



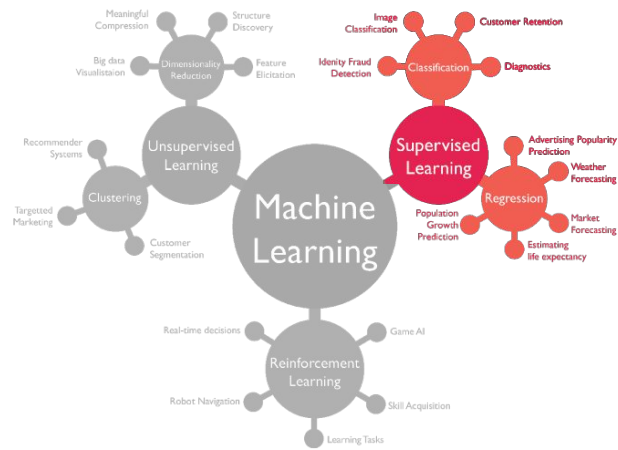
DEPARTAMENTO  
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

# Aprendizaje Automático

2024-1C

## Clase 1: Aprendizaje Supervisado



# Definición:

“Problemas de aprendizaje bien planteados” (Mitchell 1.1)

Un programa se dice que aprende a partir de la **experiencia E** con respecto a una **tarea T** y una medida de **performance P**, si su performance en la **tarea T**, medida **mediante P**, mejora con la **experiencia E**.

## Ejemplos:

**T:** Reconocimiento del habla.

**P:** Distancia de edición entre transcripción real y predicha en llamados reales.

**E:** Una base de datos de conversaciones entre agentes y clientes junto a su transcripción.

**T:** Manejo autónomo en calles reales.

**P:** Distancia recorrida hasta el primer error.

**E:** Sensores en el volante, pedales y cámaras mientras maneja un humano.

**T:** Jugar al Ajedrez.

**P:** Porcentaje de juegos ganados.

**E:** Jugar contra sí mismo.

¿Qué opinan de esta P?

# Definición:

“Problemas de aprendizaje bien planteados” (Mitchell 1.1)

Un programa se dice que aprende a partir de la **experiencia E** con respecto a una **tarea T** y una medida de **performance P**, si su performance en la **tarea T**, medida **mediante P**, mejora con la **experiencia E**.

## Ejemplos:

**T:** Reconocimiento del habla.

**P:** Distancia de edición entre transcripción real y predicha en llamados reales.

**E:** Una base de datos de conversaciones entre agentes y clientes junto a su transcripción.

**T:** Manejo autónomo en calles reales.

**P:** Distancia recorrida hasta el primer error.

**E:** Sensores en el volante, pedales y cámaras mientras maneja un humano.

**T:** Jugar al Ajedrez.

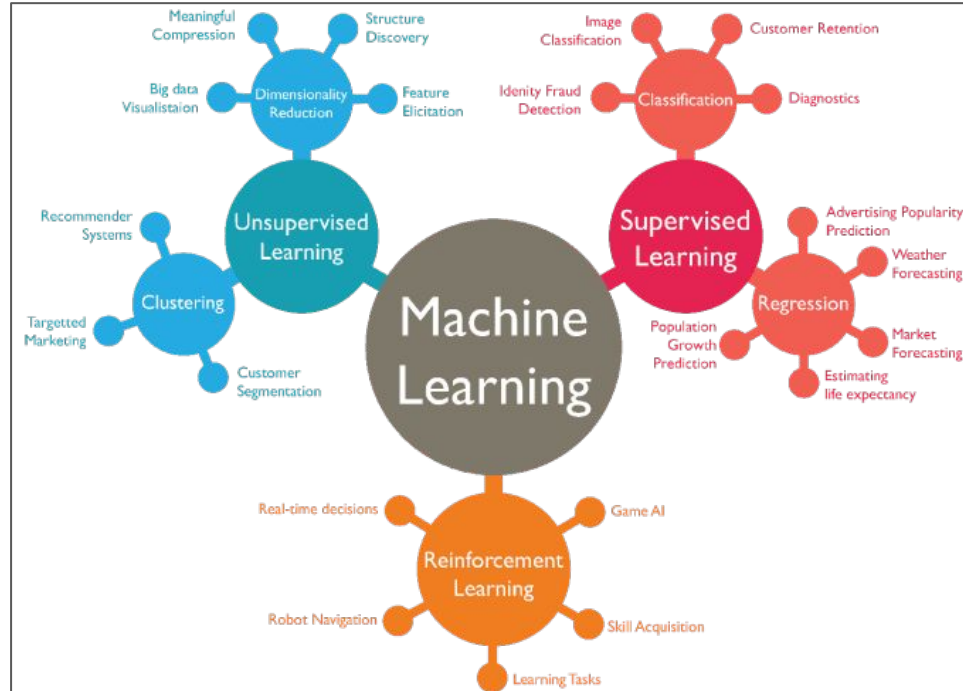
**P:** Porcentaje de juegos ganados online contra competidores humanos.

**E:** Jugar contra sí mismo.

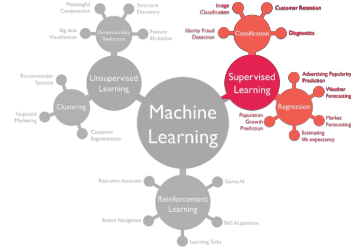
# Tipos de Aprendizaje Automático

# Aprendizaje Automático

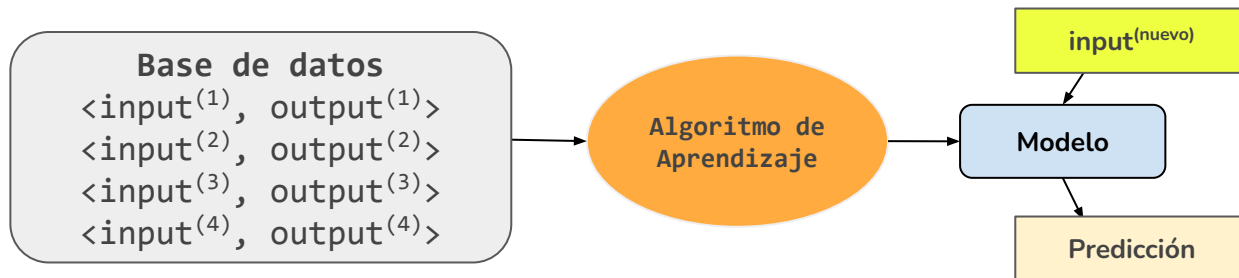
Aprendizaje **Supervisado**; Aprendizaje **No Supervisado**; Aprendizaje **Por Refuerzos**



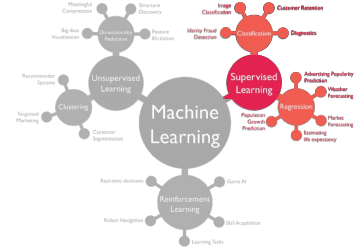
# Aprendizaje Supervisado



- Dados una serie de pares  $\{\text{input}^{(i)}, \text{output}^{(i)}\}^n$
- Se construye un **modelo** que permita crear un output a partir de un input que **nunca vio antes** sin la ayuda de decisiones hardcodeadas por humanos.



- Un algoritmo "aprende" un **mapeo** que relaciona **input** → **output**.
- ¿Cómo? Siguiendo **patrones** a partir de los ejemplos vistos.

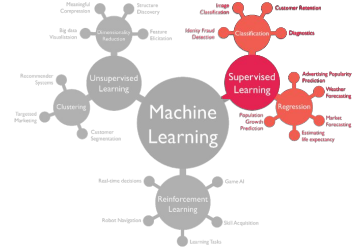


# Aprendizaje **Supervisado**: Ejemplos

- Identificar los dígitos escritos a mano en un sobre del correo.
  - DB:  $(\text{carta}^{(1)}, "4223")$ ,  $(\text{carta}^{(2)}, "1423")$ , ...,  $(\text{carta}^{(N)}, "414")$
- Determinar si un tumor es benigno basado en una tomografía.
  - DB:  $(\text{imagen}^{(1)}, "benigno")$ ,  $(\text{imagen}^{(2)}, "benigno")$ , ...,  $(\text{imagen}^{(N)}, "maligno")$
- Detectar actividad fraudulenta en una tarjeta de crédito.
  - DB:  $(\text{actividad}^{(1)}, "fraude")$ ,  $(\text{actividad}^{(2)}, "legal")$ , ...,  $(\text{actividad}^{(N)}, "legal")$

¿Cómo **represento** la carta? ¿La imagen? ¿La actividad?

¿Cómo **consigo** los datos? ¿Es difícil? ¿Es caro? ¿Es ético?



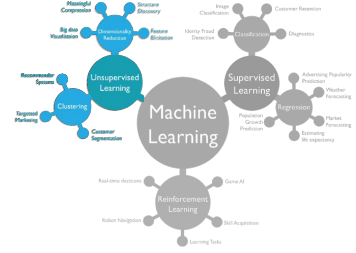
# Aprendizaje **Supervisado**: Ejemplos

- Identificar los dígitos escritos a mano en un sobre del correo.
  - DB:  $(\text{carta}^{(1)}, "4223")$ ,  $(\text{carta}^{(2)}, "1423")$ , ...,  $(\text{carta}^{(N)}, "414")$
- Determinar si un tumor es benigno basado en una tomografía.
  - DB:  $(\text{imagen}^{(1)}, "benigno")$ ,  $(\text{imagen}^{(2)}, "benigno")$ , ...,  $(\text{imagen}^{(N)}, "maligno")$
- Detectar actividad fraudulenta en una tarjeta de crédito.
  - DB:  $(\text{actividad}^{(1)}, "fraude")$ ,  $(\text{actividad}^{(2)}, "legal")$ , ...,  $(\text{actividad}^{(N)}, "legal")$

¿Cómo **represento** la carta? ¿La imagen? ¿La actividad?

¿Cómo **consigo** los datos? ¿Es difícil? ¿Es caro? ¿Es ético?

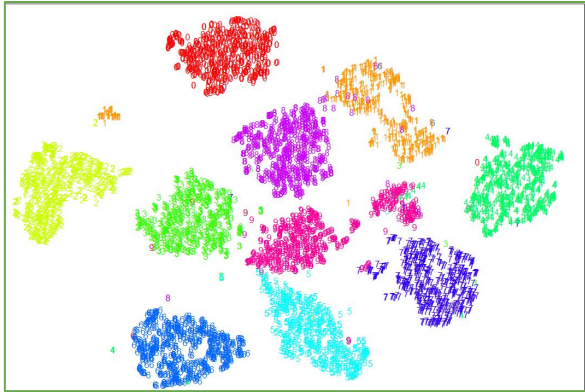




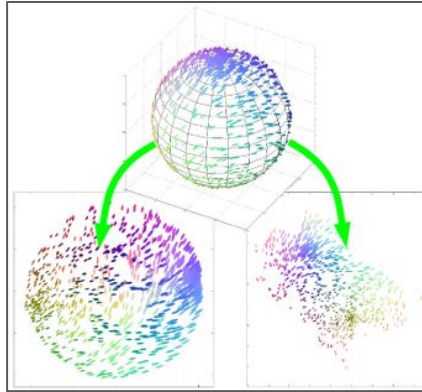
# Aprendizaje No Supervisado

En aprendizaje **no supervisado**, sólo el input es conocido. Lo que se quiere descubrir no está visible (o al menos no tengo datos previos con etiquetas).

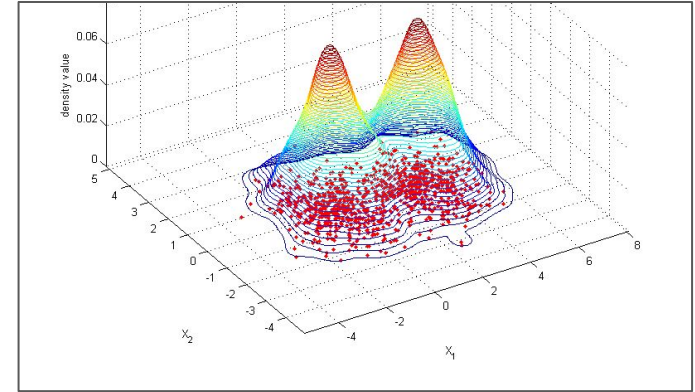
# Clustering

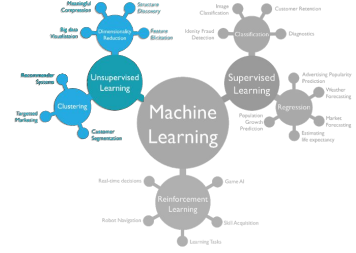


# Reducción dimensional



## Estimación de densidades

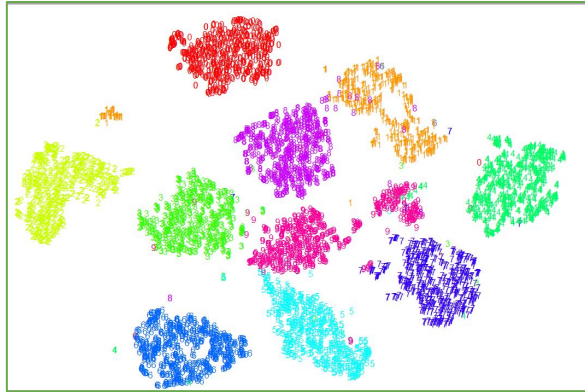




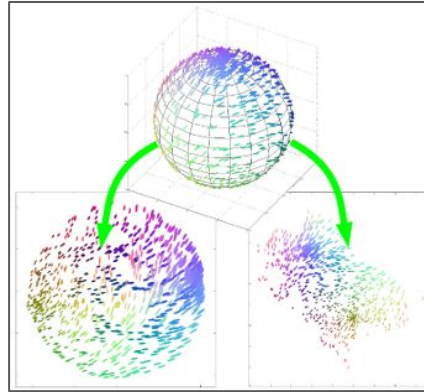
# Aprendizaje No Supervisado

En aprendizaje **no supervisado**, sólo el input es conocido. Lo que se quiere descubrir no está visible (o al menos no tengo datos previos con etiquetas).

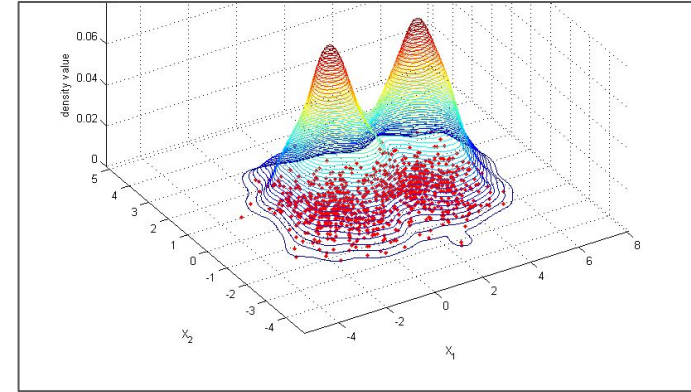
Clustering



Reducción dimensional

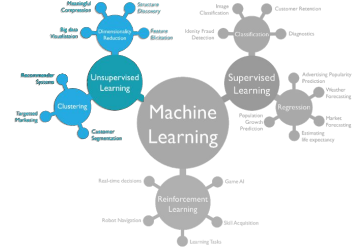


Estimación de densidades



Hay casos concretos de éxito de estas técnicas, pero suelen ser más **difíciles** de **evaluar**.

# Aprendizaje **No Supervisado**: Ejemplos

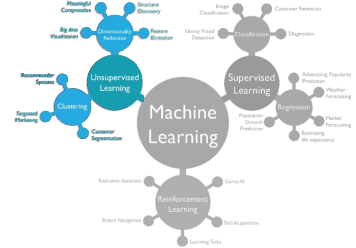


- Identificar **tópicos** presentes en un blog.
  - DB:  $\text{texto}^{(1)}$ ,  $\text{texto}^{(2)}$ , ...,  $\text{texto}^{(N)}$
- **Segmentar** clientes según preferencias en Netflix.
  - DB:  $\text{preferencias}^{(1)}$ ,  $\text{preferencias}^{(2)}$ , ...,  $\text{preferencias}^{(N)}$
- **Detectar** ataques a sitios web.
  - DB:  $\text{actividad}^{(1)}$ ,  $\text{actividad}^{(2)}$ , ...,  $\text{actividad}^{(N)}$
- **Visualizar** datos de alta dimensión (ej, canciones según que tan parecidas son)
  - DB:  $\text{canción}^{(1)}$ ,  $\text{canción}^{(2)}$ , ...,  $\text{canción}^{(N)}$

¿Cómo sé cuántos tópicos hay? ¿Qué es un tópico? ¿Cuántos segmentos hay?

¿Cómo detectar un ataque si nunca antes sucedió? ¿Cómo visualizar datos 700-D?

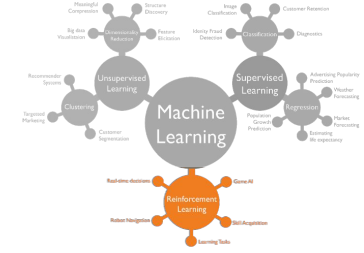
# Aprendizaje **No Supervisado**: Ejemplos



- Identificar **tópicos** presentes en un blog.
  - DB:  $\text{texto}^{(1)}$ ,  $\text{texto}^{(2)}$ , ...,  $\text{texto}^{(N)}$
- **Segmentar** clientes según preferencias en Netflix.
  - DB:  $\text{preferencias}^{(1)}$ ,  $\text{preferencias}^{(2)}$ , ...,  $\text{preferencias}^{(N)}$
- **Detectar** ataques a sitios web.
  - DB:  $\text{actividad}^{(1)}$ ,  $\text{actividad}^{(2)}$ , ...,  $\text{actividad}^{(N)}$
- **Visualizar** datos de alta dimensión (ej, canciones según que tan parecidas son)
  - DB:  $\text{canción}^{(1)}$ ,  $\text{canción}^{(2)}$ , ...,  $\text{canción}^{(N)}$

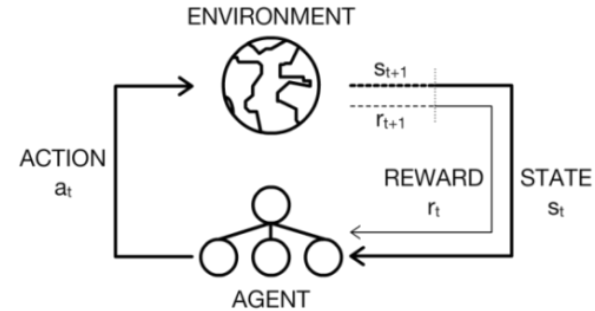
¿Cómo sé cuántos tópicos hay? ¿Qué es un tópico? ¿Cuántos segmentos hay?

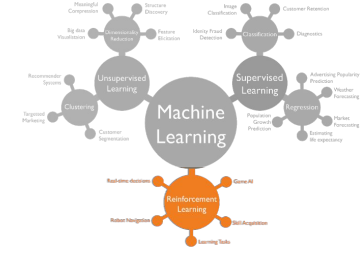
¿Cómo detectar un ataque si nunca antes sucedió? ¿Cómo visualizar datos 700-D?



# Aprendizaje por **Refuerzos**

- Agentes que aprenden interactuando con el **ambiente**.
  - El ambiente provee **recompensas**
  - Se busca una **política de acciones** que maximicen las recompensas
- 
- Robótica
  - Juegos
  - Simulaciones
  
  - ChatGPT!

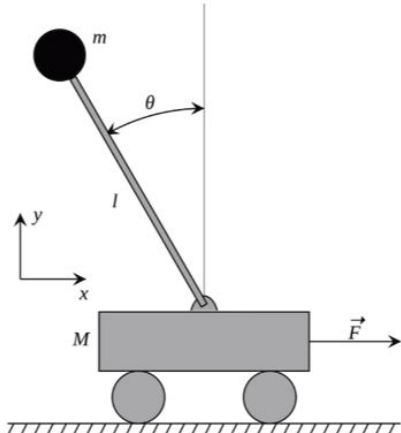




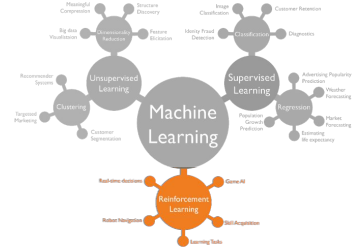
# Aprendizaje por **Refuerzos**: Ejemplo

- **Objetivo:** Balancear un péndulo sobre un carrito en movimiento
- **Estado:** <Ángulo, Posición, Velocidad horizontal>
- **Acción:** Aplicación de fuerza sobre el carrito
- **Recompensa:** 1 punto por cada momento en que el péndulo se mantuvo sin tocar el carrito.

[https://pytorch.org/tutorials/intermediate/reinforcement\\_q\\_learning.html](https://pytorch.org/tutorials/intermediate/reinforcement_q_learning.html)

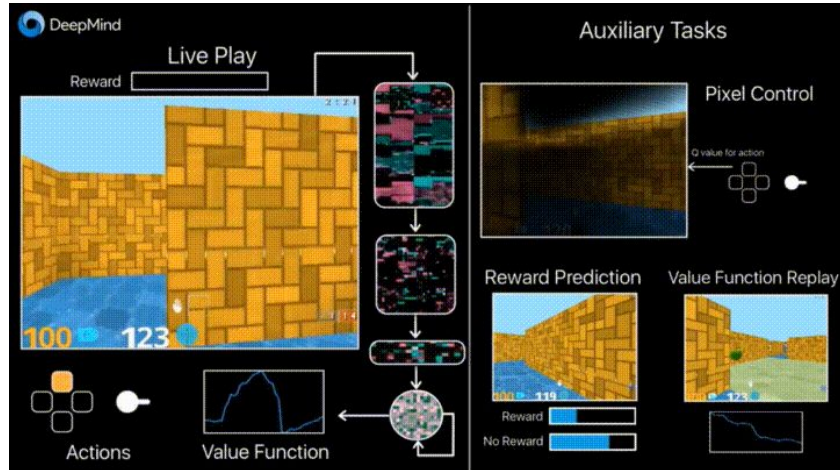


# Aprendizaje por **Refuerzos**: Ejemplo

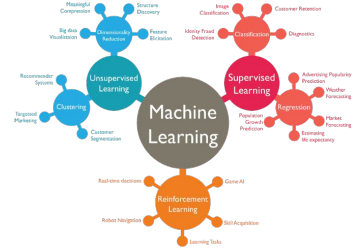


Ejemplos recomendadísimos

[https://www.youtube.com/results?search\\_query=two+minute+papers+reinforcement](https://www.youtube.com/results?search_query=two+minute+papers+reinforcement)



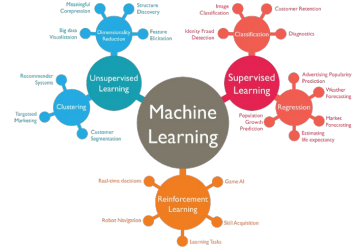
# ANTES QUE NADA



- ¿Qué **pregunta** estoy contestando? ¿En qué tipo de aprendizaje cae mi problema?
- ¿Los datos, tienen **información suficiente**? ¿Son suficientes?
- ¿Cómo voy a **medir** el éxito de mi aplicación?
- ¿Cómo va a **interactuar** con el resto de mi producto?



# ANTES QUE NADA



- ¿Qué **pregunta** estoy contestando? ¿En qué tipo de aprendizaje cae mi problema?
- ¿Los datos, tienen **información suficiente**? ¿Son suficientes?
- ¿Cómo voy a **medir** el éxito de mi aplicación?
- ¿Cómo va a **interactuar** con el resto de mi producto?

Mucha gente gasta tiempo creando modelos complicados de machine learning y luego descubren que **no estaban resolviendo el problema correcto**.

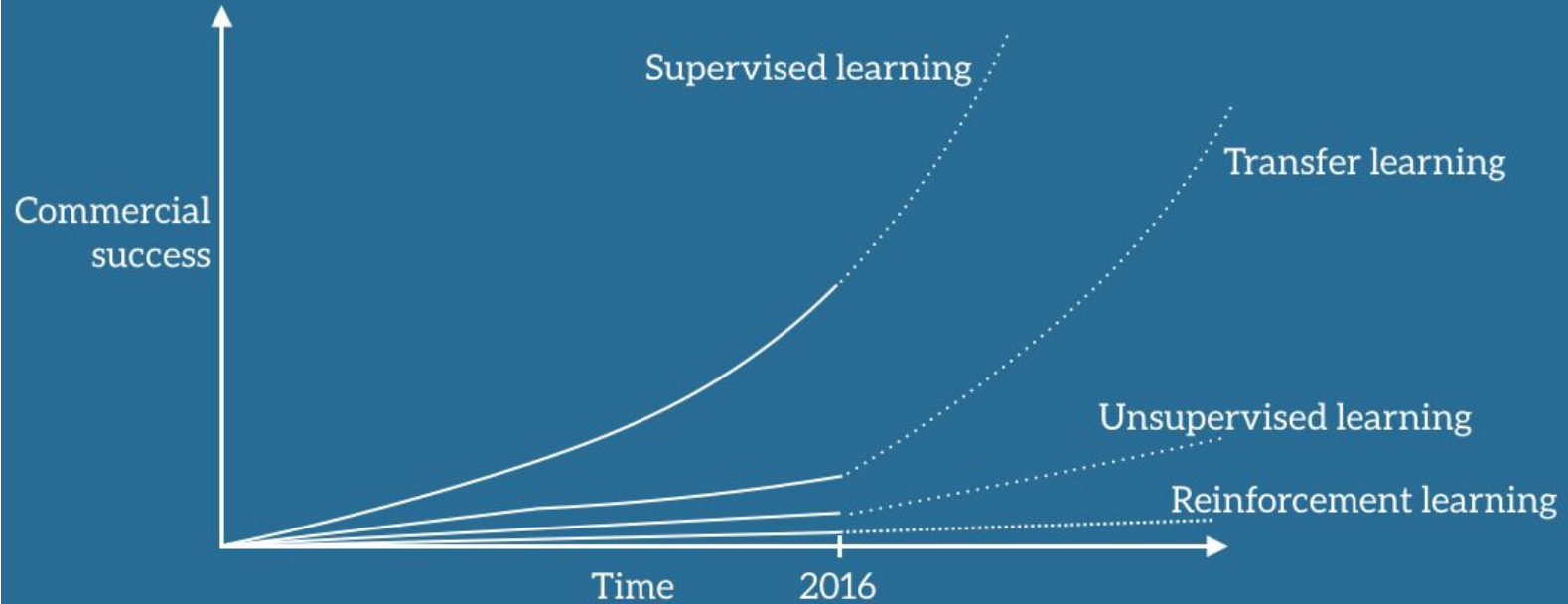
# Otros tipos de tareas de ML

- Generative learning (chat-gpt, stable-diffusion, etc).
- Semi-supervised learning
- Multi-task learning (multilabel)
- Transfer learning
- Structured output prediction
- Domain adaptation
- One shot learning
- Zero shot learning
- Muchas más

# Ejemplos de uso de ML

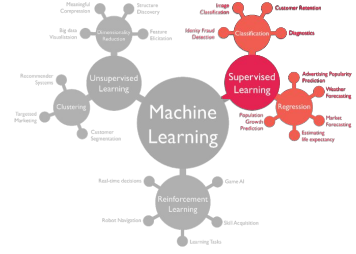
- Clasificación de mensajes (ej: spam):
  - GMail, Hotmail, Yahoo!, etc.
- Reconocimiento del habla:
  - Siri, Cortana, Google Now, Amazon Echo, ASAPP, Dialpad, etc.
- Sistemas de recomendación
  - Spotify, Facebook, Netflix, Amazon, Mercado Libre, Google, etc.
- Predicción de tiempo de viaje, camino óptimo:
  - Waze, Uber, Google Maps, Despegar, etc.
- Detección de fraude:
  - PayPal, Mercadolibre, bancos, etc.
- Publicidad online:
  - Google Ads, Jampp, etc.

# Drivers of ML success in industry



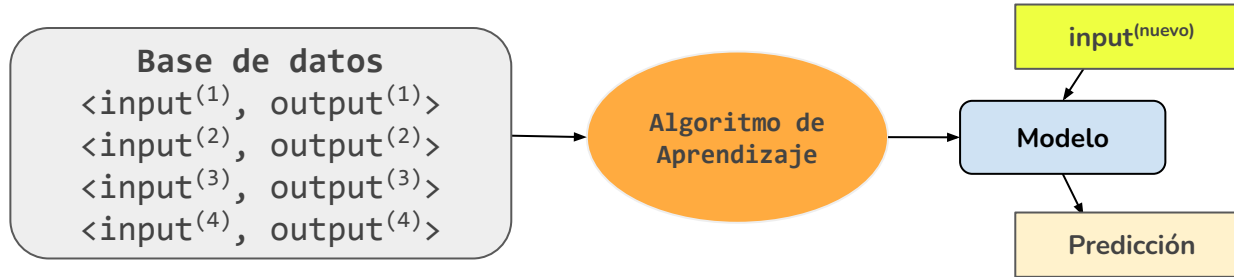
- Andrew Ng, NIPS 2016 tutorial

# Aprendizaje Supervisado



# Aprendizaje Supervisado

- Dados una serie de pares  $\{\text{input}^{(i)}, \text{output}^{(i)}\}^n$
- Se construye un **modelo** que permita crear un output a partir de un input que **nunca vio antes** sin la ayuda de decisiones hardcodeadas por humanos.



- Un algoritmo "aprende" un **mapeo** que relaciona **input** → **output**.
- ¿Cómo? Siguiendo **patrones** a partir de los ejemplos vistos.

# Aprendizaje Supervisado

## Objetivo

Estamos interesados en modelar un **proceso particular**; queremos aprender o aproximar una **función que no conocemos**, pero de la cual **tenemos ejemplos**.

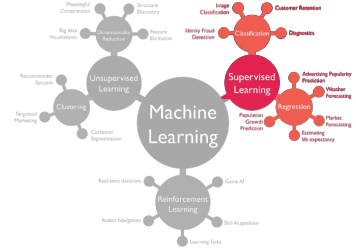
## Buscando:

- Predicciones acertadas para datos nuevos.
- Conocer la confianza de nuestro sistema al predecir.
- Entender qué partes del input afectan al output y cómo (problema de inferencia).

**Generalmente requiere** esfuerzo humano para construir el conjunto de entrenamiento, pero **provee** automatización y aceleración de trabajo laborioso o impracticable.

Importante entender las ideas detrás de **varias técnicas** para saber cuándo usarlas y cómo usarlas.

Aprender **las bases conceptuales** les permitirá poder entender técnicas modernas y poder aplicar y/o investigar en el tema.



# ¿Aprendizaje Automático vs Aprendizaje Estadístico?

El **aprendizaje automático** surgió como un subcampo de la **Inteligencia Artificial**.

El **aprendizaje estadístico** surgió como un subcampo de la **Estadística**.

Hay mucho solapamiento: ambos campos se centran en problemas tanto supervisados como no supervisados:

- El **aprendizaje automático** tiene un mayor énfasis en **aplicaciones a gran escala** y **precisión de predicción**.
- El **aprendizaje estadístico** enfatiza **modelos y su interpretabilidad**, precisión e incertidumbre.

Pero la distinción se ha vuelto cada vez más difusa y hay una gran cantidad de "cruzamientos".

El **aprendizaje automático** tiene la ventaja en Marketing

**En este curso**, las técnicas que veremos pertenecen a los dos mundos por igual y abre las puertas a aplicaciones tanto científicas como ingenieriles.



# Aprendizaje Supervisado

## Clasificación vs Regresión

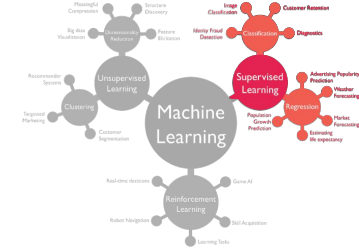
# Clasificación vs. Regresión

## Clasificación

- Se busca predecir un valor perteneciente a un conjunto finito, sin orden. "Etiquetas"
- **Ejs:** ¿Spam? ¿idioma de un sitio web? ¿Tumor maligno? ¿Cambiará de compañía telefónica? ¿Dígito nombrado?

## Regresión

- Busca predecir un valor cuantitativo (enteros, reales, etc).
- Hay continuidad en las posibles salidas.
- **Ejs:** ¿Sueldo de una persona? ¿Gasto en publicidad para el año siguiente? ¿Temperatura del viernes?



### Classification

¿Soleado, nublado o lluvioso?



### Regression

Temperatura mañana?



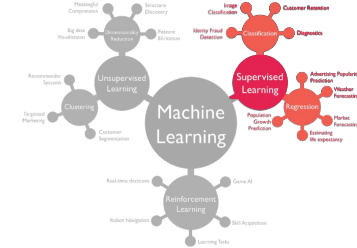
# Clasificación vs. Regresión

## Clasificación

- Se busca predecir un valor perteneciente a un conjunto finito, sin orden. "Etiquetas"
- **Ejs:** ¿Spam? ¿idioma de un sitio web? ¿Tumor maligno? ¿Cambiará de compañía telefónica? ¿Dígito nombrado?

## Regresión

- Busca predecir un valor cuantitativo (enteros, reales, etc).
- Hay continuidad en las posibles salidas.
- **Ejs:** ¿Sueldo de una persona? ¿Gasto en publicidad para el año siguiente? ¿Temperatura del viernes?



### Classification

¿Soleado, nublado o lluvioso?



### Regression

¿Temperatura media mañana?



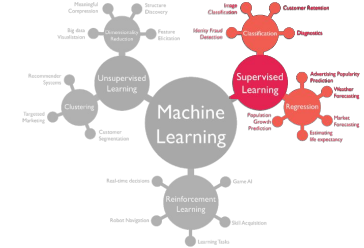
# Clasificación vs. Regresión

## Clasificación

- Se busca predecir un valor perteneciente a un conjunto finito, sin orden. "Etiquetas"
- **Ejs:** ¿Spam? ¿idioma de un sitio web? ¿Tumor maligno? ¿Cambiará de compañía telefónica? ¿Dígito nombrado?

## Regresión

- Busca predecir un valor cuantitativo (enteros, reales, etc).
- Hay continuidad en las posibles salidas.
- **Ejs:** ¿Sueldo de una persona? ¿Gasto en publicidad para el año siguiente? ¿Temperatura del viernes?



### Classification

¿Soleado, nublado o lluvioso?



### Regression

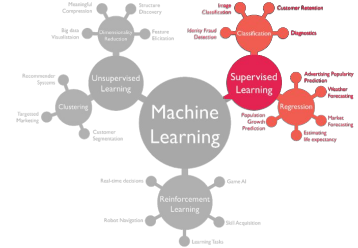
¿Temperatura media mañana?



## ¿Cuándo un problema es de un tipo o el otro?

Pensar en el tipo de datos de los ejemplos de las etiquetas de entrenamiento: **output<sup>(i)</sup> :: ?**

# Clasificación vs. Regresión



## Clasificación

$\text{output}^{(i)} :: \text{Bool} \mid \text{Enum}$

- Se busca predecir un valor perteneciente a un conjunto finito, sin orden. "Etiquetas"
- Ejs:** ¿Spam? ¿idioma de un sitio web? ¿Tumor maligno? ¿Cambiará de compañía telefónica? ¿Dígito nombrado?



## Classification

¿Soleado, nublado o lluvioso?



## Regresión

$\text{output}^{(i)} :: \mathbb{N} \mid \mathbb{Z} \mid \mathbb{R}$

- Busca predecir un valor cuantitativo (enteros, reales, etc).
- Hay continuidad en las posibles salidas.
- Ejs:** ¿Sueldo de una persona? ¿Gasto en publicidad para el año siguiente? ¿Temperatura del viernes?



## Regression

¿Temperatura media mañana?



## ¿Cuándo un problema es de un tipo o el otro?

Pensar en el tipo de datos de los ejemplos de las etiquetas de entrenamiento:  $\text{output}^{(i)} :: ?$

# Clasificación vs. Regresión

## Clasificación

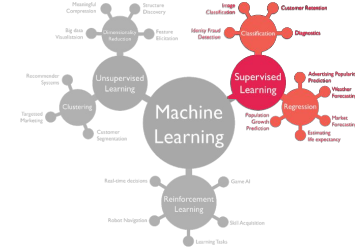
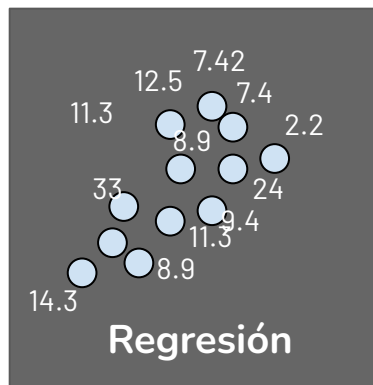
$\text{output}^{(i)} :: \text{Bool} \mid \text{Enum}$

- Se busca predecir un valor perteneciente a un conjunto finito, sin orden. "Etiquetas"
- Ejs:** ¿Spam? ¿idioma de un sitio web? ¿Tumor maligno? ¿Cambiará de compañía telefónica? ¿Dígito nombrado?

## Regresión

$\text{output}^{(i)} :: \mathbb{N} \mid \mathbb{Z} \mid \mathbb{R}$

- Busca predecir un valor cuantitativo (enteros, reales, etc).
- Hay continuidad en las posibles salidas.
- Ejs:** ¿Sueldo de una persona? ¿Gasto en publicidad para el año siguiente? ¿Temperatura del viernes?



**¿Cuándo un problema es de un tipo o el otro?**

Pensar en el tipo de datos de los ejemplos de las etiquetas de entrenamiento:  $\text{output}^{(i)} :: ?$



# Aprendizaje Inductivo



# Aprendizaje Inductivo

Guiemos la sección con un ejemplo: ¿Compró un bote?.

En este problema, estamos interesados en **predecir** si una persona compró, o no, un bote en base a ciertas características (**atributos**) consultadas mediante una encuesta telefónica:

- **Edad** :: Int {1, ..., 100}
- **#autos** :: Int {0, ..., 5}
- **¿Casa propia?** :: Bool
- **#hijos** :: Int {0, ..., 5}
- **estado civil** :: Enum {viudo/a, soltero/a, casado/a, divorciado/a, NA}
- **¿Tiene perros?** :: Bool

Edad	#autos	¿Casa propia?	#hijos	Estado civil	¿Tiene perros?
66	1	sí	2	viudo/a	no
52	2	sí	3	casado/a	no
22	0	no	0	casado/a	sí
25	1	no	1	soltero/a	no

Queremos pensar reglas que permitan predecir si alguien **compró, o no, un bote**.

# Aprendizaje Inductivo

Edad	#autos	¿Casa propia?	#hijos	Estado civil	¿Tiene perros?	¿Compró un bote?
66	1	sí	2	viudo/a	no	<i>sí</i>
52	2	sí	3	casado/a	no	<i>sí</i>
22	0	no	0	casado/a	sí	<i>no</i>
25	1	no	1	soltero/a	no	<i>no</i>
44	0	no	2	divorciado/a	sí	<i>no</i>
39	1	sí	2	casado/a	no	<i>no</i>
26	1	no	2	soltero/a	sí	<i>no</i>
40	3	sí	1	casado/a	no	<i>no</i>
53	2	sí	2	divorciado/a	no	<i>sí</i>
64	2	sí	3	divorciado/a	sí	<i>no</i>
58	2	sí	2	casado/a	sí	<i>sí</i>
33	1	no	1	soltero/a	no	<i>no</i>

**Ejemplo de Hipótesis:** Si la persona tiene 2 autos entonces querrá comprar un bote. Si no, no.

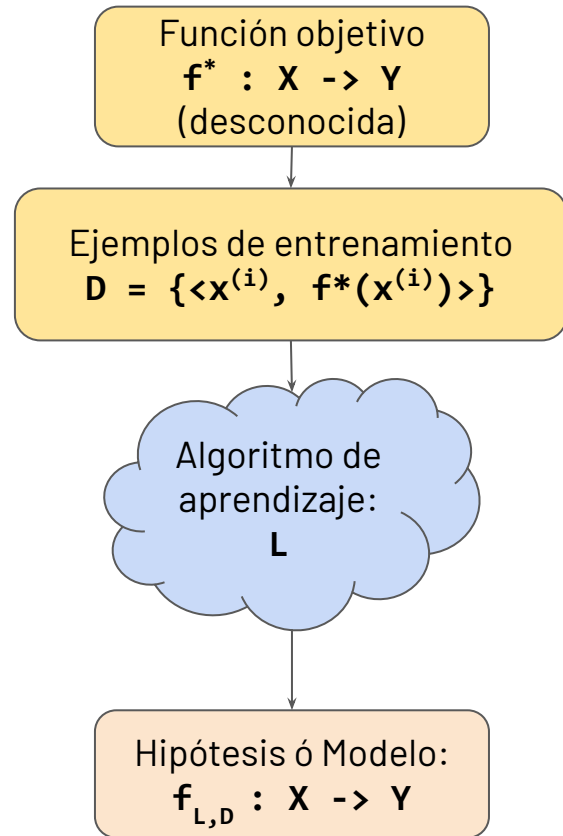
# Función objetivo, hipótesis, modelo.

**Función objetivo:** Estamos interesados en modelar un **proceso particular**; queremos aprender o aproximar una **función que no conocemos**, pero de la cual **tenemos ejemplos**.

**Hipótesis:** Una cierta **función que creemos** (o esperamos que sea) similar a la función objetivo que queremos modelar.

**Modelo:** En el campo del aprendizaje automático, los términos hipótesis y modelo a menudo se usan indistintamente.

OjO: a veces “modelo” también se utiliza para referirse al algoritmo de aprendizaje. Trataremos de evitar esta confusión.



# Aprendizaje Inductivo

Edad	#autos	¿Casa propia?	#hijos	Estado civil	¿Tiene perros?	¿Compró un bote?
66	1	sí	2	viudo/a	no	<i>sí</i>
52	2	sí	3	casado/a	no	<i>sí</i>
22	0	no	0	casado/a	sí	<i>no</i>
25	1	no	1	soltero/a	no	<i>no</i>
44	0	no	2	divorciado/a	sí	<i>no</i>
39	1	sí	2	casado/a	no	<i>no</i>
26	1	no	2	soltero/a	sí	<i>no</i>
40	3	sí	1	casado/a	no	<i>no</i>
53	2	sí	2	divorciado/a	no	<i>sí</i>
64	2	sí	3	divorciado/a	sí	<i>no</i>
58	2	sí	2	casado/a	sí	<i>sí</i>
33	1	no	1	soltero/a	no	<i>no</i>

**Hipótesis 1:** Si el cliente tiene casa, entonces querrá comprar un bote. Si no, no.

**Accuracy** (exactitud): ?

# Aprendizaje Inductivo

Edad	#autos	¿Casa propia?	#hijos	Estado civil	¿Tiene perros?	¿Compró un bote?	
66	1	sí	2	viudo/a	no	sí	Pred=sí ✓
52	2	sí	3	casado/a	no	sí	Pred=sí ✓
22	0	no	0	casado/a	sí	no	Pred=no ✓
25	1	no	1	soltero/a	no	no	Pred=no ✓
44	0	no	2	divorciado/a	sí	no	Pred=no ✓
39	1	sí	2	casado/a	no	no	Pred=sí ✗
26	1	no	2	soltero/a	sí	no	Pred=no ✓
40	3	sí	1	casado/a	no	no	Pred=sí ✗
53	2	sí	2	divorciado/a	no	sí	Pred=sí ✓
64	2	sí	3	divorciado/a	sí	no	Pred=sí ✗
58	2	sí	2	casado/a	sí	sí	Pred=sí ✓
33	1	no	1	soltero/a	no	no	Pred=no ✓

**Hipótesis 1:** Si el cliente tiene casa, entonces querrá comprar un bote. Si no, no.

**Accuracy** (exactitud): **75%** (9/12)

# Aprendizaje Inductivo

Edad	#autos	¿Casa propia?	#hijos	Estado civil	¿Tiene perros?	¿Compró un bote?
66	1	sí	2	viudo/a	no	<i>sí</i>
52	2	sí	3	casado/a	no	<i>sí</i>
22	0	no	0	casado/a	sí	<i>no</i>
25	1	no	1	soltero/a	no	<i>no</i>
44	0	no	2	divorciado/a	sí	<i>no</i>
39	1	sí	2	casado/a	no	<i>no</i>
26	1	no	2	soltero/a	sí	<i>no</i>
40	3	sí	1	casado/a	no	<i>no</i>
53	2	sí	2	divorciado/a	no	<i>sí</i>
64	2	sí	3	divorciado/a	sí	<i>no</i>
58	2	sí	2	casado/a	sí	<i>sí</i>
33	1	no	1	soltero/a	no	<i>no</i>

**Hipótesis 2:** Si el cliente es mayor de 45 años y tiene menos de 3 chicos o no está divorciado, entonces querrá comprar un bote. Si no, no.

**Accuracy** (exactitud): ?

# Aprendizaje Inductivo

Edad	#autos	¿Casa propia?	#hijos	Estado civil	¿Tiene perros?	¿Compró un bote?	
66	1	sí	2	viudo/a	no	sí	Pred=sí ✓
52	2	sí	3	casado/a	no	sí	Pred=sí ✓
22	0	no	0	casado/a	sí	no	Pred=no ✓
25	1	no	1	soltero/a	no	no	Pred=no ✓
44	0	no	2	divorciado/a	sí	no	Pred=no ✓
39	1	sí	2	casado/a	no	no	Pred=no ✓
26	1	no	2	soltero/a	sí	no	Pred=no ✓
40	3	sí	1	casado/a	no	no	Pred=no ✓
53	2	sí	2	divorciado/a	no	sí	Pred=sí ✓
64	2	sí	3	divorciado/a	sí	no	Pred=no ✓
58	2	sí	2	casado/a	sí	sí	Pred=sí ✓
33	1	no	1	soltero/a	no	no	Pred=no ✓

**Hipótesis 2:** Si el cliente es mayor de 45 años y tiene menos de 3 chicos o no está divorciado, entonces querrá comprar un bote. Si no, no.

**Accuracy** (exactitud): **100%** (12/12)

# Aprendizaje Inductivo

Edad	#autos	¿Casa propia?	#hijos	Estado civil	¿Tiene perros?	¿Compró un bote?
66	1	sí	2	viudo/a	no	<i>sí</i>
52	2	sí	3	casado/a	no	<i>sí</i>
22	0	no	0	casado/a	sí	<i>no</i>
25	1	no	1	soltero/a	no	<i>no</i>
44	0	no	2	divorciado/a	sí	<i>no</i>
39	1	sí	2	casado/a	no	<i>no</i>
26	1	no	2	soltero/a	sí	<i>no</i>
40	3	sí	1	casado/a	no	<i>no</i>
53	2	sí	2	divorciado/a	no	<i>sí</i>
64	2	sí	3	divorciado/a	sí	<i>no</i>
58	2	sí	2	casado/a	sí	<i>sí</i>
33	1	no	1	soltero/a	no	<i>no</i>

**Hipótesis 3:** Si el cliente tiene 52, 53, 58 o 66 años, entonces querrá comprar un bote. Si no, no.

**Accuracy** (exactitud): ?



# Aprendizaje Inductivo

Edad	#autos	¿Casa propia?	#hijos	Estado civil	¿Tiene perros?	¿Compró un bote?	
66	1	sí	2	viudo/a	no	sí	Pred=sí ✓
52	2	sí	3	casado/a	no	sí	Pred=sí ✓
22	0	no	0	casado/a	sí	no	Pred=no ✓
25	1	no	1	soltero/a	no	no	Pred=no ✓
44	0	no	2	divorciado/a	sí	no	Pred=no ✓
39	1	sí	2	casado/a	no	no	Pred=no ✓
26	1	no	2	soltero/a	sí	no	Pred=no ✓
40	3	sí	1	casado/a	no	no	Pred=no ✓
53	2	sí	2	divorciado/a	no	sí	Pred=sí ✓
64	2	sí	3	divorciado/a	sí	no	Pred=no ✓
58	2	sí	2	casado/a	sí	sí	Pred=sí ✓
33	1	no	1	soltero/a	no	no	Pred=no ✓

**Hipótesis 3:** Si el cliente tiene 52, 53, 58 o 66 años, entonces querrá comprar un bote. Si no, no.

**Accuracy** (exactitud): **100%** (12/12)

# Aprendizaje Inductivo

Edad	#autos	¿Casa propia?	#hijos	Estado civil	¿Tiene perros?	¿Compró un bote?
66	1	sí	2	viudo/a	no	<i>sí</i>
52	2	sí	3	casado/a	no	<i>sí</i>
22	0	no	0	casado/a	sí	<i>no</i>
25	1	no	1	soltero/a	no	<i>no</i>
44	0	no	2	divorciado/a	sí	<i>no</i>
39	1	sí	2	casado/a	no	<i>no</i>
26	1	no	2	soltero/a	sí	<i>no</i>
40	3	sí	1	casado/a	no	<i>no</i>
53	2	sí	2	divorciado/a	no	<i>sí</i>
64	2	sí	3	divorciado/a	sí	<i>no</i>
58	2	sí	2	casado/a	sí	<i>sí</i>
33	1	no	1	soltero/a	no	<i>no</i>

**Hipótesis 4:** Si el cliente tiene casa, más de 40 años y tiene al menos 1 auto, entonces querrá comprar un bote.  
Si no, no.

**Accuracy** (exactitud): ?

# Aprendizaje Inductivo

Edad	#autos	¿Casa propia?	#hijos	Estado civil	¿Tiene perros?	¿Compró un bote?	
66	1	sí	2	viudo/a	no	sí	Pred=sí ✓
52	2	sí	3	casado/a	no	sí	Pred=sí ✓
22	0	no	0	casado/a	sí	no	Pred=no ✓
25	1	no	1	soltero/a	no	no	Pred=no ✓
44	0	no	2	divorciado/a	sí	no	Pred=no ✓
39	1	sí	2	casado/a	no	no	Pred=no ✓
26	1	no	2	soltero/a	sí	no	Pred=no ✓
40	3	sí	1	casado/a	no	no	Pred=sí ✗
53	2	sí	2	divorciado/a	no	sí	Pred=sí ✓
64	2	sí	3	divorciado/a	sí	no	Pred=sí ✗
58	2	sí	2	casado/a	sí	sí	Pred=sí ✓
33	1	no	1	soltero/a	no	no	Pred=no ✓

**Hipótesis 4:** Si el cliente tiene casa, más de 40 años y tiene al menos 1 auto, entonces querrá comprar un bote.  
Si no, no.

**Accuracy** (exactitud): 83% (10/12)

# Generalización

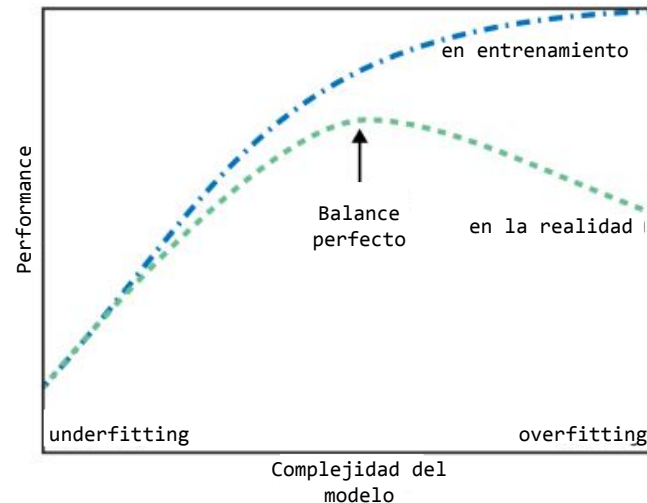
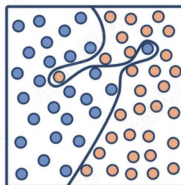
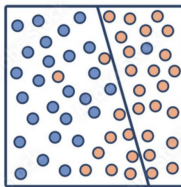
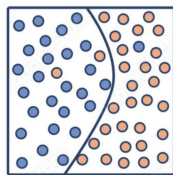
Queremos un modelo lo **suficientemente rico** para captar ideas complejas. Pero sin caer en:

## Subajuste (underfitting)

Construcción de un modelo **demasiado simple** que no capture la información disponible.

## Sobreajuste (overfitting)

Construcción de un **modelo demasiado complejo** para la cantidad de información que disponemos.



Müller, A. C., & Guido, S. (2016). Introduction to machine learning with Python: a guide for data scientists.

Además, queremos encontrar reglas que funcionen para **nuevos** ejemplos. No para los que ya conocemos su respuesta. Veremos más adelante que la forma principal de medir cómo funciona un modelo será evaluando las predicciones sobre un **conjunto "nuevo" de datos**.

# Generalización

Para intentar formalizar un poco estos conceptos, podemos utilizar las siguientes definiciones basadas en [1]

Se desea aprender una función objetivo  $\mathbf{f}^*$  **desconocida** mediante un conjunto finito de datos de entrenamiento  $\mathbf{D} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_{i=1}^N$ , una muestra de la distribución real (y de nuevo.. desconocida) de los datos.

Decimos que  $\mathbf{h}_{L,D}$ , un modelo construido usando el algoritmo  $L$  sobre  $\mathbf{D}$  **subajusta** si:

$$(\exists h' : H) Err_{train}(h_{L,D}) \geq Err_{train}(h') \wedge Err_{true}(h_{L,D}) > Err_{true}(h')$$

Análogamente,  $\mathbf{h}_{L,D}$  **sobreajusta** si:

$$(\exists h' : H) Err_{train}(h_{L,D}) \leq Err_{train}(h') \wedge Err_{true}(h_{L,D}) > Err_{true}(h')$$

En donde

$$Err_{train}(h) = \frac{1}{|D|} \sum_{(x^{(i)}, y^{(i)}) \in D} error(h(x^{(i)}), y^{(i)})$$

$$Err_{true}(h) = \mathbb{E}_x[error(h(x), y)] = \int error(h(x), y) P(x, y) dx$$

¿De donde sale y?

Ya veremos más adelante que  $\mathbf{y} \approx \mathbf{f}^*(\mathbf{x})$

¿Y qué es  $error(\mathbf{A}, \mathbf{B})$ ?

Depende del problema y del tipo de problema (reg, clasif, etc)

# Modelos Simples

En general queremos encontrar el modelo que cumpla el siguiente principio:

## Navaja de Ockham

(un principio metodológico y filosófico).

Cuando tenemos dos modelos que compiten, **producen las mismas predicciones**, debemos elegir el más simple.

Un modelo más simple es aquel que tiene menos parámetros o menor complejidad estructural.

## ¿Por qué?

En la práctica se ve que ayuda a:

- Evitar el sobreajuste y por lo tanto mejorar el rendimiento en datos no vistos.
- Mejora la interpretación: Los modelos más simples son más fáciles de interpretar y entender.
- Reduce los costos computacionales.

# Aprendizaje Inductivo

Edad	#autos	¿Casa propia?	#hijos	Estado civil	¿Tiene perros?	¿Compró un bote?
66	1	sí	2	viudo/a	no	sí
52	2	sí	3	casado/a	no	sí
22	0	no	0	casado/a	sí	no
25	1	no	1	soltero/a	no	no
44	0	no	2	divorciado/a	sí	no
39	1	sí	2	casado/a	no	no
26	1	no	2	soltero/a	sí	no
40	3	sí	1	casado/a	no	no
53	2	sí	2	divorciado/a	no	sí
64	2	sí	3	divorciado/a	sí	no
58	2	sí	2	casado/a	sí	sí
33	1	no	1	soltero/a	no	no

**Otro ejemplo de Hipótesis:** Si el cliente es mayor de 45 años y tiene menos de 3 chicos o no está divorciado, entonces querrá comprar un bote. Si no, no.

¿Cuál es el tamaño del **Espacio de Instancias**? (cuántas instancias posibles existen)

¿Cuál es el tamaño del **Espacio de Hipótesis**? (cuántos modelos posibles se pueden construir)

# Sustento del Aprendizaje Inductivo

## ¿Compra Bote?

Veamos cuántas posibilidades hay para cada atributo suponiendo que cada atributo puede tener:

- Edad: 100 valores posibles
- #autos: 6
- ¿casa propia?: 2
- #hijos: 5
- estado civil: 5
- ¿tiene perros?: 2

El **espacio de instancias  $X$**  contiene exactamente:  $100 * 6 * 2 * 6 * 5 * 2$  elementos.

**$|X| = 72000$**

El **espacio de hipótesis** (cuántos modelos posibles se pueden construir) contiene:

$2^{72000}$  posibles elementos (subconjuntos de  $X$ )

## Sustento del Aprendizaje Inductivo

Cualquier **hipótesis (o modelo)** que aproxime bien a la **función objetivo** sobre un conjunto **suficientemente grande de ejemplos** de entrenamiento, también **aproximará bien** a la función objetivo sobre otros **ejemplos no observados<sup>(a)</sup>**.

Si vemos el aprendizaje como un **problema de búsqueda**, necesitamos diferentes estrategias para buscar en el espacio de hipótesis.

Estaremos particularmente interesados en **algoritmos capaces de buscar eficientemente** espacios de hipótesis **muy grandes o infinitos**, para encontrar las hipótesis que “mejor” se ajusten a los datos de entrenamiento<sup>(b)</sup>.

**(a)** [Mitchell 2.2.2 “THE INDUCTIVE LEARNING HYPOTHESIS”]

**(b)** [Mitchell 2.3 “CONCEPT LEARNING AS SEARCH”]



# Sesgo Inductivo

Un **conjunto finito** de datos nunca alcanza para **inferir** un único modelo. Es por ello que tenemos que agregar supuestos.

## Definición: Sesgo Inductivo

Considere un algoritmo de aprendizaje  $L$  para el dominio  $X$ . Sean

$f^*$  una función arbitraria (generalmente desconocida) definida sobre  $X$ .

$D = \{(x^{(i)}, f^*(x^{(i)}))\}_{i=1}^N$  un conjunto arbitrario de ejemplos de entrenamiento (o dataset).

$f_{L,D}(x^{(i)})$  la clasificación asignada a la instancia  $x^{(i)}$  por  $f_{L,D}$  (una función que aproxima a  $f^*$ ) construida utilizando  $L$  sobre los datos  $D$ .

Dada cualquier instancia del dominio  $X$ , llamémosla  $x$ , el **sesgo inductivo** del modelo  $f_{L,D}$  es cualquier conjunto mínimo de afirmaciones  $B$  tal que para cualquier función objetivo  $f^*$  y los ejemplos de entrenamiento de  $D$ , se puede deducir el valor  $f_{L,D}(x)$ .

Mitchell lo sintetiza como:

$$(\forall x \in X)[(B \wedge D \wedge x) \vdash f_{L,D}(x)]$$

En otras palabras: **Conjunto de supuestos** que el algoritmo de aprendizaje utiliza para hacer predicciones sobre nuevos datos no vistos, basándose en los datos de entrenamiento que ha visto.

Estos supuestos pueden provenir de una variedad de fuentes, incluyendo:

- Estructura del modelo
- La función objetivo que el algoritmo esté optimizando
- Características de funcionamiento del algoritmo (cómo recorre el espacio de hipótesis hasta elegir un único modelo, la semilla utilizada)
- etc

El sesgo inductivo determina los **tipos de funciones** que el algoritmo **puede aprender** y los **tipos de errores** que **se espera que cometa**.

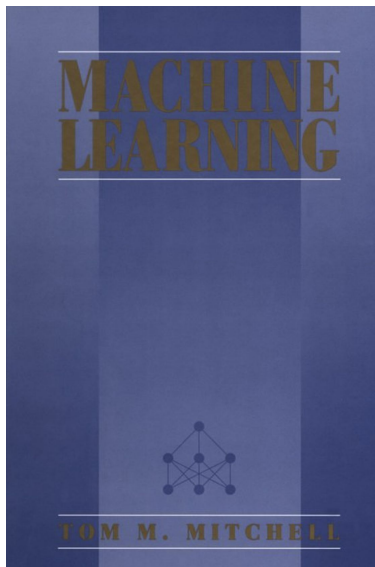
# Conclusiones

- Introdujimos los distintos tipos de aprendizaje automático.
- Hablamos de la diferencia entre **aprendizaje automático** y **aprendizaje estadístico**.
- Introdujimos la diferencia entre **clasificación** y **regresión**.
- Definimos:
  - Función objetivo
  - Hipótesis / Modelo.
  - Generalización, Overfitting, Underfitting.
  - Espacio de Hipótesis e Instancias.
  - Navaja de Ockham.
  - Sustento del Aprendizaje Inductivo.
  - Aprendizaje automático como un problema de búsqueda.
  - Sesgo Inductivo.

Próxima clase: primeros modelos predictivos, Árboles de Decisión.

# TAREA

- A) Leer **Secciones 2.1, 2.2, 2.3, 2.7, 2.8** del “Machine Learning” (Mitchell). [Descargar](#)
- B) Completar el Notebooks 01 (Herramientas) y 02 (Titanic).
- C) Completar el Cuestionario.



Tom Mitchell