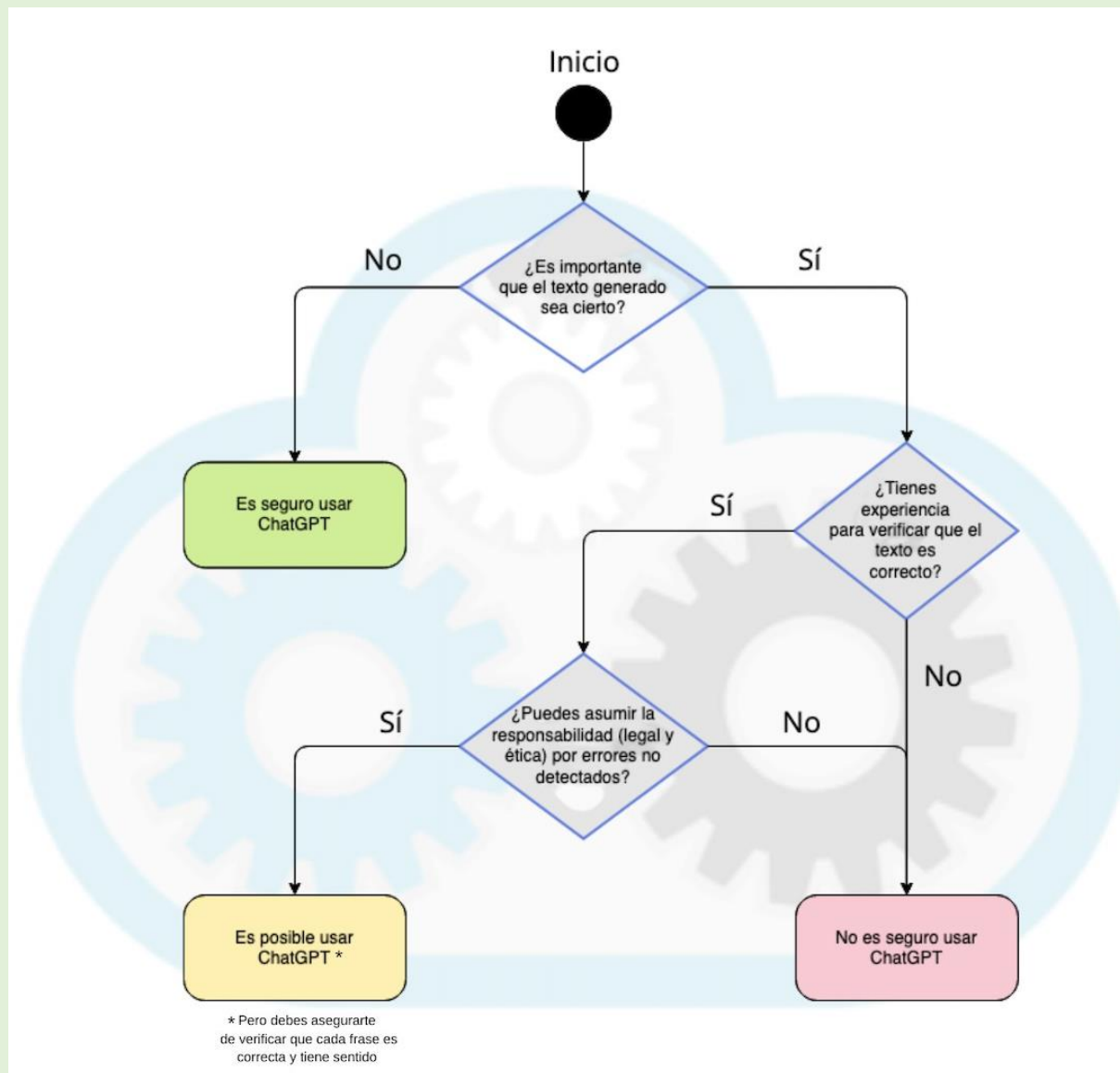
- Dí a la IA quien es (rol)
- Dí a la IA lo que quieres que haga
- Dale instrucciones paso a paso
- Personaliza la pregunta
- Añade tu pedagogía

[Generación de explicaciones de conceptos](#)

[Generación de ejemplos sobre un concepto](#)

[Generación de cuestionarios](#)

# Cómo usar chat GPT



**Usar trucos de prompts  
con herramientas de IA**

**Construir nuestros  
propios sistemas de IA**

**Equivale a saber *leer***

**Equivale a saber *leer y escribir***

**Genera sensación de *caja negra***

**Contribuye a su explicabilidad**

**Fomenta el antropomorfismo**

**Desmitifica y protege frente a *hype***

**Conocimiento efímero y  
poco generalizable**

**Conocimiento perenne y  
generalizable**



Para saber más:  
<https://programamos.es/ia/>





# 5.

¿Dónde puedo encontrar actividades y herramientas para enseñar IA y ML en mi asignatura?

Todos los datasets usados en los ejemplos de este taller te los puedes descargar en el siguiente enlace:

<https://1drv.ms/u/s!Aic1RMUsB4nkh7IBp95b34Kbgio6yQ?e=rTfMja>

# Proyecto REA / DUA Andalucía

**Mi asistente personal** Página 1 de 14 Menú

Mi asistente personal

1. IA para mejorar nuestra vida
2. Asistente personal
3. Una máquina que aprende
4. Introducción a la IA
5. Agentes inteligentes
6. El asistente personal
7. Presenta tu trabajo
8. Sacamos conclusiones

Créditos

## Mi asistente personal

Otra publicada con Licencia Creative Commons Reconocimiento No comercial Compartir Igual 4.0

Página 1 de 14

**Somos inteligentes** Página 1 de 16 Menú

Somos inteligentes

1. Ayúdame a clasificar
2. ¿Cómo aprendes?
3. Conocemos un poco más
4. Empieza a enseñar
5. Mineral o roca
6. Presenta tu trabajo
7. Sacamos conclusiones

Créditos

## Somos inteligentes

Diccionario Lectura Facilitada

Otra publicada con Licencia Creative Commons Reconocimiento Compartir Igual 4.0

Página 1 de 16

**Periodismo de datos** Página 1 de 14 Menú

Periodismo de datos

1. ¿Qué vamos a hacer?
2. Antes de empezar vamos a recordar
3. Recordando la importancia de los datos
4. El poder de los datos
5. Vamos a crear nuestro asistente ficticio
6. Compartimos nuestro trabajo
7. Sacamos conclusiones

Créditos

## Periodismo de datos

Otra publicada con Licencia Creative Commons Reconocimiento No comercial Compartir Igual 4.0

Página 1 de 14

**Hola, ¿necesitas ayuda?** Página 1 de 14 Menú

Hola, ¿necesitas ayuda?

1. ¿Qué vamos a hacer?
2. Antes de empezar debemos recordar
3. ¿Por qué necesitamos la Inteligencia Artificial?
4. ¿Qué es la Inteligencia Artificial?
5. Entorno de programación
6. Creamos nuestro Chatbot
7. Compartimos nuestro trabajo
8. Sacamos conclusiones

Créditos

## Hola, ¿necesitas ayuda?

Otra publicada con Licencia Creative Commons Reconocimiento No comercial Compartir Igual 4.0

Página 1 de 14

**Nos comunicamos con la Lengua de Signos** Página 1 de 16 Menú

Nos comunicamos con la Lengua de Signos

1. Una imagen vale más que mil palabras
2. Inteligencia Artificial... Dejame pensar
3. El poder de la Inteligencia Artificial
4. Aprendamos más sobre el reconocimiento de imágenes
5. Desarrollamos nuestro reto
6. Mostramos nuestro trabajo
7. Para terminar sacamos conclusiones

Créditos

## Nos comunicamos con la Lengua de Signos

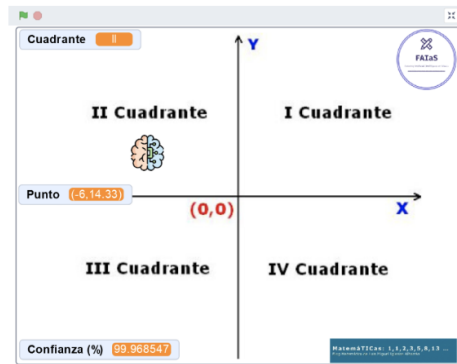
Otra publicada con Licencia Creative Commons Reconocimiento No comercial Compartir Igual 4.0

Página 1 de 16

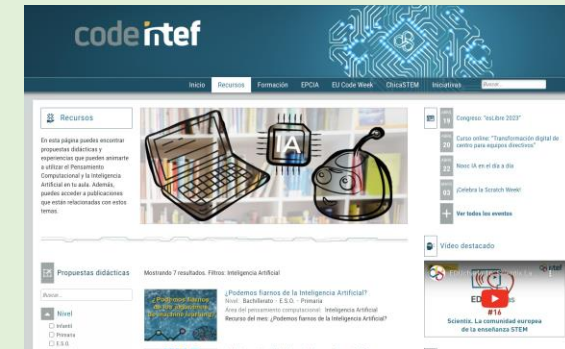
# Más sitios con actividades de IA

## Inteligencia Artificial

Algunas publicaciones y propuestas didácticas para introducir la Inteligencia Artificial en el aula.



- Participación en «Learning, Training & Teaching Activity (LTTA) FAIaS Braga · Fomentando la Inteligencia Artificial en las Escuelas»
- Propuesta didáctica: Inteligencia artificial con LearningML. Modelo numérico. Matemáticas; puntos, coordenadas y cuadrantes



# Más sitios con actividades de IA

**LearningML - AI made easy**

QUÉ ES POR QUÉ APRENDER DESARROLLO BLOG DESCARGAS ACERCA DE CONTACTO

Manual de LearningML, Videotutoriales, Recursos para aprender IA y ML, Actividades guías con LearningML.

**LearningML v1.2 (la original)**  
Ideal para introducirse en el mundo del Machine Learning. Se puede usar desde los últimos cursos de primaria.

**LearningML v1.3 (avanzado)**  
En esta versión se añade la clasificación de conjuntos de números y el modo avanzado, con el que podrás explorar el comportamiento de los algoritmos de ML.

**LearningML Desktop**  
Para los que prefieren tener instalado LearningML v1.3 en su ordenador (Linux, Windows, Mac) y pasar de Internet. Ideal para ser incorporado en distribuciones educativas de Linux y para colegios que tengan problemas de conexión a Internet.

**Recopila datos**  
Recopila textos o imágenes sobre algo que quieras clasificar de forma automática y añádelos a LearningML indicando a qué clase pertenece cada uno de ellos. Estos datos constituyen el conjunto de entrenamiento.

**Creas un modelo**  
Construye con LearningML un modelo capaz de clasificar correctamente otros datos distintos, aunque similares, a los del conjunto de entrenamiento.

**Construye una aplicación**  
Exporta tu modelo de Machine Learning a Scratch y programa una aplicación con capacidad para clasificar datos sobre el tema que hayas elegido. ¡EchidnaBurr ha incorporado Inteligencia Artificial a tu programa Scratch!

Machine learning IA en Educación Primaria

## Aplicación Turística

2. Aprender  
Llévate el momento de aprender a clasificar imágenes.  
Aprender a reconocer imágenes

3. Probar  
Introduce imágenes nuevas y comprueba si se clasifican correctamente.

CC BY SA



## ASISTENTE VIRTUAL

Con la posibilidad de combinar bloques de control de Echidna con de Machine Learning gracias al LearningML es muy sencillo incluir Inteligencia Artificial en nuestros proyectos. A continuación se

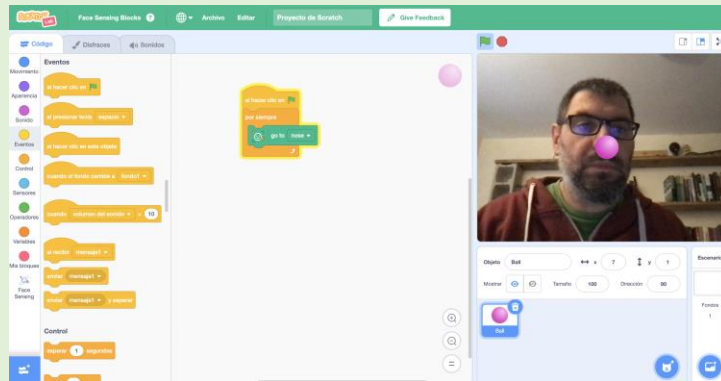
## IA

Aunque no se haya presentado formalmente, una de las grandes novedades que Echidna Educación propone para este nuevo curso es **EchidnaScratch** (que podéis encontrar en la sección **A programar de la web**), un entorno de programación por bloques para controlar las placas Echidna creado por Juan David Rodríguez.

Para ponerlos un poco en contexto, Juan David, uno de los nuevos miembros de Echidna Educación, es el desarrollador de un fork de Scratch que incluye bloques de Machine Learning.

# Herramientas para aprender/enseñar IA a través de aplicaciones prácticas

Scratch Face



Machine Learning for Kids

Enseñale a una computadora a jugar un juego

1. Agrega ejemplos de cosas que quieres que la computadora reconozca/identifique automáticamente
2. Usa los ejemplos para entrenar una computadora y así aprende a reconocer este tipo de objetos (texto/imágenes/números) automáticamente
3. Crea un juego en Scratch que use la habilidad de la computadora para reconocerlos

Empezar    Aprender más

Teachable Machine

Teachable Machine

Preparación

Modelo preparado

Avanzado

Épocas: 50

Tamaño del lote: 16

Tasa de aprendizaje: 0.001

Restablecer valores predeterminados

Más datos

Vista previa

Exportar modelo

Selecciona imágenes de tus archivos o arrástralas aquí

Importar imágenes desde Google Drive

Salida

avión

marip...

Cognimates

COGNIMATES

Get Started    Projects    Tutorials    Events    Research    Blog    Team

An AI education platform for building games, programming robots & training AI models

Train Models    Code & Play

# Herramientas para aprender/enseñar IA

<https://quickdraw.withgoogle.com/>, es una aplicación que usa la IA para reconocer los dibujos que el usuario hace. Utiliza redes neuronales artificiales, uno de los algoritmos cuyas bases aprenderemos en el curso, para funcionar

<https://www.scroobly.com/>, se presenta una aplicación que usa inteligencia artificial para reconocer partes del cuerpo y hacer que un muñecote imite los gestos que el usuario hace frente a la cámara

URL <https://experiments.withgoogle.com/ai-duet>, se presenta una aplicación en la que el usuario puede tocar un piano y el ordenador responde de manera coherente y armónica tocando también el piano

<https://andymatuschak.org/scrying-pen/>, te ayuda a dibujar un objeto a medida que vas trazándolo. Primero escoges el modelo, es decir, lo que pretendes dibujar. Y después comienzas a dibujar. La aplicación te va presentando trazos posibles que puedes seguir para completar el dibujo.

# Herramientas para aprender/enseñar IA

Uno de los momentos estelares de la inteligencia artificial ocurrió en 1966, cuando se creó el primer bot conversacional. Se llamaba Eliza y fue la primera aplicación que permitía establecer una conversación “coherente” con una computadora. Emula a un psicólogo que sigue los principios de la terapia centrada en el cliente (terapia rogeriana). En realidad, es una parodia a esta terapia. En el artículo <https://www.bbc.com/mundo/noticias-44290222>, te lo cuentan mejor. Lo importante es que se consiguió programar una computadora para mantener una conversación con un humano. Lo bueno es que muchas personas, al usar la aplicación, pensaron que realmente hablaban con un psicólogo. Así que, por lo menos, las frases estaban bien construidas y algo tenían que ver con las respuestas del humano. Y esto no deja de ser un magnífico logro en el campo del procesamiento del lenguaje natural (NLP, Natural Language Processing). Ahora puedes probar tú mismo este bot usando alguna de las muchas reimplementaciones que existen en la web. Esta versión, <https://www.masswerk.at/elizabot/>, es fiel a la original, por lo que hay que escribir en inglés. Pero puedes probar también esta versión en español, <http://deixilabs.com/eliza.html>. ¿Sería capaz de hacerte creer que hablas con un psicólogo?



# Herramientas para aprender/enseñar IA

<https://es.akinator.com/> Un juego realmente divertido y sorprendente. Piensa un personaje. La aplicación te irá haciendo preguntas sobre él y al cabo de unas cuantas lo descubre. Puedes llegar a pensar en personajes muy rebuscados, reales o de ficción. Aunque en la web no dicen nada acerca de su funcionamiento, lo más probable es que haga uso de una técnica llamada árboles de decisión.

<https://freddiemeter.withyoutube.com/>. ¿Te atreves a cantar una canción de Freddy Mercury? Esta aplicación te dirá hasta qué punto te pareces a él cantando.

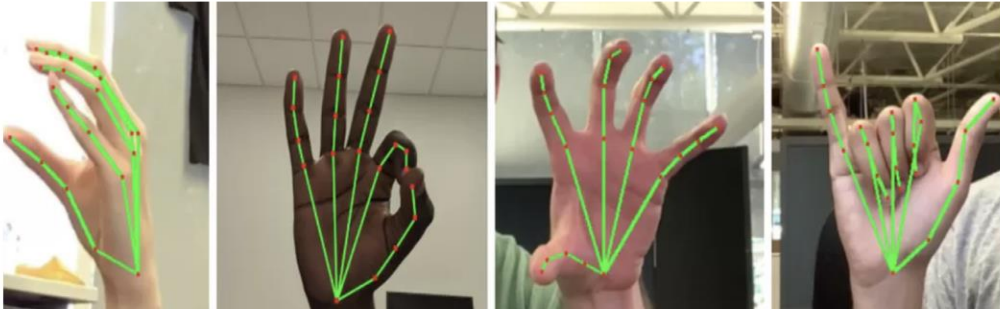
Esta aplicación es realmente divertida, <https://www.hownormalami.eu/>. Simplemente activa la cámara y deja que la aplicación vaya asignándote puntos sobre tu “belleza”, predicciones sobre tu edad, género, esperanza de vida, etcétera. A medida que va extrayendo esta información a partir de tu cara (o por la cara, no sé 😊), en un video te va explicando cómo lo hace. Puedes saltarte esta explicación haciendo llevando el puntero del ratón al video y haciendo clic en el botón “SKIP”.

Importante: No te tomes muy en serio los resultados

# Machine learning para intentar traducir el lenguaje de signos en tiempo real

## Google sign language AI turns hand gestures into speech

© 20 August 2019

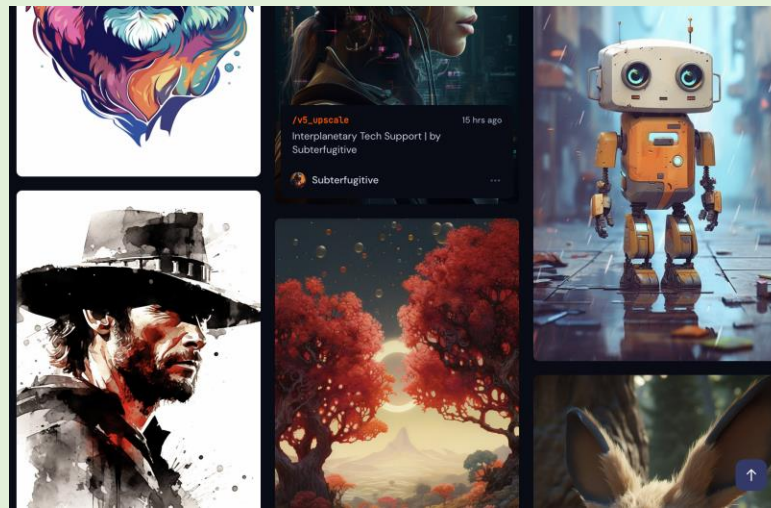
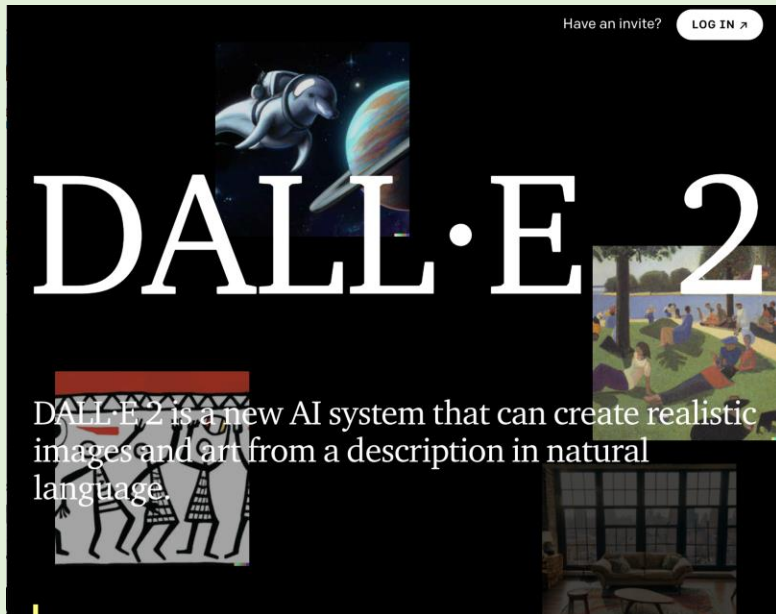


# Un gui3n escrito por un algoritmo



# Generación de imágenes a partir de textos

## Midjourney



## Stable Diffusion Launch Announcement



Stability AI and our collaborators are proud to announce the first stage of

Esta persona no existe



# Fake videos




VI XORNADAS MONOGRÁFICAS

APETEGA

Asociación de Profesorado de Tecnología de Galicia



¡GRACIAS!

 @juandalibaba

<https://learningml.org>

