

Curso de Inteligencia Artificial para Ingenieros

Introducción al *Machine Learning*

Prof. Carlos A. Toro N.
carlos.toro.ing@gmail.com
2022

Objetivos

Al finalizar esta presentación, usted podrá:

- ☐ Dominar la definición de algoritmos basados en aprendizaje y el proceso de *Machine Learning* (aprendizaje de máquina).
- ☐ Conocer los algoritmos más comunes de *Machine Learning* (M.L.).
- ☐ Comprender conceptos tales como hiperparámetros, descenso del gradiente y validación cruzada.

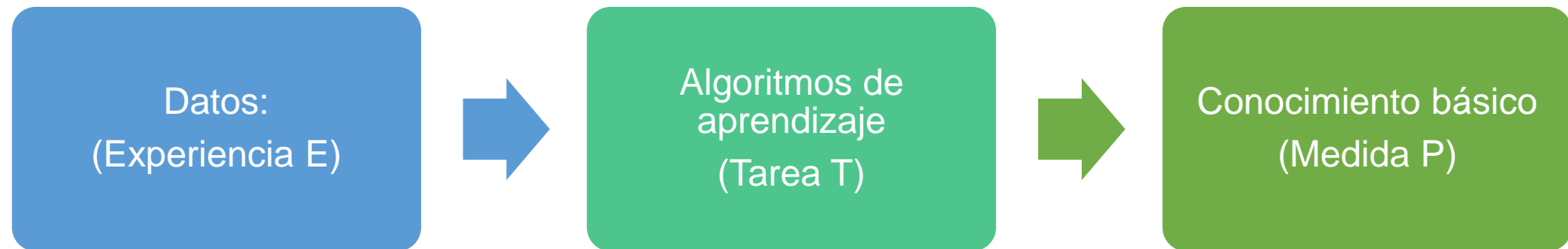
Contenidos

- ❑ Definición del *Machine Learning*
- ❑ Tipos de *Machine Learning*
- ❑ Proceso de *Machine Learning*
- ❑ Otros métodos clave de *Machine Learning*
- ❑ *Ejercicios prácticos: conociendo las librerías de herramientas de Python útiles para M.L.*

Definición de *Machine Learning*

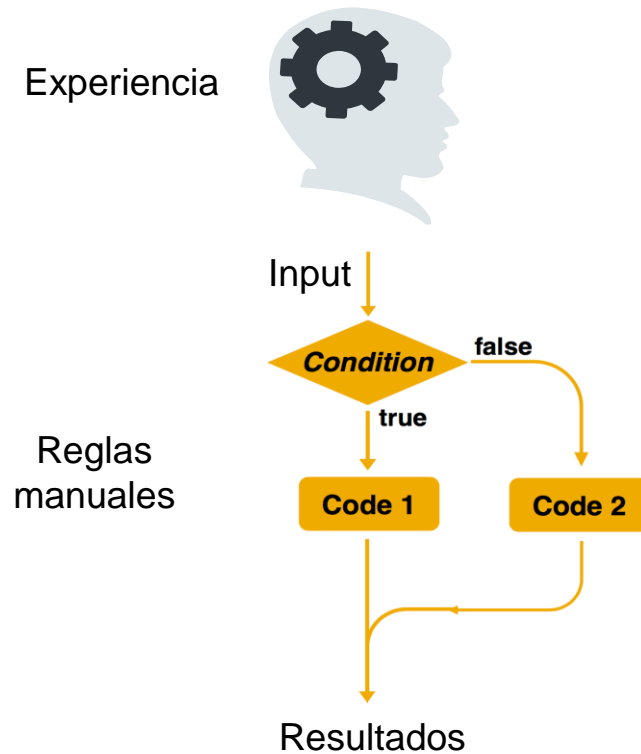
Algoritmos de *Machine Learning*

El *Machine Learning* (M.L.) o Aprendizaje de Máquinas (incluido *Deep Learning* (D.L.) o aprendizaje profundo) es un estudio de algoritmos de aprendizaje. “Se dice que un programa de computadora aprende de la **experiencia E** con respecto a alguna clase de **tareas T** y **medida de desempeño P** si su desempeño en las tareas en T , medido por P , mejora con la experiencia E ”. (Tom M. Mitchell)



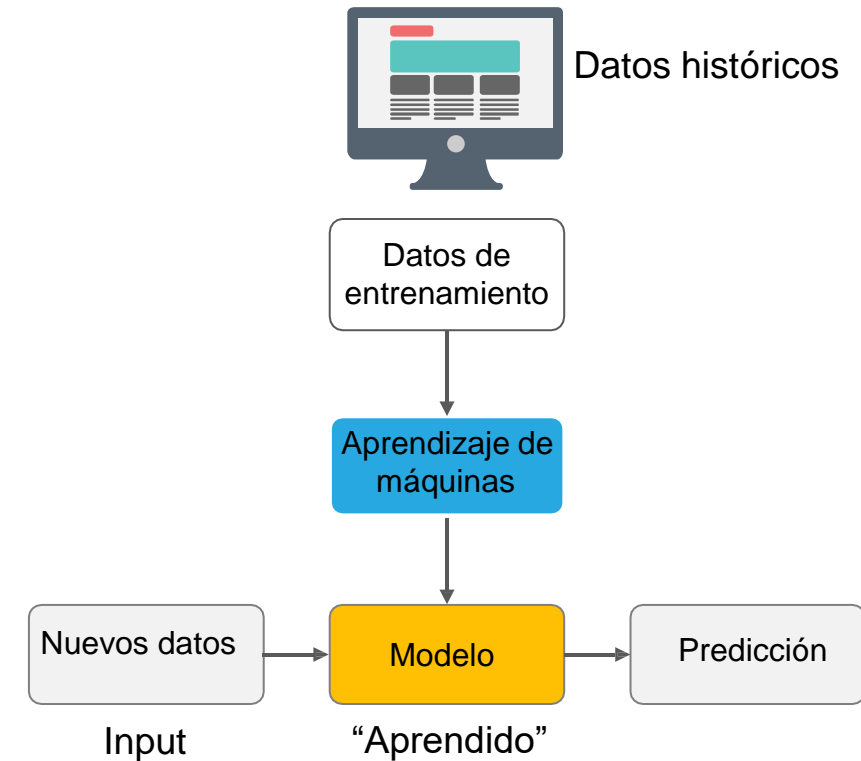
Algoritmos tradicionales vs *machine learning*

Algoritmos basados en reglas



- La programación explícita se utiliza para resolver problemas.
- Las reglas se pueden especificar manualmente.

Machine Learning

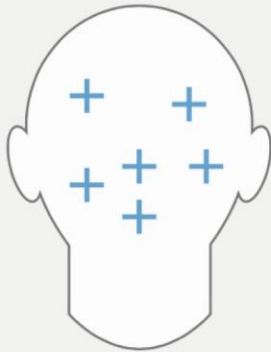


- Las muestras se utilizan para el entrenamiento.
- Las normas de toma de decisiones son complejas o difíciles de describir.
- Las reglas son aprendidas automáticamente por las máquinas.

Escenarios de aplicación del *Machine Learning* (1)

- ❑ La solución a un problema es compleja, o el problema puede implicar una gran cantidad de datos sin una función clara de distribución de datos.
- ❑ El M.L. se puede utilizar en los siguientes escenarios:

Las reglas son complejas o no pueden describirse, como el reconocimiento facial y el reconocimiento de voz.



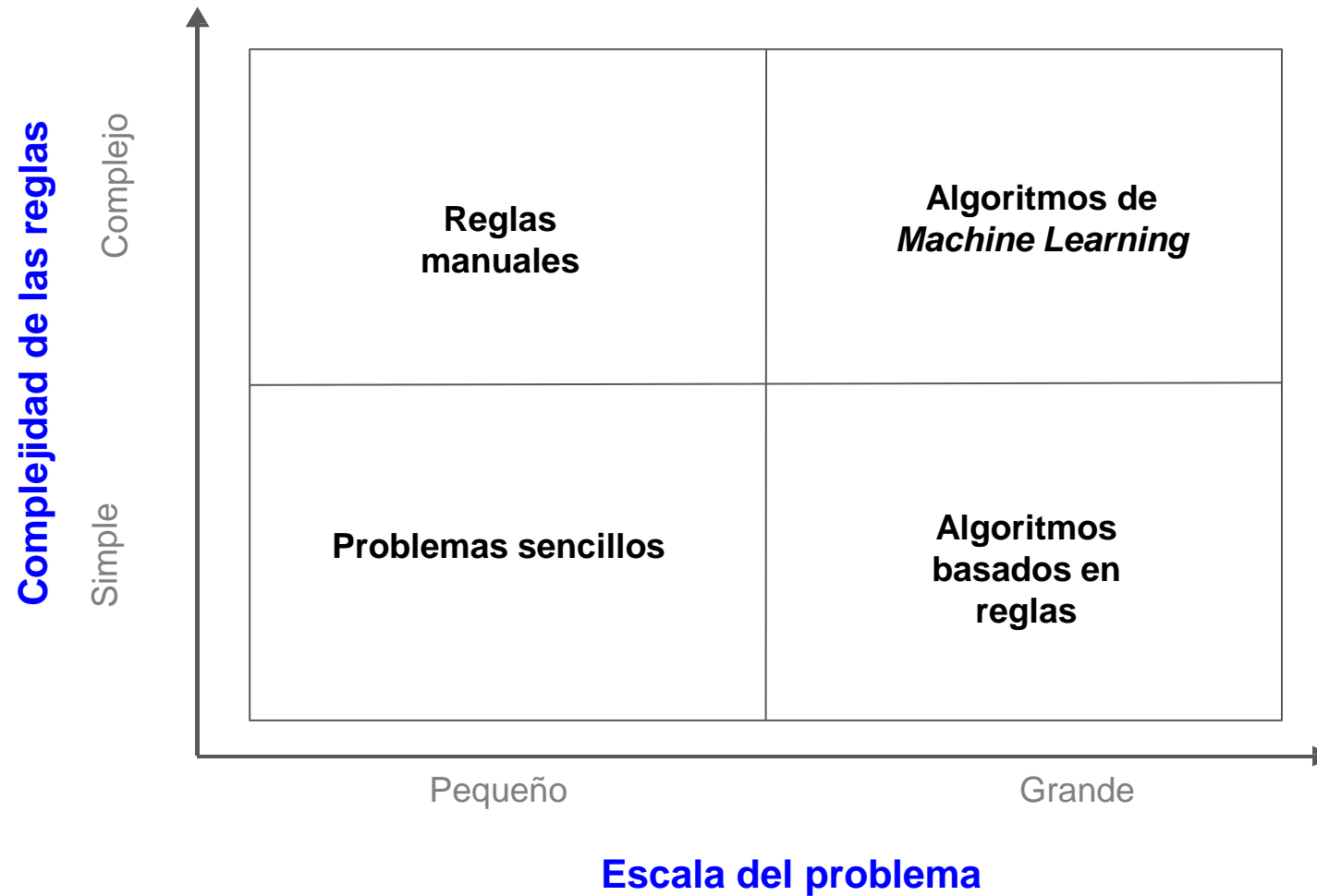
Las reglas de la tarea cambian con el tiempo. Por ejemplo, en la tarea de etiquetado de la parte de habla, se generan nuevas palabras o significados en cualquier momento.



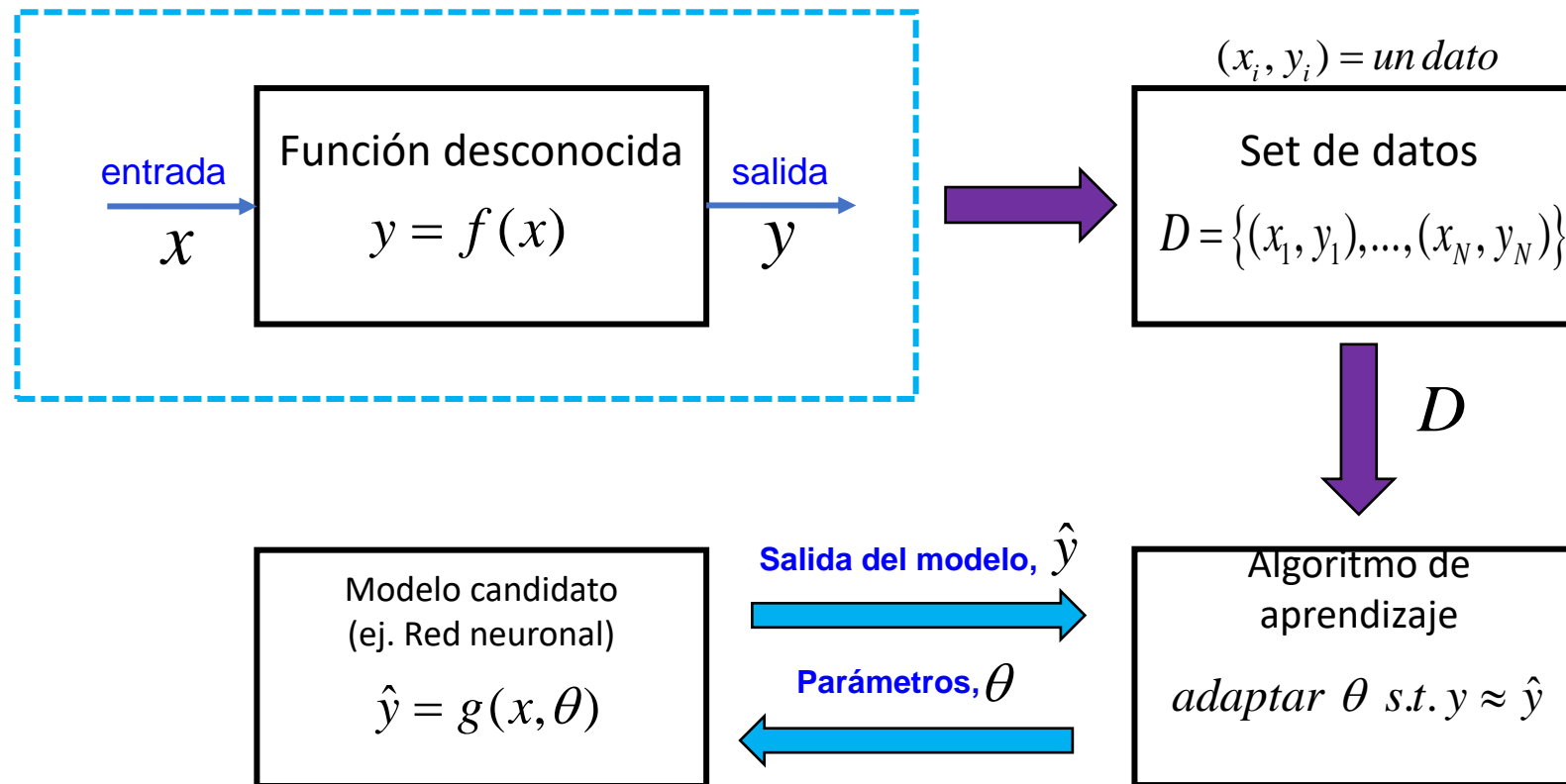
La distribución de datos cambia con el tiempo, lo que requiere una constante readaptación de los programas, como predecir la tendencia de las ventas de productos básicos.



Escenarios de aplicación del *Machine Learning* (2)

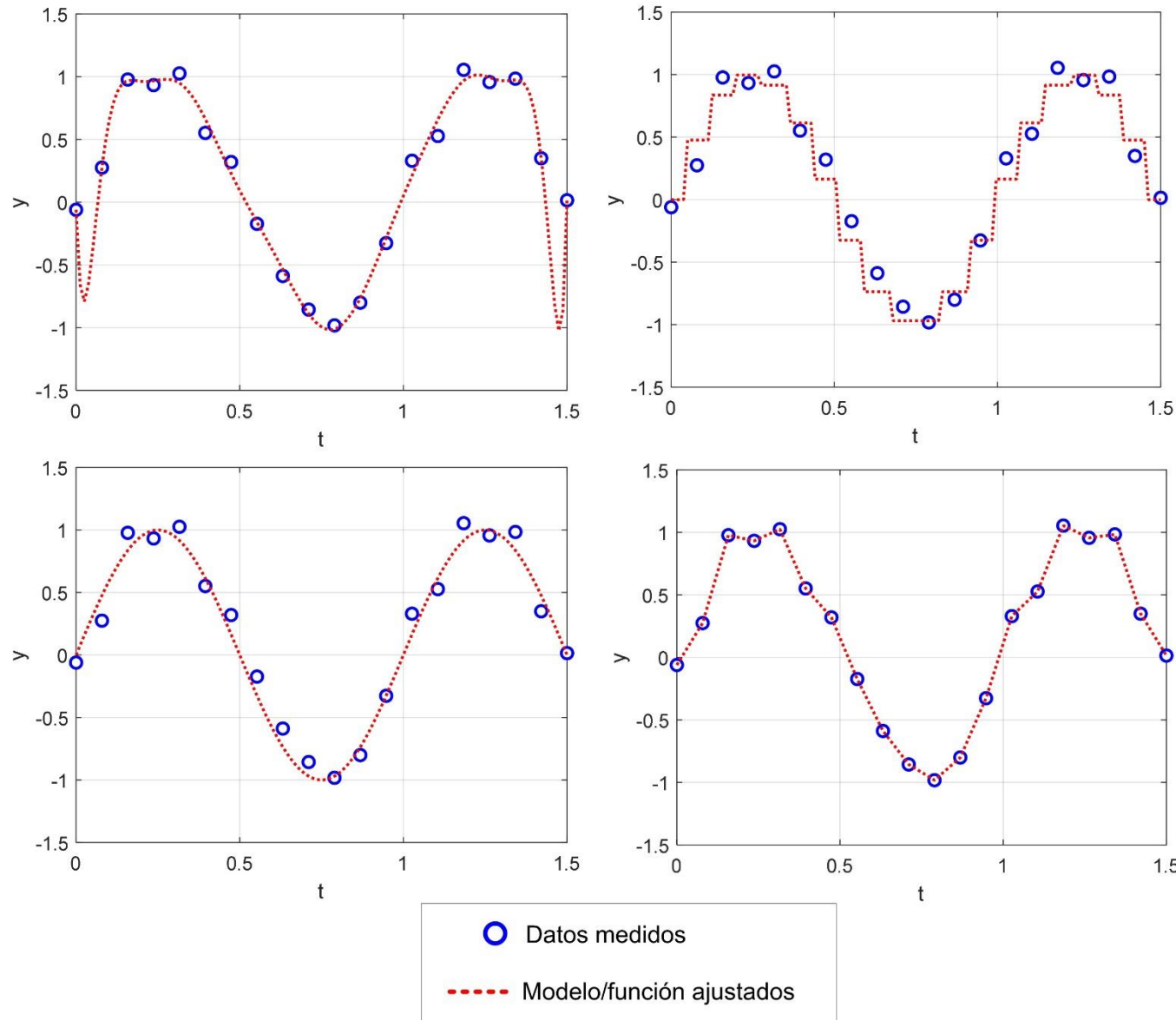


Esquema general de los algoritmos de *Machine Learning*



- La función objetivo f se desconoce. Los algoritmos de aprendizaje no pueden obtener una función perfecta f .
- Supongamos que la función de hipótesis g **se aproxima** a la función f , pero puede ser diferente de la función f .

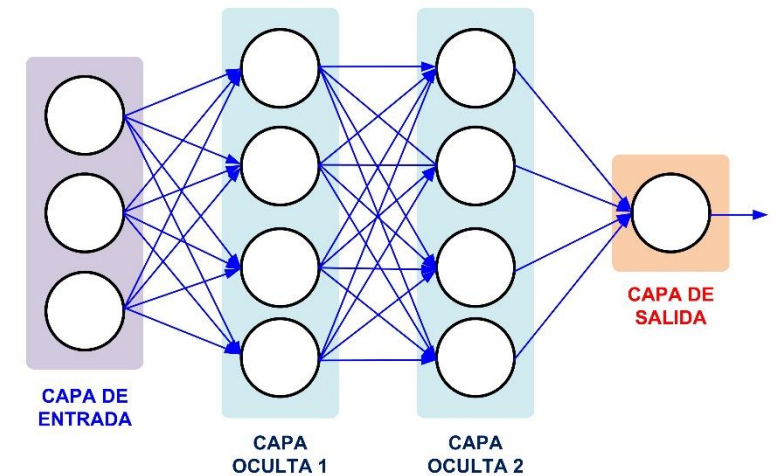
¿Cuál es el modelo correcto?



$$y = f(t) = ?$$



Polinomio?
 $\text{Asin}(2\pi ft)$?
 Red neuronal?



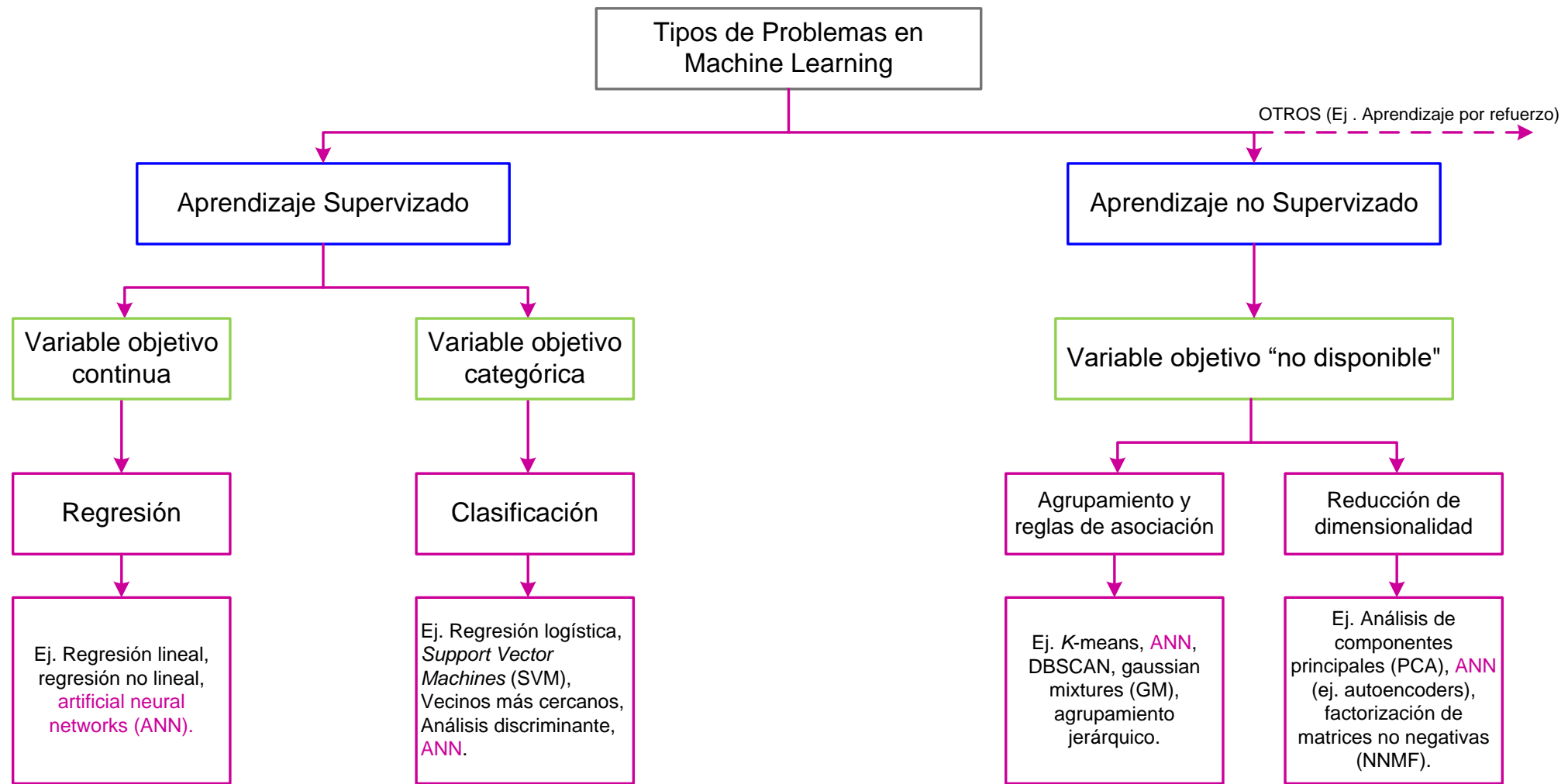
Tipos de *Machine Learning*

Clasificación de *Machine Learning*: los más usados

Dentro de los tipos más usados encontramos:

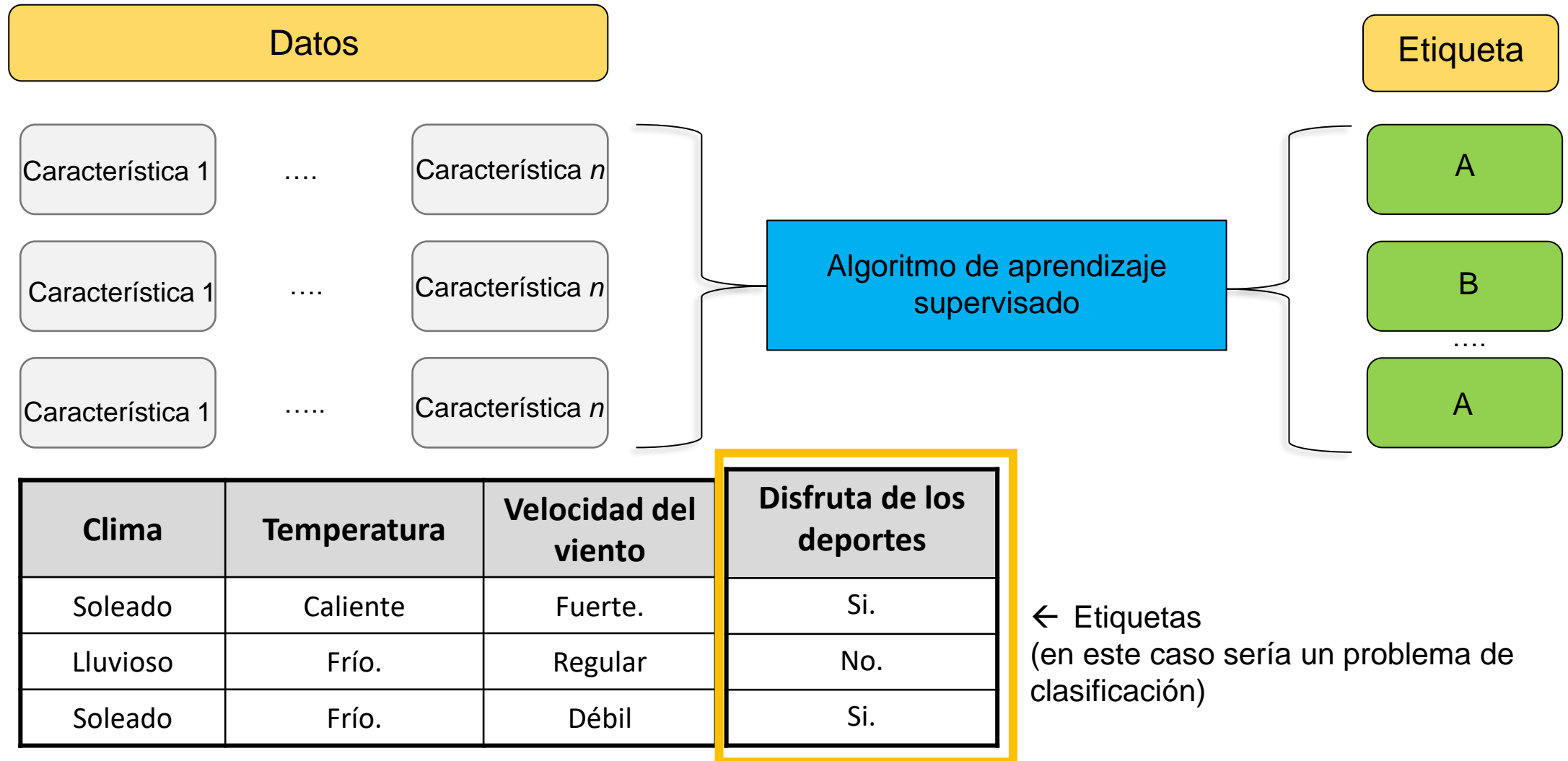
- ❑ **Aprendizaje supervisado:** Obtener un modelo óptimo con el desempeño requerido a través del entrenamiento y el aprendizaje basado en las muestras con variables objetivo o etiquetas conocidas. A continuación, utilizar el modelo para predecir las salidas de nuevos datos.
- ❑ **Aprendizaje no supervisado:** Para muestras sin etiqueta, los algoritmos de aprendizaje modelan directamente los conjuntos de datos de entrada. La agrupación (*Clustering*) es una forma común de aprendizaje no supervisado. Solo necesitamos juntar muestras muy similares, calcular la similitud entre las nuevas muestras y las existentes, y agruparlas por similitud.

Principales problemas resueltos por el *Machine Learning*



★ No perder de vista todas las técnicas de extracción de patrones y características relacionadas al dominio de la aplicación o especialidad.

Aprendizaje supervisado



Aprendizaje supervisado: Preguntas de regresión

□ **Regresión:** mapea las características o variables de las muestras presentes en un conjunto de datos a **valores continuos (CUANTITATIVO)**. La dependencia entre las características y la variable objetivo se descubre expresando la relación a través de funciones que mapean las variables. Los algoritmos de aprendizaje suelen generar una función $f: R^n \rightarrow R$.

Ejemplos de preguntas que responde este tipo de problema:

- ¿Cuánto me beneficiaré de las acciones la semana que viene?
- ¿Cuál será la temperatura del martes?
- ¿Cuál es la concentración del compuesto X en un estanque medida a partir del color de la solución?



¿Cuál será la temperatura de mañana?



[Fuente imagen](#)

Aprendizaje supervisado: Preguntas de clasificación

□ **Clasificación:** mapea muestras de un conjunto de datos a una categoría/clase (k variables discretas) específica utilizando un modelo de clasificación **(CUALITATIVO)**. Para realizar esta tarea, los algoritmos de aprendizaje suelen generar una función $f: R^n \rightarrow (1, 2, \dots, k)$.

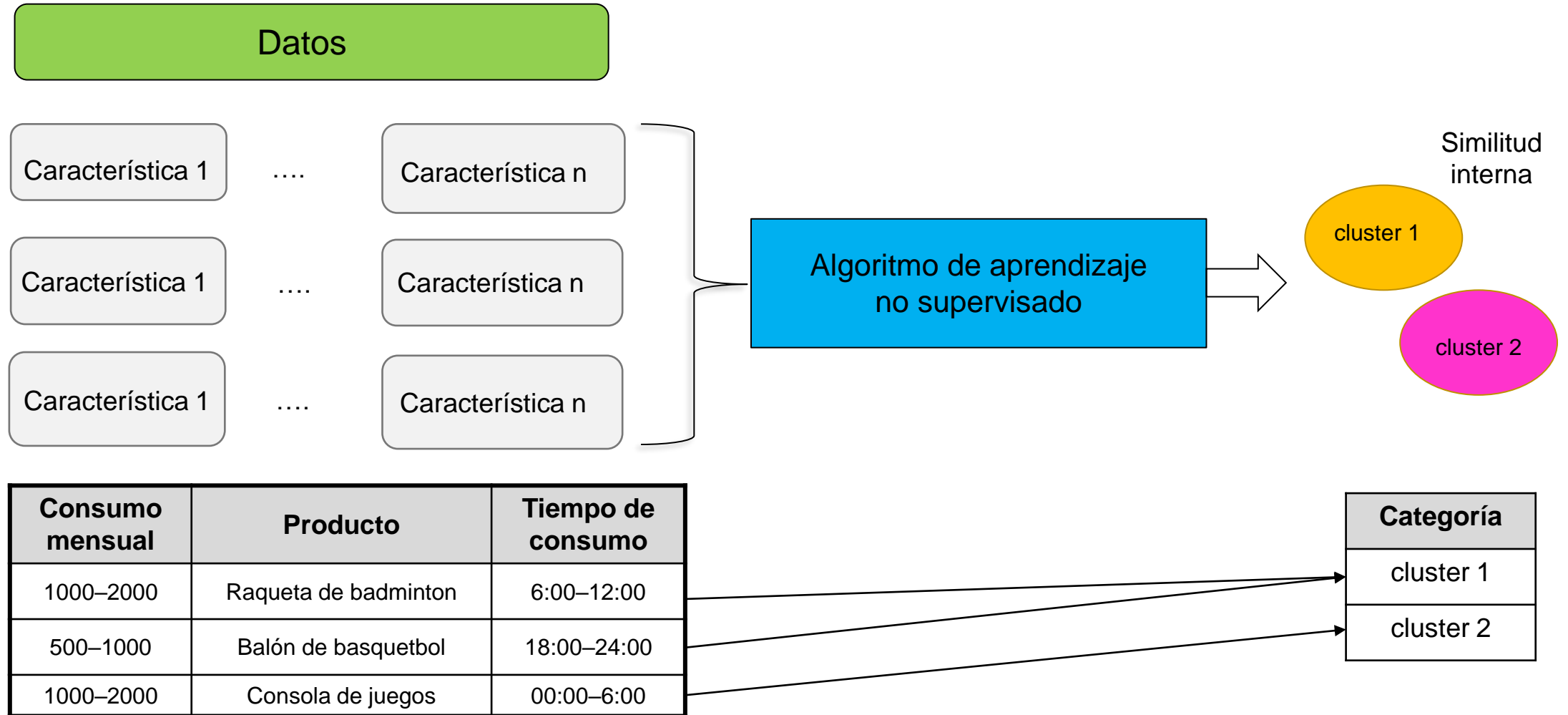
Ejemplos de preguntas que responde este tipo de problema:

- ¿Habrá un embotellamiento en la calle X mañana en la hora punta?
- ¿Qué método es más atractivo para los clientes?
¿Voucher de X USD, o 25% de descuento?
- ¿Presenta fallas la pieza mecánica fabricada?



Nota: El valor k dependerá del tipo de problema, por ejemplo, si se quieren clasificar datos en solo dos categorías, $k = 2$, se trata de un problema de clasificación binaria, para $k \geq 2$ se tendrá un problema de clasificación multiclase.

Aprendizaje no supervisado

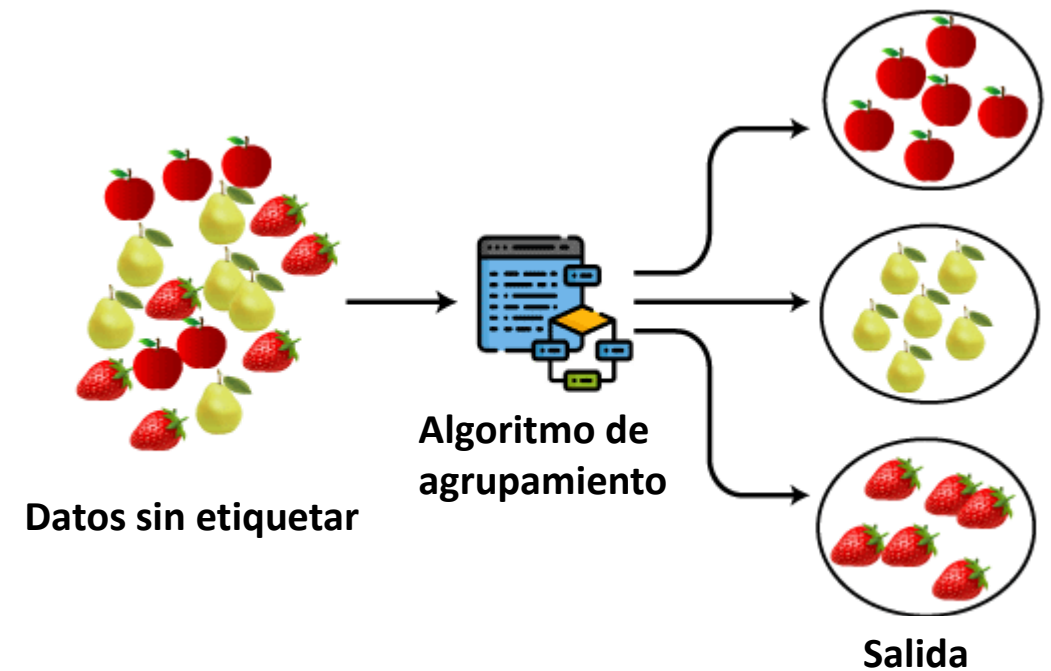


Aprendizaje no supervisado: Preguntas de agrupamiento

❑ **Agrupamiento o Clustering:** **agrupa** las muestras de un conjunto de **datos sin etiquetar** según la similitud interna de los datos. La similitud de las muestras pertenecientes a la misma categoría es elevada.

Ejemplos de preguntas que responde este tipo de problema:

- ¿A qué público le gusta ver películas del mismo tema?
- ¿Cuáles de estos componentes están dañados de una manera similar?



[Fuente imagen](#)

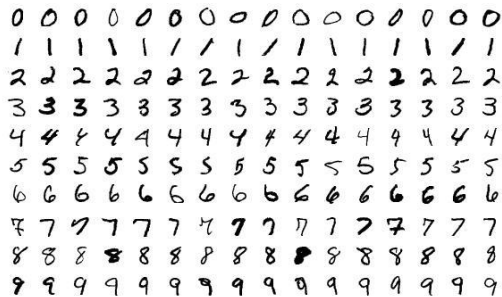
Nota: uno de los algoritmos que cae dentro de esta categoría es el de K-medias (K-means).

Aprendizaje no supervisado: Preguntas de agrupamiento

- ❑ **Reducción de dimensionalidad:** toman el conjunto de datos donde cada muestra se representa por múltiples variables para así mapearlas a un nuevo espacio vectorial donde se representen con menos variables. Un algoritmo típico dentro de estas técnicas es el análisis de componentes principales o PCA por sus siglas en inglés.

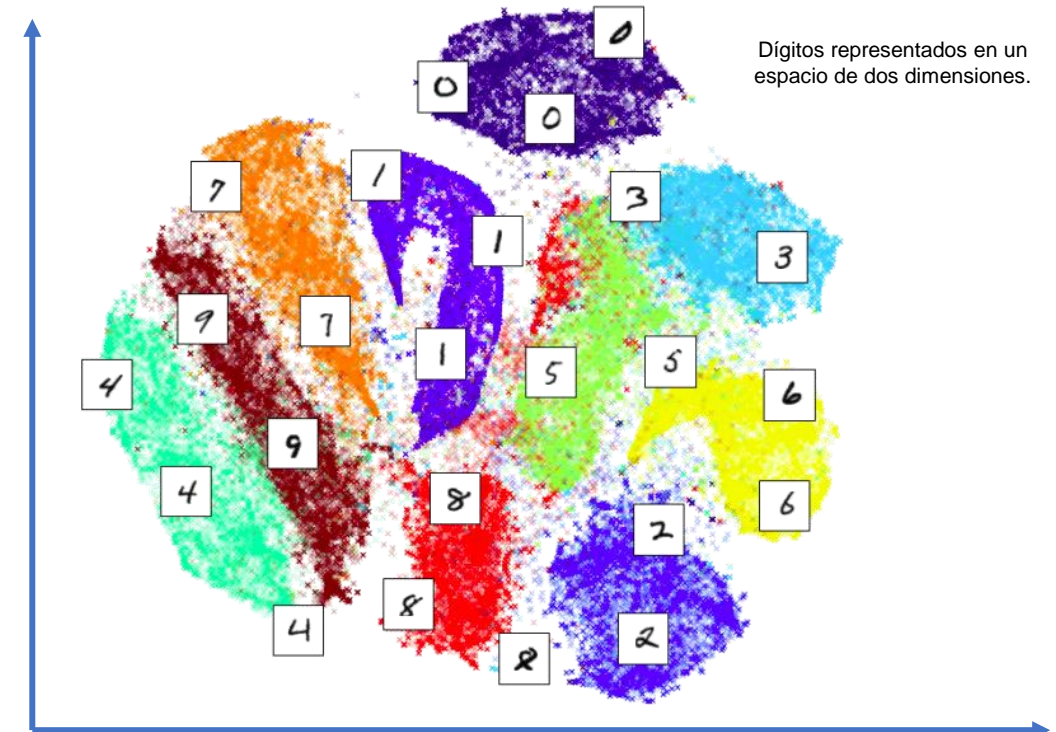
Ejemplo de preguntas que responde este tipo de problema:

- ¿es posible representar las imágenes de un conjunto de datos en un espacio de dos dimensiones que facilite su visualización?



Imágenes de 28x28 píxeles de dígitos escritos a mano del MNIST
(Modified National Institute of Standards and Technology).

Método de reducción de dimensionalidad.



Dígitos representados en un espacio de dos dimensiones.

[Fuente imagen](#)

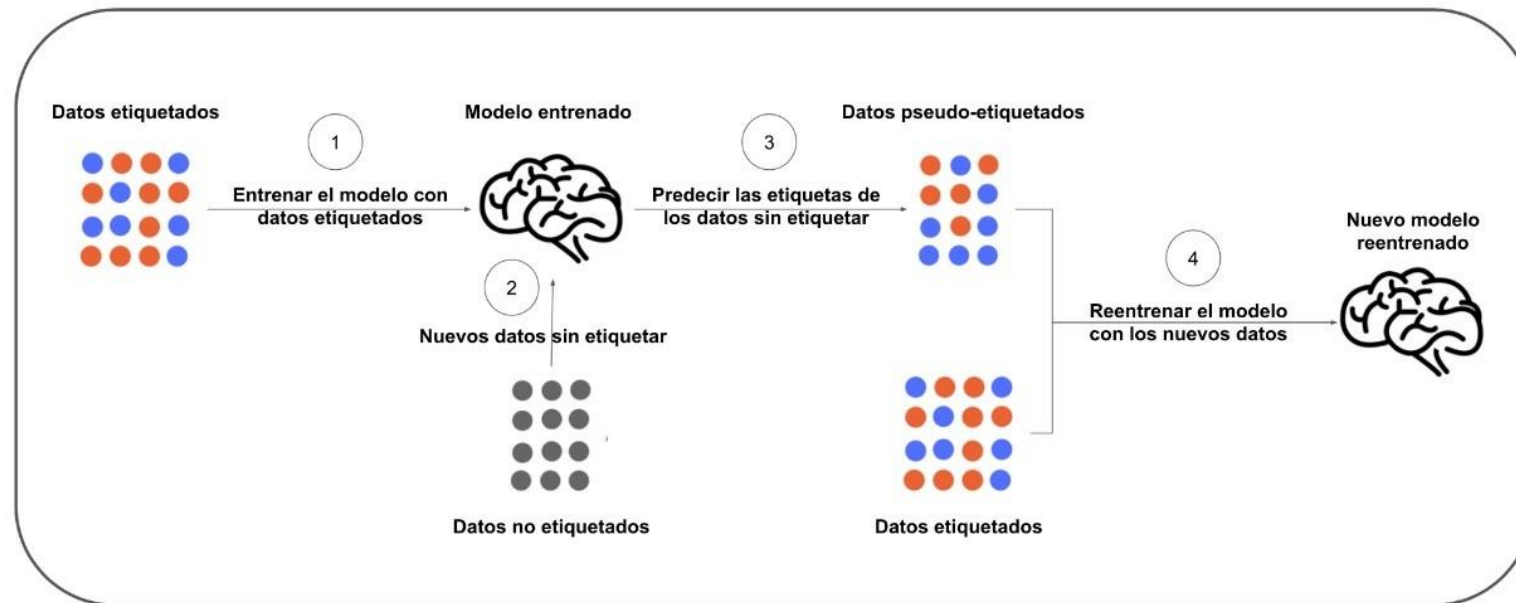
Clasificación de *Machine Learning*: otros tipos de aprendizaje

Otros tipos de aprendizaje que encontramos son:

- ❑ **Aprendizaje semi-supervisado:** En esta tarea, un modelo de *M.L.* que automáticamente utiliza una gran cantidad de datos sin etiquetar para ayudar a aprender directamente desde una pequeña cantidad de datos etiquetados.
- ❑ **Aprendizaje auto-supervisado:** En este caso, el modelo se entrena para aprender una parte de los datos de entrada desde otra parte de esos mismos datos. Por ejemplo, encuentran bastante uso en modelación de lenguaje natural, modelos como BERT y GPT son un ejemplo de esto.
- ❑ **Aprendizaje reforzado:** Se trata de un área de *M.L.* que se ocupa de cómo los agentes deben tomar medidas en un entorno para maximizar cierta noción de recompensa acumulativa. La diferencia entre el aprendizaje reforzado y el aprendizaje supervisado es la señal del profesor. La señal de refuerzo proporcionada por el entorno en el aprendizaje reforzado se utiliza para evaluar la acción (señal escalar) en lugar de decirle al sistema de aprendizaje cómo realizar las acciones correctas.

Aprendizaje semi-supervisado

- ❑ Este tipo de aprendizaje surge como respuesta al costoso trabajo que puede ser para algunas tareas el etiquetar los datos. Una forma de ejecutar este tipo de aprendizaje consiste en aplicar alguna estrategia de aprendizaje no supervisado para agrupar los datos no etiquetados y luego usar los datos etiquetados para asignarle la etiqueta a los no etiquetados, finalmente, se reentrena el modelo para conseguir un modelo supervisado mas robusto. Por supuesto, la calidad del resultado dependerá de que tan bien fue el entrenamiento del modelo inicial.

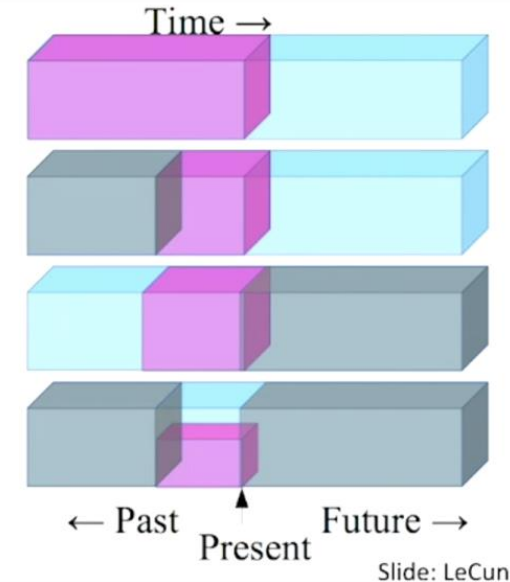


[Fuente imagen](#)

Aprendizaje auto-supervisado

- ❑ Algunos ejemplos de este tipo de aprendizaje lo encontramos en tareas como predecir la siguiente oración en un texto dado una oración inicial, por ejemplo, en este caso tenemos modelos como GPT-3 (*Generative Pre-Trained Transformer 3*) de [OpenAI](#). Este tipo de modelación aprovecha porciones de una muestra de los datos para predecir la restante, cómo se observa en las figura. Puede ser usado también para imágenes o señales.

- ▶ Predict any part of the input from any other part.
- ▶ Predict the **future** from the **past**.
- ▶ Predict the **future** from the **recent past**.
- ▶ Predict the **past** from the **present**.
- ▶ Predict the **top** from the **bottom**.
- ▶ Predict the occluded from the visible
- ▶ **Pretend there is a part of the input you don't know and predict that.**



[Fuente imagen](#)

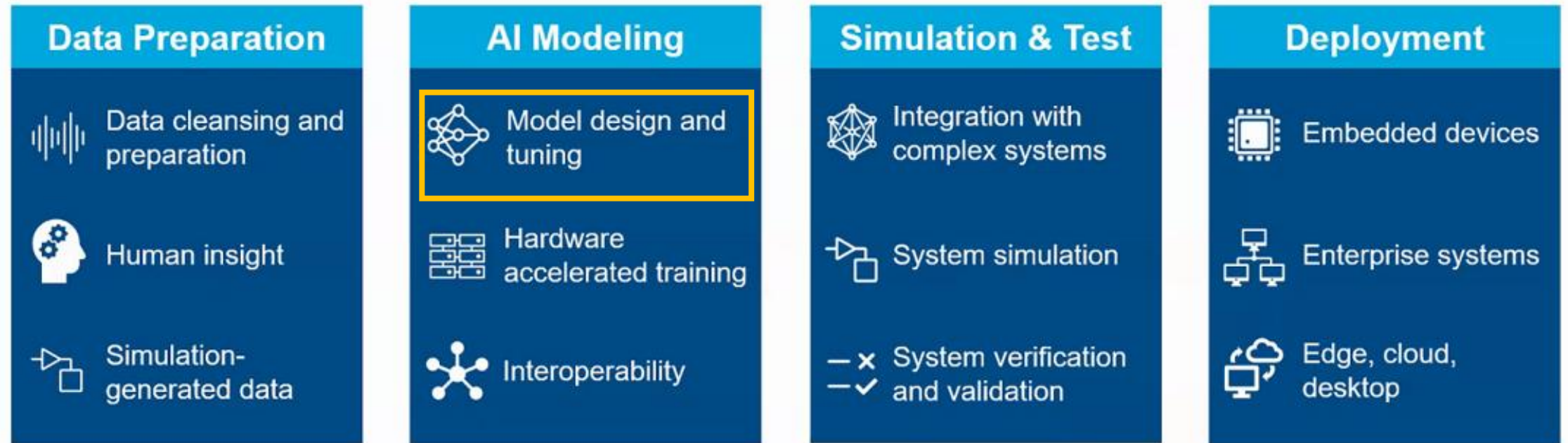
- Principalmente usado en juegos y sistemas robóticos.



Proceso de *Machine Learning*

Proceso de *Machine Learning*

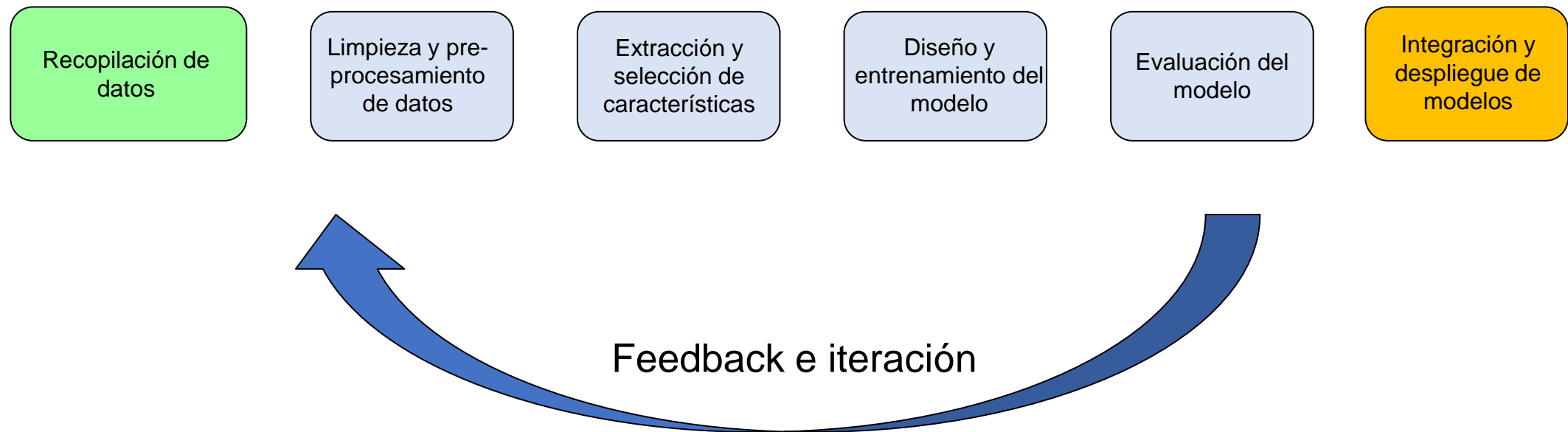
- ❑ La generación de un modelo de M.L. es solo una parte del proceso completo del diseño de un sistema que usa estas tecnologías.



Fuente imagen: [Mathworks](https://www.mathworks.com)

Proceso de *Machine Learning*

- ❑ En particular, antes del proceso de modelación, definir bien los requerimientos del problema y de la solución, quizás no sea necesario usar estas técnicas.



Concepto básico de machine learning: el *Dataset* o conjunto de datos

- ❑ **Dataset:** una colección de datos utilizados en tareas de M.L. Cada registro de datos se llama muestra. Los eventos o atributos que reflejan el desempeño o la naturaleza de una muestra en un aspecto particular se llaman características (*features*).
- ❑ **Conjunto de entrenamiento:** conjunto de datos (dataset) utilizado en el proceso de entrenamiento, en el que cada muestra se denomina una muestra de entrenamiento. El proceso de creación de un modelo a partir de estos datos se llama aprendizaje (entrenamiento).
- ❑ **Conjunto de pruebas:** El proceso de prueba del modelo se refiere a su uso después del aprendizaje para la predicción de resultados en muestras nuevas. El conjunto de datos utilizado se llama un conjunto de pruebas, y cada muestra es llamada una muestra de prueba.

Estructura general de un conjunto de datos

- ❑ En el caso de problemas de M.L. es típico encontrar los conjuntos de datos estructurados de la siguiente forma:

		DESCRIPTORES/CARACTERÍSTICAS			Etiqueta/variable objetivo	
		Característica 1	Característica 2	Característica 3		
		No.	Area	Distritos escolares	Dirección	Precio de la casa
Set de Entrenamiento	1	100	8	South	1000	
	2	120	9	Southwest	1300	
	3	60	6	North	700	
	4	80	9	Southeast	1100	
Set de prueba	5	95	3	South	850	

OBSERVACIONES O MUESTRAS

Nota: típicamente, el conjunto de datos original se divide en los dos que se muestran (más uno de validación que discutiremos luego), el porcentaje de división para uno u otro dependerá de la cantidad de datos disponibles, aunque generalmente es en una proporción 80% (train)/20% (test).

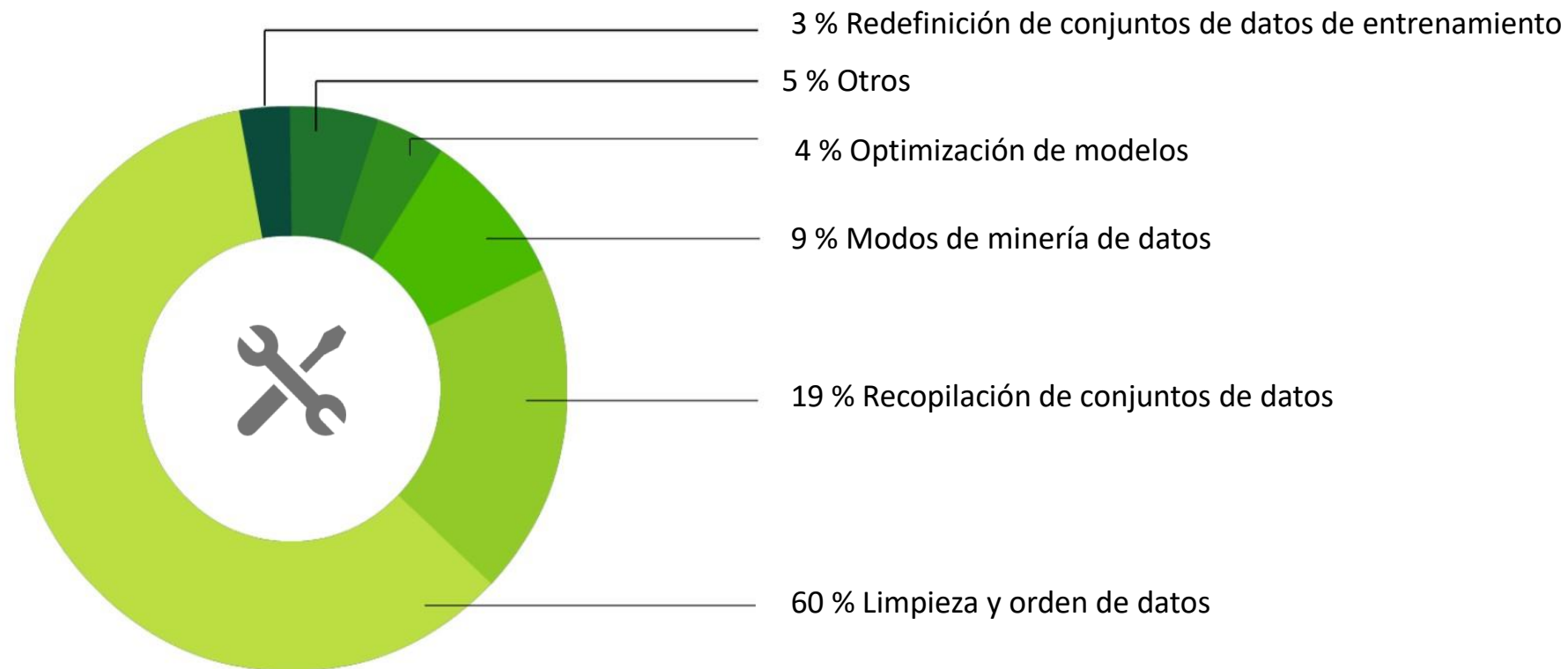
Importancia del pre-procesamiento de los datos

- ❑ Los datos son cruciales para los modelos. Es el techo de las capacidades del modelo. Sin buenos datos, no hay un buen modelo.



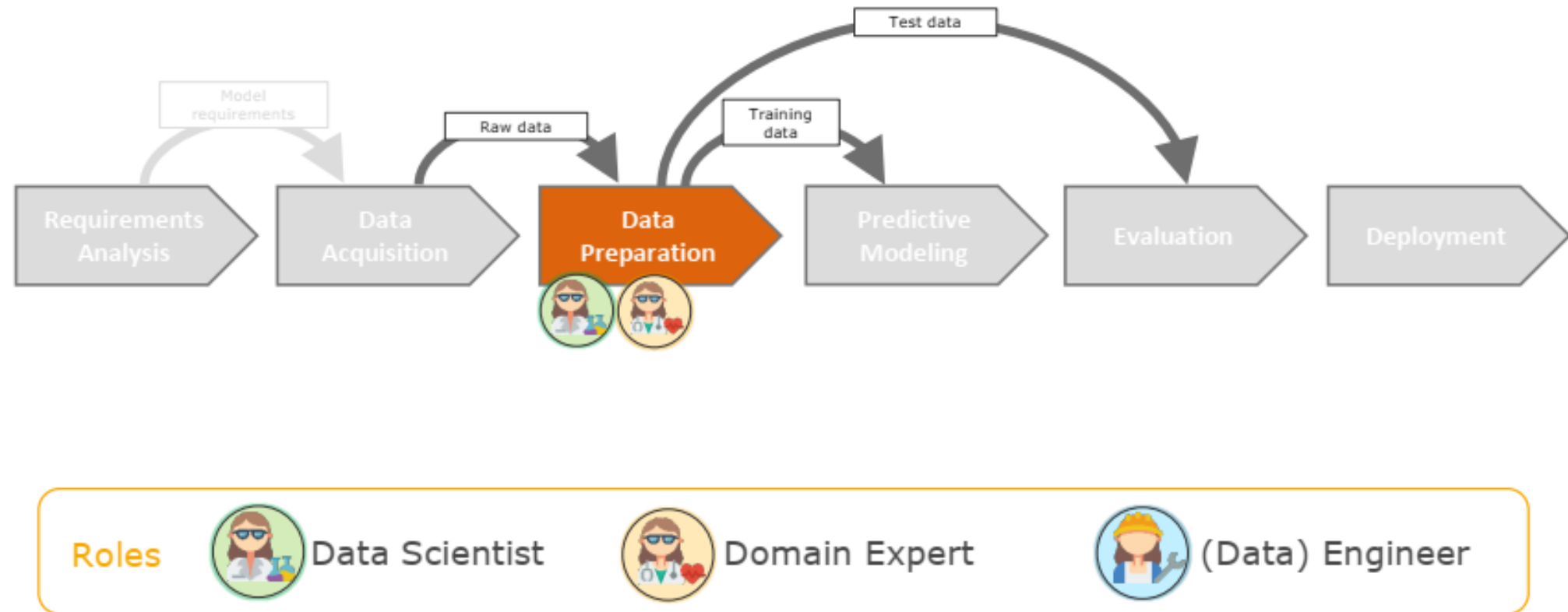
Tareas de limpieza de datos

- ❑ Estadísticas sobre el trabajo de los científicos en M.L. (un aproximado, algunas etapas pueden variar según disciplina de aplicación).



Informe CrowdFlower Data Science 2016

Tareas de limpieza de datos



Recordar: es importante trabajar de la mano siempre con el experto del dominio de aplicación, ej. biólogo, médico, metalurgista, mecánico, etc. son ellos quienes finalmente aportan contexto a los datos y modelos.

Limpieza de datos

- ❑ La mayoría de los modelos de ML procesan características, que suelen ser representaciones numéricas de variables de entrada que se pueden utilizar en el modelo.
- ❑ En la mayoría de los casos, los datos recogidos pueden ser utilizados por algoritmos sólo después de ser preprocesados. Las operaciones de preprocesamiento incluyen las siguientes:
 - Filtrado de datos
 - Procesamiento de datos perdidos
 - Procesamiento de posibles excepciones, errores o valores anormales
 - Combinación de datos de múltiples fuentes de datos
 - Consolidación de datos

Datos sucios (1)

- Generalmente, los datos reales pueden tener algunos problemas de calidad:
 - **Incompletitud:** contiene valores que faltan o datos que carecen de atributos
 - **Ruido:** contiene registros incorrectos o excepciones.
 - **Inconsistencia:** contiene registros inconsistentes.

Datos sucios (2)

#	Id	Name	Birthday	Gender	Is Teacher	#Students	Country	City
1	111	John	31/12/1990	M	0	0	Ireland	Dublin
2	222	Mery	15/10/1978	F	1	15	Iceland	
3	333	Alice	19/04/2000	F	0	0	Spain	Madrid
4	444	Mark	01/11/1997	M	0	0	France	Paris
5	555	Alex	15/03/2000	A	1	23	Germany	Berlin
6	555	Peter	1983-12-01	M	1	10	Italy	Rome
7	777	Calvin	05/05/1995	M	0	0	Italy	Italy
8	888	Roxane	03/08/1948	F	0	0	Portugal	Lisbon
9	999	Anne	05/09/1992	F	0	5	Switzerland	Geneva
10	101010	Paul	14/11/1992	M	1	26	Ytali	Rome

Valor faltante

Valor inválido

Valor que debería estar en otra columna

Error de ortografía

Item duplicado-Inválido

Formato incorrecto

Dependencia de atributos

Conversión de datos

- ❑ Después de ser preprocesados, los datos deben ser convertidos en una representación adecuada para el modelo de M.L. Las formas más comunes de conversión de datos incluyen los siguientes:
- **Codificación:** Con respecto a la clasificación, los datos de las categorías se codifican en una representación numérica correspondiente, ej. *one hot encoding* o *codificación binaria*.
 - Los datos numéricos se pueden convertir en datos de tipo categórico, ej. para generar segmentos etarios, categorías de alarma de acuerdo a variables, etc.

Ej. one hot encoding

Color	Red	Yellow	Green
Red	1	0	0
Red	1	0	0
Yellow	0	1	0
Green	0	0	1
Yellow	0	1	0

Conversión de datos

❑ Otros datos

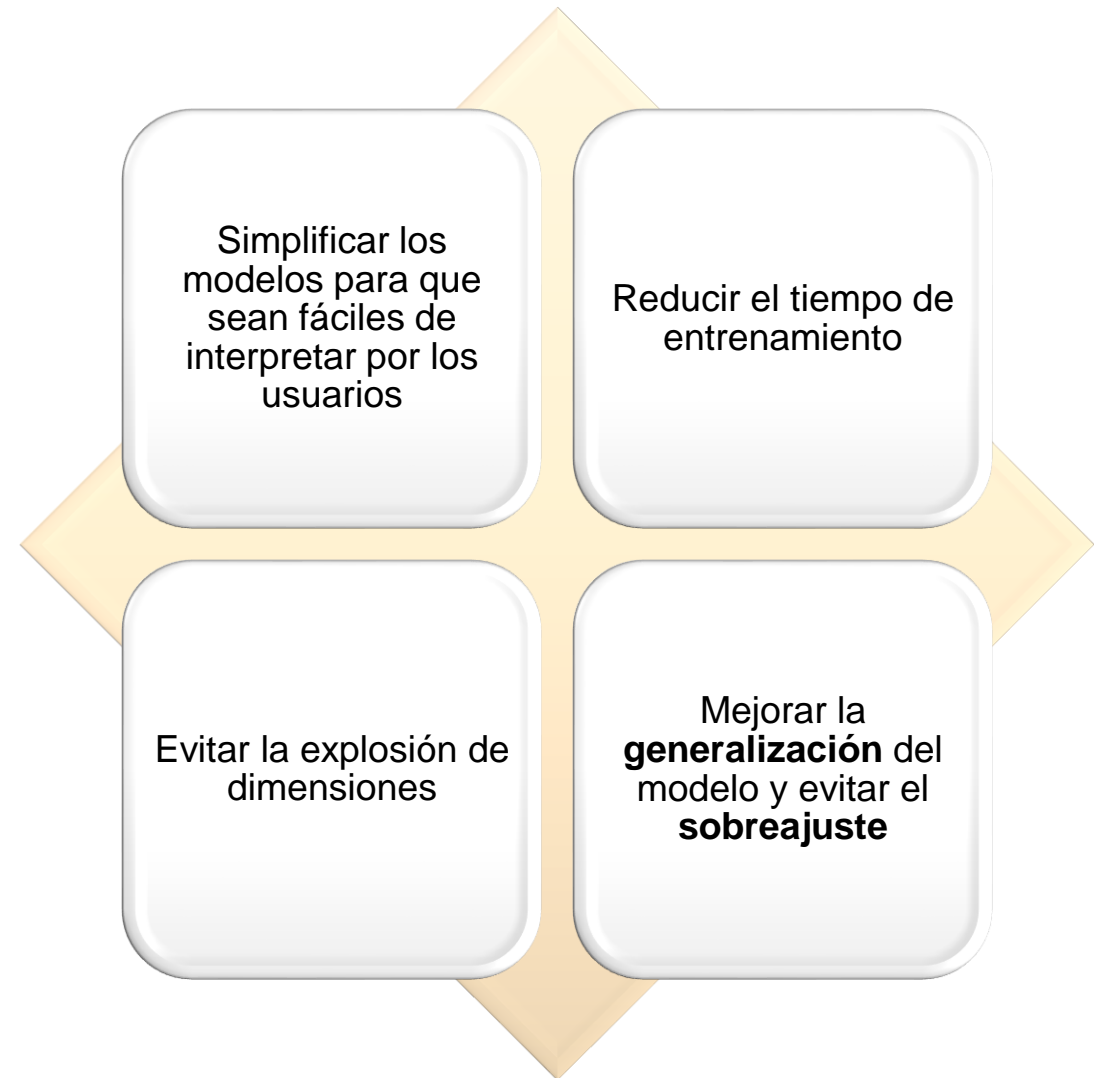
- **En texto**, la palabra se convierte en un vector numérico a través de vectorizaciones de palabras/**Word embeddings** (generalmente usando el modelo **word2vec**, el modelo **BERT**, etc.)
- **Procesar imágenes**: espacio de color, escala de grises, cambio geométrico, función Haar, mejora de imagen, etc.

❑ Ingeniería de características (*feature engineering*)

- **Normalizar** características para garantizar los mismos rangos de valores para las variables de entrada del mismo modelo.
- **Expansión de características** : Combinar o convertir variables existentes para generar nuevas características, como el promedio, agregar transformaciones polinómicas, etc.

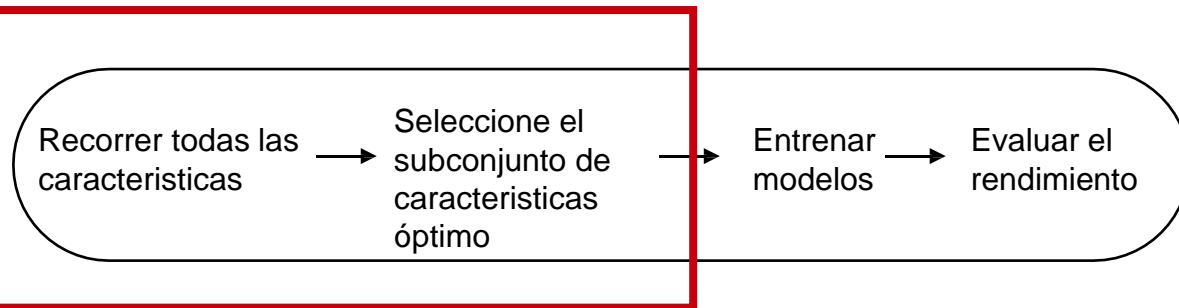
Necesidad de la selección de características

- ❑ Generalmente, un conjunto de datos tiene muchas características, algunas de las cuales pueden ser redundantes o irrelevantes para el valor a predecir.
- ❑ La selección de características es necesaria en los siguientes aspectos:



Métodos de selección de características - Filtro

- ❑ Los métodos de filtrado son independientes del modelo durante la selección de características.



Procedimiento de un método de filtro

Al evaluar la correlación entre cada característica y el atributo de destino, **estos métodos utilizan una medida estadística para asignar un valor a cada característica**. Luego, las características se clasifican por puntuación, lo que ayuda a preservar o eliminar características específicas.

Métodos comunes

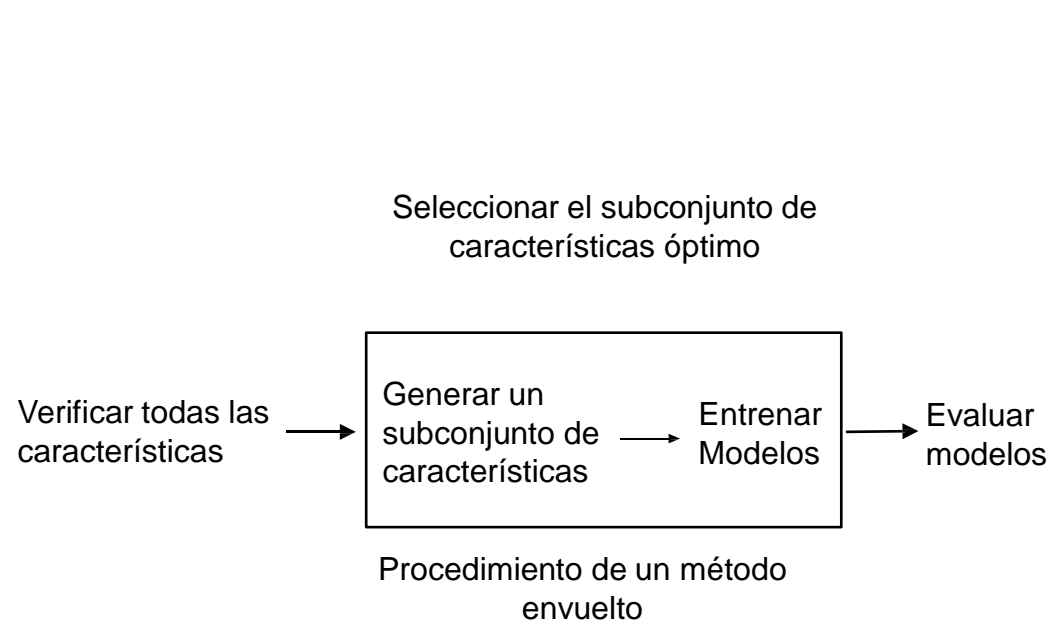
- Coeficiente de correlación de Pearson
- Coeficiente de chi-cuadrado
- Información mutua

Limitaciones

- El método de filtrado tiende a seleccionar variables redundantes ya que no se tiene en cuenta la relación entre las características.

Métodos de selección de características – Wrapper (envueltos)

- ❑ Los métodos envueltos utilizan un modelo de predicción para puntuar subconjuntos de características.



Los métodos envueltos consideran la **selección de características como un problema de búsqueda para el cual se evalúan y comparan diferentes combinaciones**. Un modelo predictivo se utiliza para evaluar una combinación de características y asignar una puntuación basada en alguna métrica de desempeño del modelo.

Métodos comunes

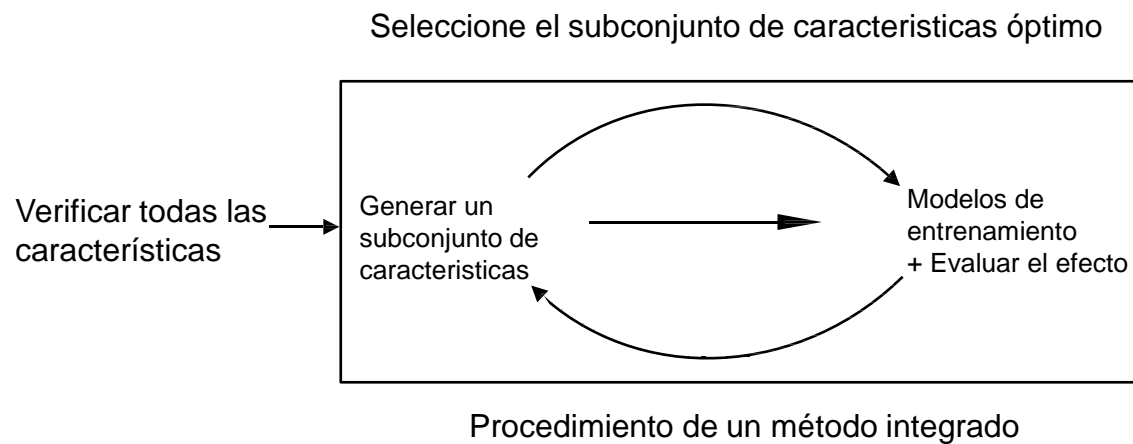
- Eliminación de funciones recursivas (RFE)

Limitaciones

- Los métodos envueltos entrenan un nuevo modelo para cada subconjunto, lo que da lugar a un **gran número de cálculos**.
- Normalmente un conjunto de características puede dar el mejor rendimiento para un tipo específico de modelo, pero para otro podría ser mejor otro subconjunto.

Métodos de selección de características – Integrados (Embedded)

- ❑ Los métodos integrados consideran la selección de características como parte de la construcción de modelos.



El método de selección de características incorporadas más común es el **método de regularización**.

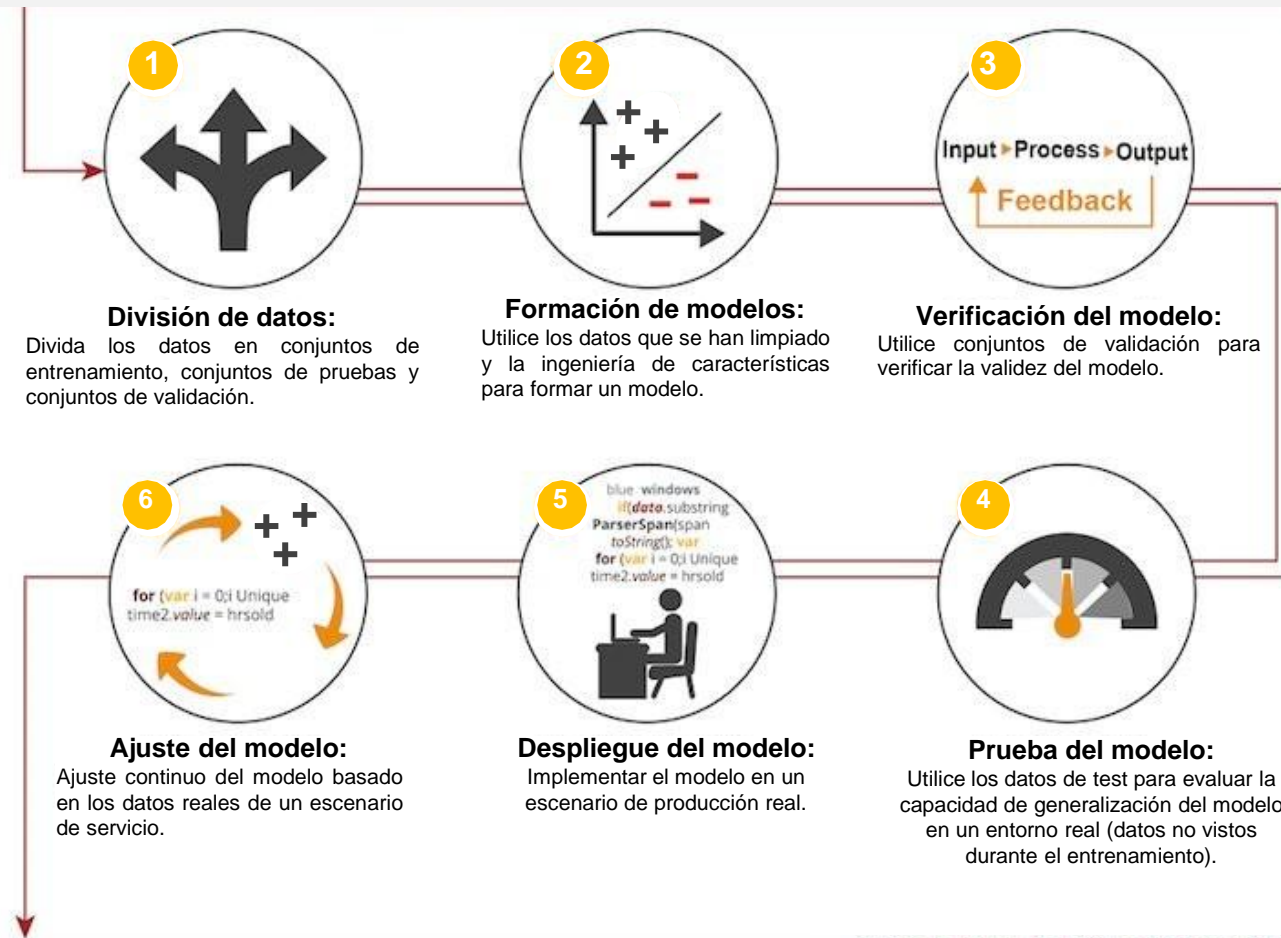
Los métodos de regularización también se llaman métodos de penalización que introducen restricciones adicionales en la optimización de un algoritmo predictivo, que **presiona al modelo hacia una menor complejidad y reducen el número de características**.

Métodos comunes

- Regresión Lasso
- Regresión Ridge

Procedimiento general para la construcción de un modelo

Procedimiento de construcción del modelo



Qué es un buen modelo



- **Capacidad de generalización**

¿Puede predecir con precisión los datos reales del servicio?

- **Interpretabilidad**

¿Es fácil interpretar el resultado de la predicción?

- **Velocidad de predicción**

¿Cuánto tiempo se tarda en predecir cada pieza de datos?

- **Practicidad**

¿La velocidad de predicción sigue siendo aceptable cuando el volumen de servicio aumenta con un gran volumen de datos?

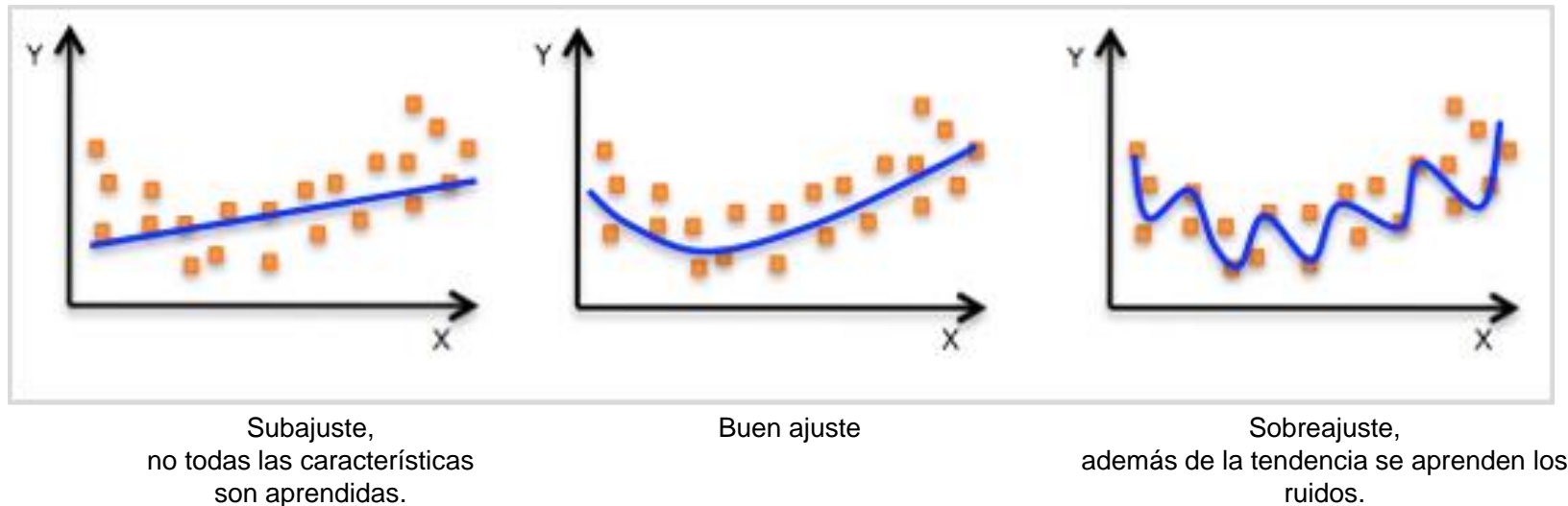
Validez del Modelo (1)

- ❑ **Capacidad de generalización:** el objetivo de Machine Learning es que el modelo obtenido después del aprendizaje tenga un buen rendimiento en nuevas muestras de datos, no sólo en las muestras utilizadas para el entrenamiento. La capacidad de aplicar un modelo a nuevas muestras de datos se denomina **generalización o robustez**.
- ❑ **Error:** diferencia entre el resultado predicho por el modelo obtenido después del aprendizaje, y el resultado de la muestra real.
 - **Error de entrenamiento:** error que se obtiene al ejecutar el modelo en los datos de entrenamiento.
 - **Error de generalización:** error que se obtiene al ejecutar el modelo en nuevas muestras. Obviamente, preferimos un modelo con un error de generalización menor.
- ❑ **Subajuste (underfitting):** se produce cuando el modelo o el algoritmo no se ajusta a los datos lo suficientemente bien.
- ❑ **Sobreajuste (overfitting):** se produce cuando el error obtenido al usar modelo entrenado en los mismos datos de entrenamiento es pequeño pero el error de generalización es grande (capacidad de generalización deficiente).

Generalizar \neq Memorizar

Validez del Modelo (2)

- **Capacidad del modelo:** se refiere a la capacidad del modelo de ajustar funciones, que también se denomina complejidad del modelo.
- Cuando la capacidad se adapta a la complejidad de la tarea y a la cantidad de datos de entrenamiento proporcionados, el efecto de algoritmo suele ser **óptimo**.
 - Los modelos con capacidad insuficiente no pueden resolver tareas complejas y pueden producirse **subajustes (underfitting)**
 - Un modelo de alta capacidad puede resolver tareas complejas, pero puede producirse un **sobreajuste (overfitting)** si la capacidad es mayor que la requerida por una tarea.



Sobreajuste: descripción gráfica
[\[fuente\]](#)

En qué se mide la capacidad?

Errores involucrados en la predicción

❑ **Error total de predicción final = $\text{Bias}^2 + \text{Varianza} + \text{Error Irreducible}$**

❑ Generalmente, el error de predicción se puede dividir en dos tipos:

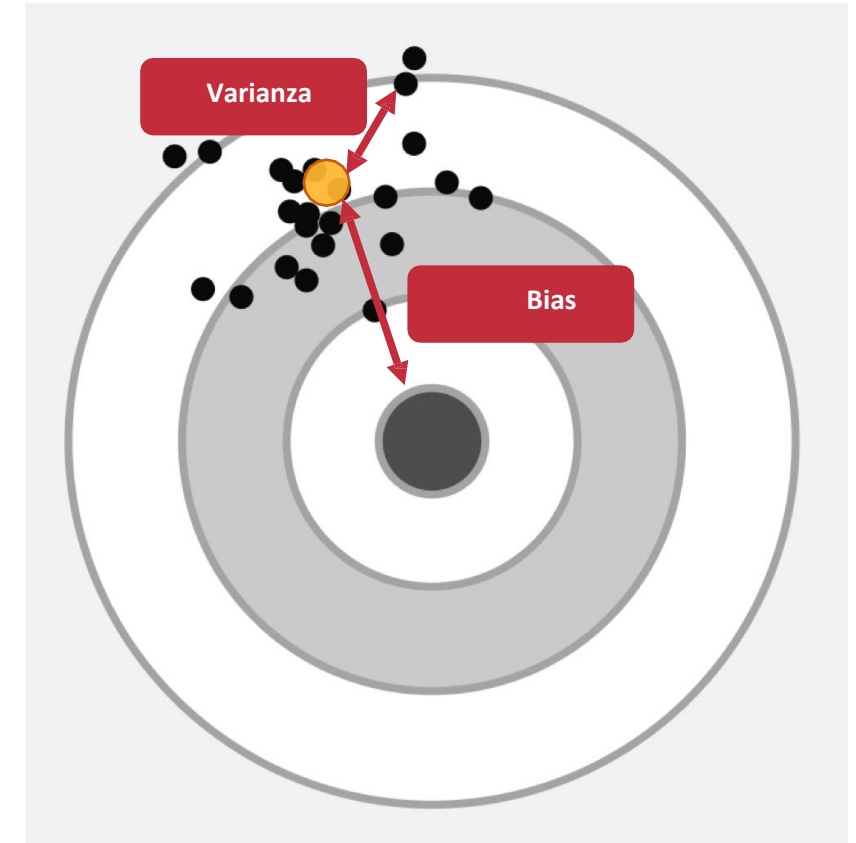
- Error causado por "bias" (sesgo)
- Error causado por "variancia"

❑ **Varianza del modelo:**

- Desfase del resultado de la predicción a partir del valor promedio
- Error causado por la sensibilidad del modelo a pequeñas fluctuaciones en el conjunto de entrenamiento

❑ **Bias (sesgo) del modelo:**

- Diferencia entre el valor predicho (o promedio) por el modelo y el valor correcto que estamos intentando predecir.

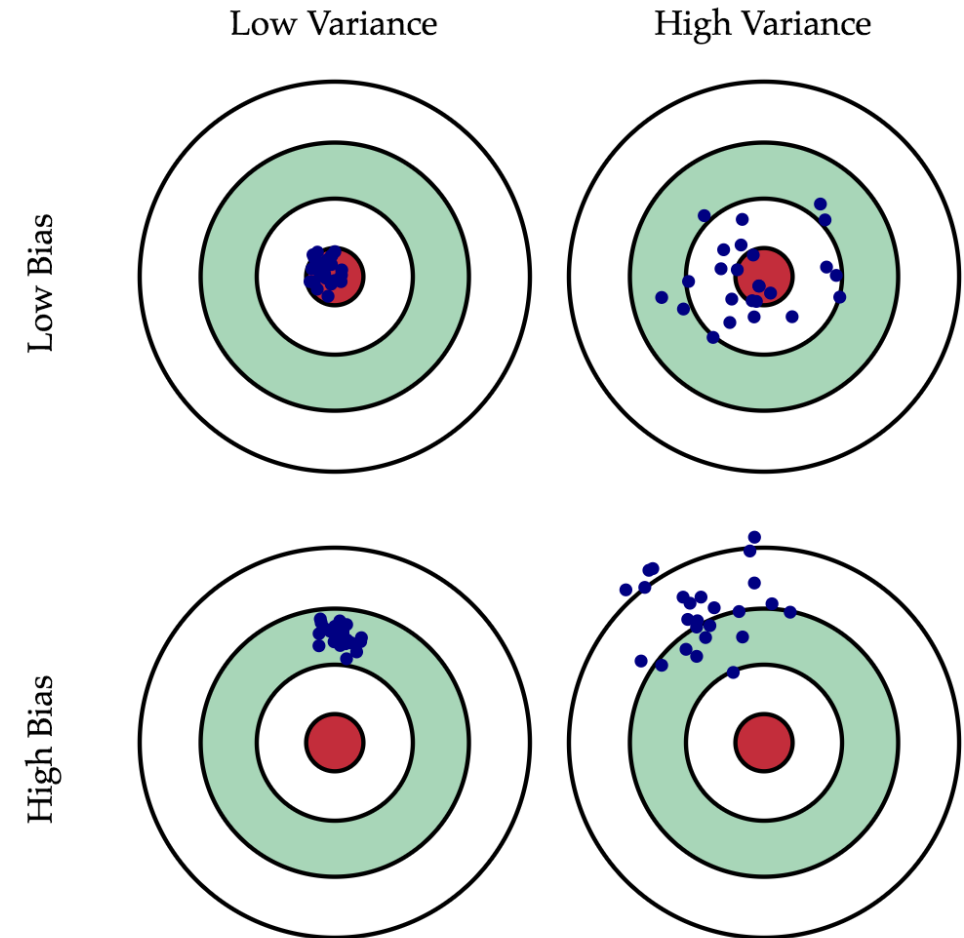


Compromiso entre Varianza y Bias (sesgo)

□ Las combinaciones de varianza y sesgo son las siguientes:

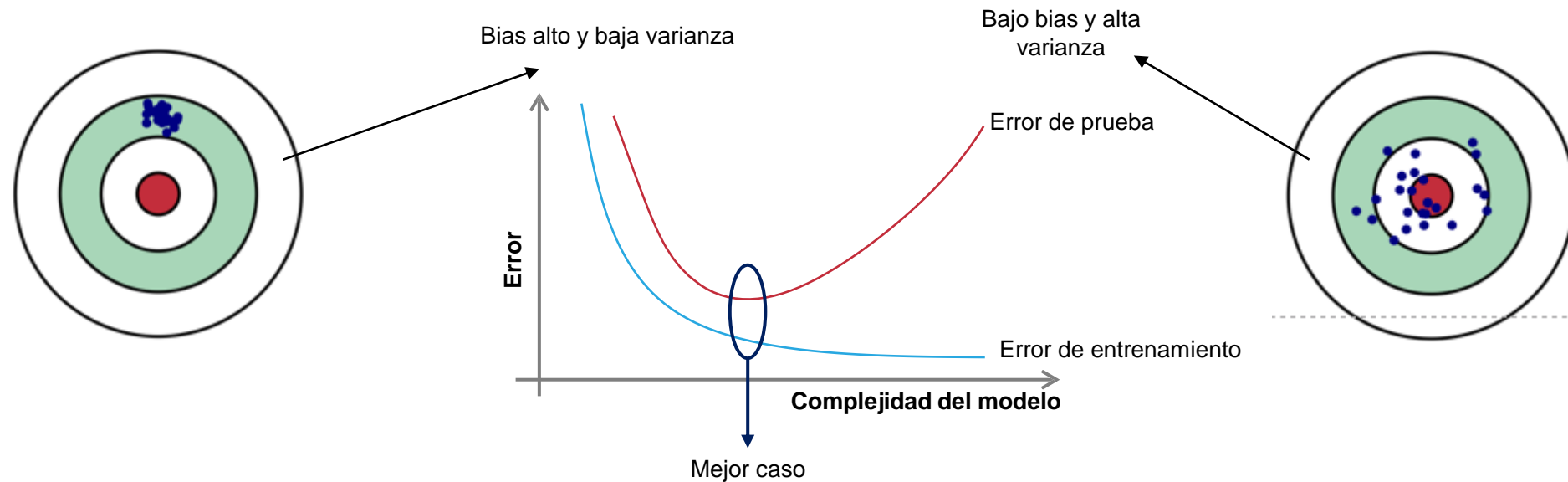
- **Bajo sesgo y baja varianza → Buen modelo!**
- Bajo sesgo y alta varianza
- Sesgo alto y baja varianzas
- **Sesgo alto y alta varianza → Mal modelo!**

□ Idealmente, queremos un modelo que pueda capturar con precisión las reglas en los datos de entrenamiento y resumir los datos invisibles (nuevos datos). Pero por lo general es imposible para el modelo completar ambas tareas al mismo tiempo.

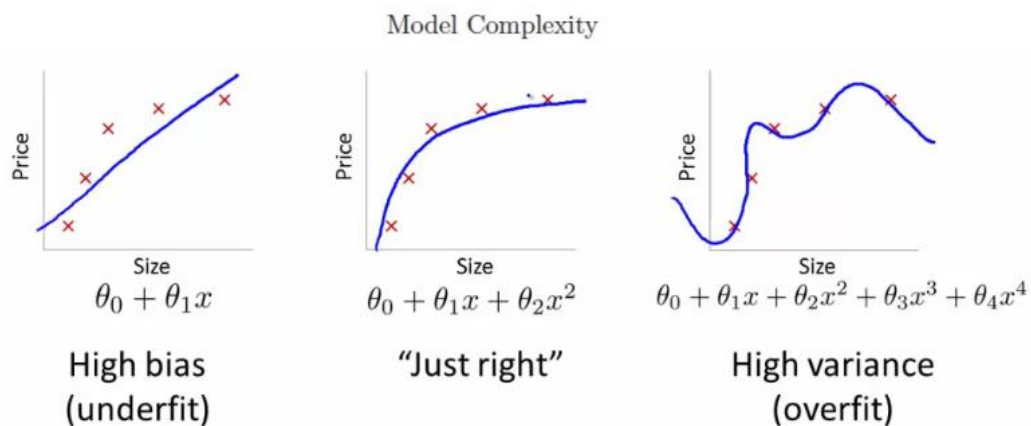
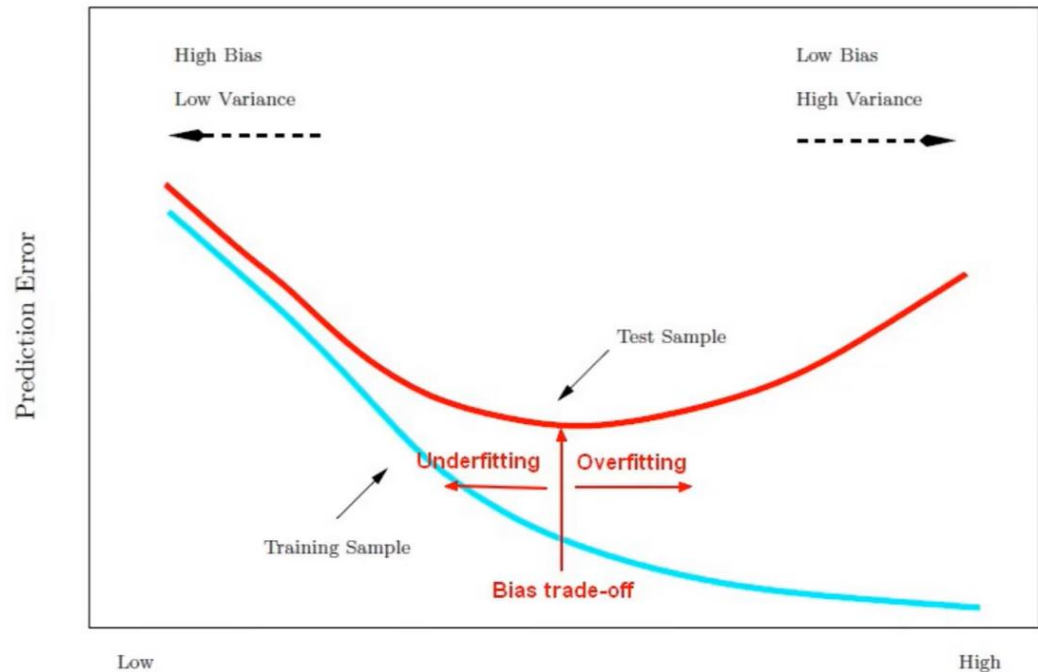


Complejidad y error del modelo

- ❑ A medida que aumenta la complejidad del modelo, disminuye el error de entrenamiento.
- ❑ A medida que aumenta la complejidad del modelo, el error de prueba disminuye a un punto determinado y luego aumenta en la dirección inversa, formando una curva convexa.



Complejidad y error del modelo



Conclusiones:

A. Para reducir el error de bias

- Usar un modelo más complejo
- Agregar más variables o predictores
- Incrementar las iteraciones de entrenamiento
- Cambiar el modelo

B. Para reducir el error de varianza del modelo

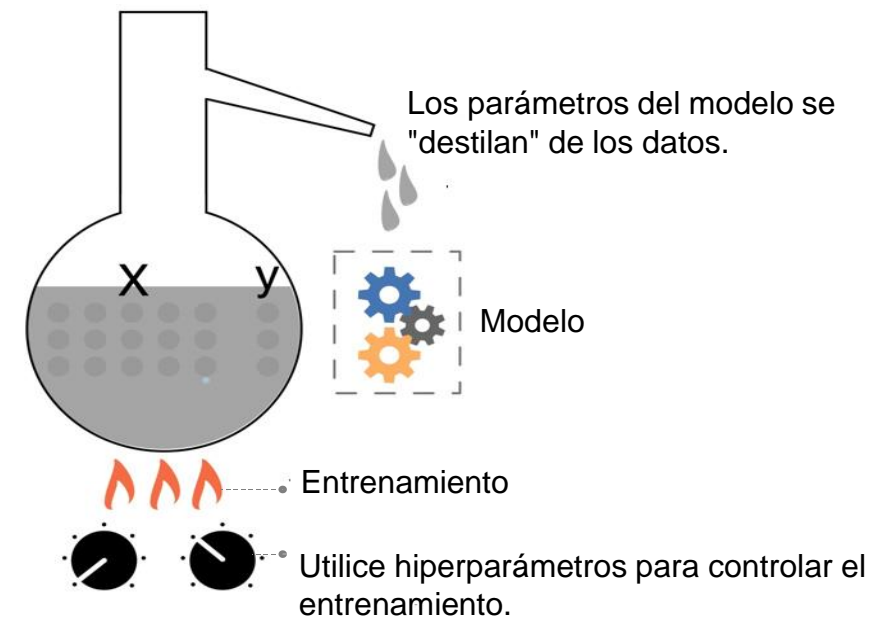
- Agregar mas datos de entrenamiento
- Reducir la complejidad del modelo
- Reducir el número de variables o predictores (ej. usando PCA)
- Cambiar el modelo

El sesgo y la varianza relativos al modelo son complementarios, el incremento de uno resulta en el decrecimiento del otro y viceversa. Se debe encontrar el balance entre ambos.

Otros métodos clave de *Machine Learning*

Parámetros e Hiperparámetros en Modelos

- ❑ El modelo contiene no sólo parámetros sino también hiperparámetros. El objetivo es permitir que el modelo aprenda los parámetros óptimos.
 - Los parámetros se aprenden automáticamente por los modelos.
 - Los hiperparámetros se configuran manualmente.



Hiperparámetros en Modelos

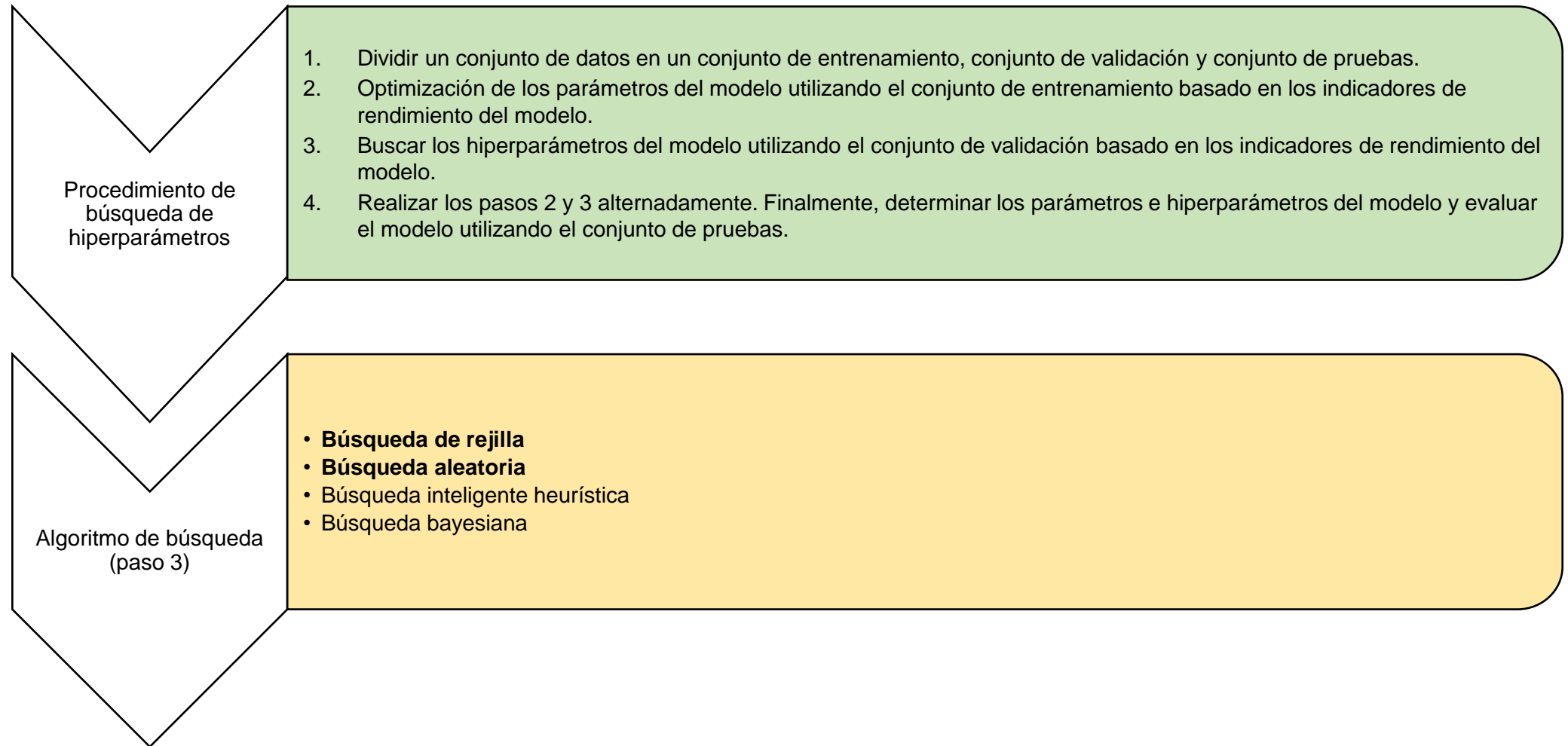
- Con frecuencia se utilizan en los procesos de estimación de parámetros del modelo.
- A menudo especificados por el practicante.
- A menudo se puede configurar usando heurística.
- A menudo sintonizados para un problema de modelado predictivo dado.

Los hiperparámetros de los modelos son configuraciones externas de los modelos.

- λ durante la regression LASSO/RIDGE
- Tasa de aprendizaje para entrenar una red neuronal, número de iteraciones, tamaño de lote, función de activación, número de neuronas, número de capas, tipo de capas.
- C y σ en máquinas de vectores de soporte (Support vector Machine – SVM).
- K en k-vecino más cercano (K in K-Nearest Neighbor KNN)
- Número de árboles en un bosque aleatorio

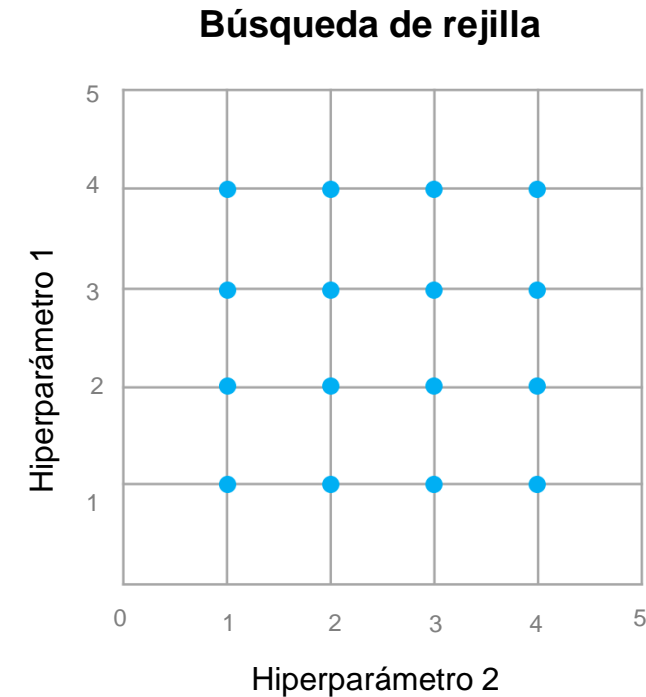
Hiperparámetros comunes de algunos modelos

Procedimiento y método de búsqueda de hiperparámetros



Método de búsqueda de hiperparámetros: Búsqueda de rejilla

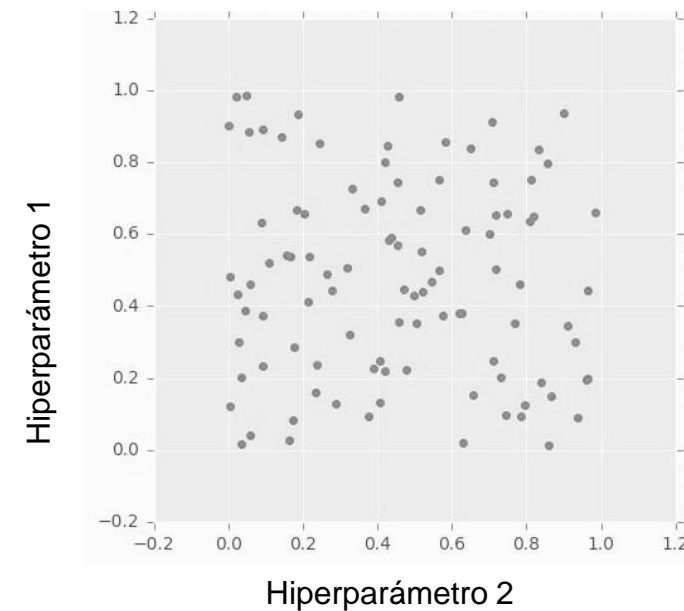
- ❑ La búsqueda de rejilla (**grid or exhaustive search**) intenta **buscar exhaustivamente todas las combinaciones de hiperparámetros** posibles para formar una rejilla de valores de hiperparámetros.
- ❑ En la práctica, el rango de valores de hiperparámetros a buscar se especifica manualmente.
- ❑ La búsqueda de rejilla es un método costoso y que requiere mucho tiempo.
 - Este método funciona bien cuando el número de hiperparámetros es relativamente pequeño. Por lo tanto, se aplica generalmente a los algoritmos de M.L., pero no a las redes neuronales.



Método de búsqueda de hiperparámetros: Búsqueda aleatoria

- ❑ Cuando el espacio de búsqueda de hiperparámetros es grande, la **búsqueda aleatoria (random search)** es mejor (más rápido) que la búsqueda de rejilla.
- ❑ En la búsqueda aleatoria, cada configuración se muestrea de la distribución de los posibles valores de parámetros, en un intento de encontrar el mejor subconjunto de hiperparámetros.
- ❑ **Notas:**
 - La búsqueda se realiza dentro de un rango amplio, que se reducirá en función del lugar donde aparece el mejor resultado.
 - Algunos hiperparámetros son más importantes que otros, y la desviación de búsqueda se verá afectada durante la búsqueda aleatoria.

Búsqueda aleatoria



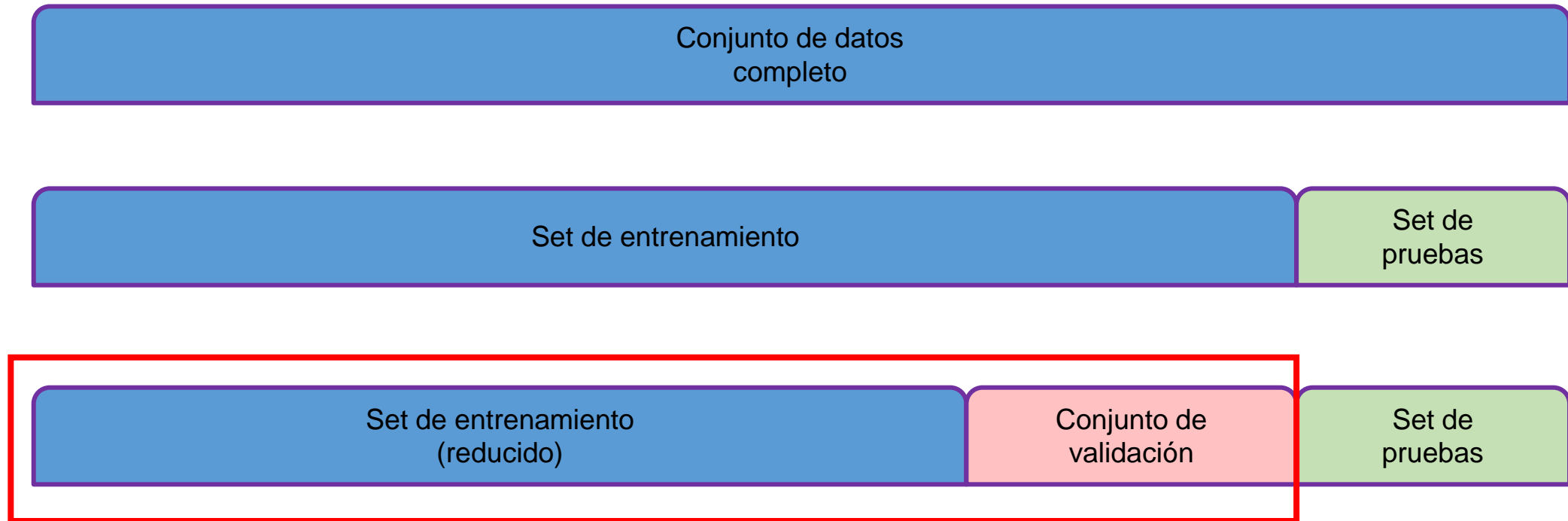
Validación cruzada (1)

❑ **Cross validation (Validación Cruzada):** Es un método de análisis estadístico que se utiliza para validar el desempeño de un algoritmo de M.L. **La idea básica es dividir el conjunto de entrenamiento original en dos partes: conjunto de entrenamiento y conjunto de validación.** Entrene el modelo usando el conjunto de entrenamiento y pruebe el modelo usando el conjunto de validación para verificar el rendimiento.

❑ **k-fold cross validation ($K - CV$):**

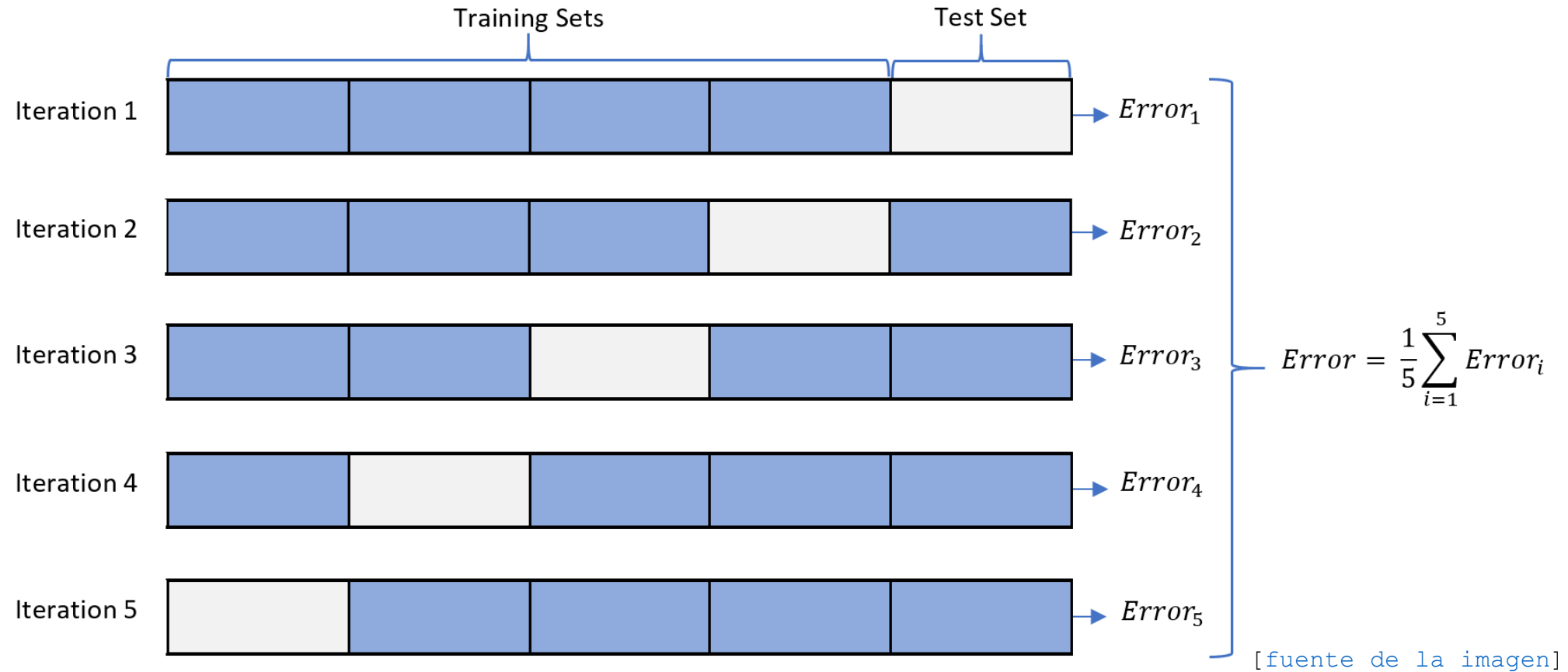
- Divida los datos brutos en k grupos (generalmente divididos de manera uniforme).
- Use cada subconjunto como un conjunto de validación y use los otros subconjuntos $k - 1$ como el conjunto de entrenamiento. Se pueden obtener un total de k modelos.
- Utilice la métrica de desempeño media de los conjuntos de validación final de los modelos k como indicador de rendimiento del modelo $K - CV$.

Validación cruzada (2)



Nota: El valor K en K-CV es también un hiperparámetro.

Validación cruzada (3)



Ejemplo k-fold cross validation, dividiendo el conjunto de entrenamiento en 5 subconjuntos de datos, y realizando 5 entrenamientos y pruebas distintas para entregar una métrica de desempeño más realista para el entrenamiento. Finalmente, el modelo se entrena en todos los datos, pero la métrica reportada es la que sale de este procedimiento.

Referencias

[1] T. Karras, S. Laine, T. Aila, A Style-Based Architecture for Generative Adversarial Networks, arXiv:1812.04948, 2019.

[2] Apuntes Curso: Convolutional Neural Networks for Visual Recognition, Stanford University, Disponible Online: [link](#)

- Parte del material de este curso esta basado en:
 - Certificación Huawei HCIA-AI, [enlace](#).
 - Documentación oficial:
 - Matlab, <https://la.mathworks.com/>
 - Numpy, <https://numpy.org/>
 - Pandas, <https://pandas.pydata.org/>
 - Matplotlib, <https://matplotlib.org/>
 - Scikit Learn, <https://scikit-learn.org/stable/>
 - Tensorflow, <https://www.tensorflow.org/>
 - Keras, <https://keras.io/>
 - Sci-kit Learn MOOC, <https://www.inria.fr/en/mooc-scikit-learn>