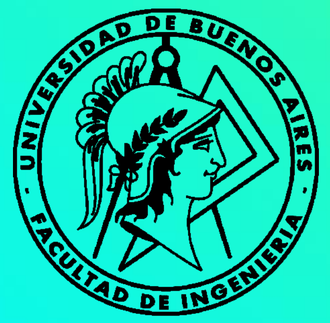


INTRODUCCIÓN

APRENDIZAJE DE MAQUINA - CEIA - FIUBA

Dr. Ing. Facundo Adrián Lucianna

Esp. Lic. María Carina Roldán

INTRODUCCIÓN

Materia de ocho clases teórico-prácticas.

Las clases se impartirán con diapositivas y desarrollo en notebooks.

Estructuras de las clases:

10 minutos de repaso de la clase anterior.

Tres bloques de 50 minutos de contenido teórico-práctico.

Dos recreos de 10 minutos.

INTRODUCCIÓN

Aula virtual

- https://campusposgrado.fi.uba.ar/course/view.php?id=251

Repositorio de la materia:

- https://github.com/FIUBA-Posgrado-Inteligencia-Artificial/aprMaqI_CEIA

Consultas

- Foro de consulta en el aula virtual

Correo (consultas generales envíen con copia a los dos)

- Facundo Adrián Lucianna: facundolucianna@gmail.com
- María Carina Roldán: macroldan@fi.uba.ar

EVALUACIÓN

El trabajo práctico final deberá implementarse y entregarse dentro de los **tres (3) días posteriores a la última clase.**

El trabajo deberá realizarse en grupos de un mínimo de dos (2) y un máximo de seis (6) integrantes. Se podrán realizar excepciones únicamente con un justificativo debidamente fundamentado.

La consigna consiste en la implementación de un modelo de Machine Learning sobre un conjunto de datos a elección.

El trabajo podrá entregarse en cualquier formato, preferentemente en un notebook de Jupyter (.ipynb). También se aceptará su presentación mediante Google Colab o un enlace a un repositorio.

El enfoque principal de este trabajo es la etapa de Machine Learning: la aplicación, manejo y evaluación de modelos. Si bien el procesamiento de datos y el feature engineering son pasos esenciales para la aplicación del modelo, en esta materia tendrán menor relevancia.

Se deberá justificar la elección de las herramientas utilizadas y presentar un desarrollo teórico correcto. No olvidar incluir las referencias bibliográficas correspondientes.

DATASETS

¿No tienen algún dataset?

- Kaggle: https://www.kaggle.com/datasets
- Awesome public datasets: https://github.com/awesomedata/awesome-public-datasets
- Real world fake data: https://sonsofhierarchies.com/real-world-fake-data/
- Maven analytics: https://www.mavenanalytics.io/data-playground

Si siguen sin encontrar un dataset, les dejamos algunos en el aula virtual.

PROGRAMA

Clase a clase

01 Introducción a Machine Learning. Definiciones. Ciclo de vida de un proyecto de Aprendizaje Automático. Metodología para construir modelos.

02 Aprendizaje supervisado. Clasificador KNN. Medidas de distancia. Encontrando a los vecinos. Métodos de Ajuste de los hiper-parámetros. Frameworks de búsqueda

03 Máquinas de vectores de soporte. Maximal Margin Classifier. Clasificador de Vector de soportes. Máquinas de vectores de soporte en regresión.

04 Arboles de decisión. Arboles de clasificación y arboles regresión.

PROGRAMA

Clase a clase

05	Métodos de ensamble. Modelos que votan. Bagging. Bosques aleatorios. Boosting.
06	Calibración de modelos. Discriminación vs. Calibración. Métodos de Calibración.
07	Aprendizaje No Supervisado. Clustering. K-Means. Suma de Cuadrados Intracluster. Índice de la silueta. Fuerza de Predicción. Hierarchical clustering. Modelo de Mixtura Gaussiana.

BIBLIOGRAFIA

- Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python - Peter Bruce (Ed. O'Reilly)
- The Elements of Statistical Learning - Trevor Hastie (Ed. Springer)
- An Introduction to Statistical Learning - Gareth James (Ed. Springer)
- Pattern Recognition And Machine Learning - Christopher Bishop (Ed. Springer)
- Deep Learning - Ian Goodfellow https://www.deeplearningbook.org/

APRENDIZAJE AUTOMÁTICO

APRENDIZAJE AUTOMÁTICO

Tal como se aprendió en Inteligencia Artificial, **Aprendizaje Automático** se entiende como:

Una computadora que observa algunos datos, construye un modelo basado en ellos y utiliza ese modelo como una hipótesis sobre el mundo, además de como una pieza de software capaz de resolver problemas.

- Recibe un conjunto de datos y aprende por sí misma.
- Reconoce patrones entre los datos y realiza predicciones.
- No requiere que una persona programe instrucciones de forma explícita.

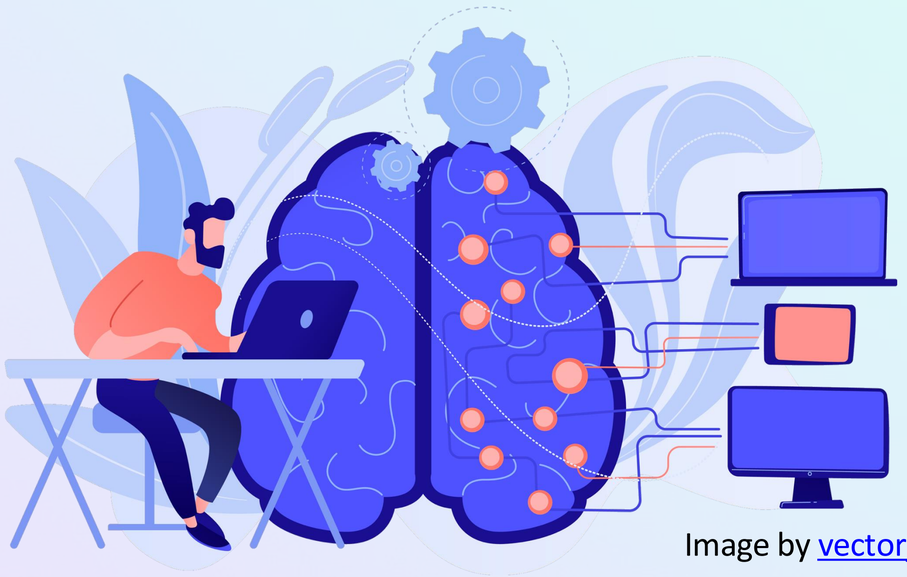


Image by vectorjuice

¿CUÁNDO USARLO?

¿CUÁNDO USARLO?

El Aprendizaje Automático no es una herramienta mágica que pueda resolver todos los problemas. Incluso en los casos en que puede aplicarse, las soluciones obtenidas pueden no ser las óptimas.

Antes de iniciar un proyecto de Aprendizaje Automático, debemos evaluar si realmente es necesario o rentable:

- Para que un sistema de Machine Learning aprenda, debe haber algo de qué aprender: necesitamos una cantidad suficiente y adecuada de datos**.**
- Las soluciones basadas en Aprendizaje Automático solo son útiles cuando **existen patrones que puedan aprenderse**. Por ejemplo, no tiene sentido usar modelos para implementar un controlador PID.

  A veces no es evidente si realmente existe un patrón, o bien, aunque lo haya, el conjunto de datos o el algoritmo podrían no ser suficientes para capturarlo. Por ejemplo, *podría existir un patrón en cómo los tweets de Elon Musk afectan los precios de las criptomonedas*.

¿CUÁNDO USARLO?

- No solo se necesitan datos, sino que estos deben **estar disponibles o ser posibles de recopilar**. Sin datos y sin aprendizaje continuo, muchas empresas siguen un enfoque de *"fake-it- til-you make it"* ; es decir, lanzan un producto que ofrece **predicciones hechas por humanos** en lugar de modelos de Machine Learning, con la esperanza de utilizar los datos generados para entrenar posteriormente dichos modelos.

- Los modelos de Machine Learning hacen predicciones, por lo que solo **pueden resolver problemas que requieran respuestas predictivas**. Este enfoque puede resultar especialmente atractivo cuando se puede beneficiar de una gran cantidad de predicciones baratas, aunque aproximadas.

  Por ejemplo: *¿cómo será el clima mañana?, ¿quién ganará el Super Bowl este año?, ¿qué película querrá ver un usuario a continuación?*

- Los datos utilizados en el entrenamiento y los datos futuros sobre los que se aplicará el modelo deben provenir de distribuciones similares.

  ¿Cómo sabemos de qué distribución provienen? No lo sabemos con certeza, pero podemos hacer suposiciones y esperar que se cumplan. Si no es así, obtendremos un modelo con un rendimiento deficiente.

¿CUÁNDO USARLO?

- **Cuando una tarea es repetitiva**, cada patrón se repite varias veces, lo que facilita que las máquinas lo aprendan**.**
- Los patrones cambian constantemente, las culturas cambian, los gustos cambian y las tecnologías también. Lo que está de moda hoy puede ser una noticia vieja mañana.
- Consideremos la tarea de clasificar el correo no deseado: hoy, un indicio de spam puede ser un *príncipe nigeriano*, pero mañana podría ser un *escritor vietnamita angustiado.*
- Debido a que el Aprendizaje Automático aprende de los datos, **es posible actualizar los modelos con nueva información sin necesidad de redescubrir cómo han cambiado esos datos.**

¿CUÁNDO USARLO?

- Los modelos se equivocan, por lo que es necesario prestar especial atención a los problemas que pueden tener **consecuencias catastróficas**.

  Desarrollar vehículos autónomos es un desafío, ya que un error puede provocar la muerte. Aun así, muchas empresas continúan invirtiendo en su desarrollo, dado que estos vehículos tienen el potencial de salvar muchas vidas una vez que sean estadísticamente más seguros que los conductores humanos.

- Las soluciones de Aprendizaje Automático suelen requerir una inversión inicial considerable en datos, cómputo, infraestructura y talento, por lo que su aplicación **se justifica principalmente cuando el uso de dicha solución es frecuente o sostenido en el tiempo**.

¿CUÁNDO USARLO?

La mayoría de los algoritmos de Aprendizaje Automático **no** deberían utilizarse bajo ninguna de las siguientes condiciones:

- No es ético.
- Existen soluciones más simples que funcionan bien.
- No resulta rentable.

DEFINICIONES

TERMINOLOGIA BÁSICA

En Machine Learning generalmente se utilizan arrays 2D y notaciones vectoriales para referirse a los datos, de la siguiente forma:

- Cada fila del array representa una **muestra, observación o dato puntual**.
- Cada columna corresponde a una **característica** (feature o atributo) de la observación mencionada en el punto anterior.
- En el caso más general, habrá una columna que llamaremos **objetivo, label, etiqueta o respuesta**, y que representará el valor que se pretende predecir.

TERMINOLOGIA BÁSICA

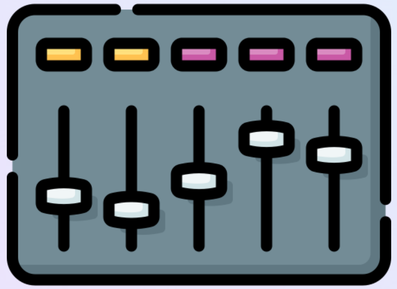
Observation →

Features					Label
Position	Experience	Skill	Country	City	Salary ($)
Developer	0	1	USA	New York	103100
Developer	1	1	USA	New York	104900
Developer	2	1	USA	New York	106800
Developer	3	1	USA	New York	108700
Developer	4	1	USA	New York	110400
Developer	5	1	USA	New York	112300
Developer	6	1	USA	New York	116100
Developer	7	1	USA	New York	117800

TERMINOLOGIA BÁSICA

Los algoritmos de Machine Learning tienen parámetros “internos” que no dependen directamente de los datos. A estos parámetros se los denomina hiperparámetros.

Por ejemplo, una red neuronal tiene como hiperparámetros la función de activación o la tasa de aprendizaje (learning rate).



Llamamos **generalización** a la capacidad de un modelo para realizar predicciones correctas utilizando datos nuevos.

Consola iconos creados por Freepik - Flaticon

TIPOS DE APRENDIZAJES

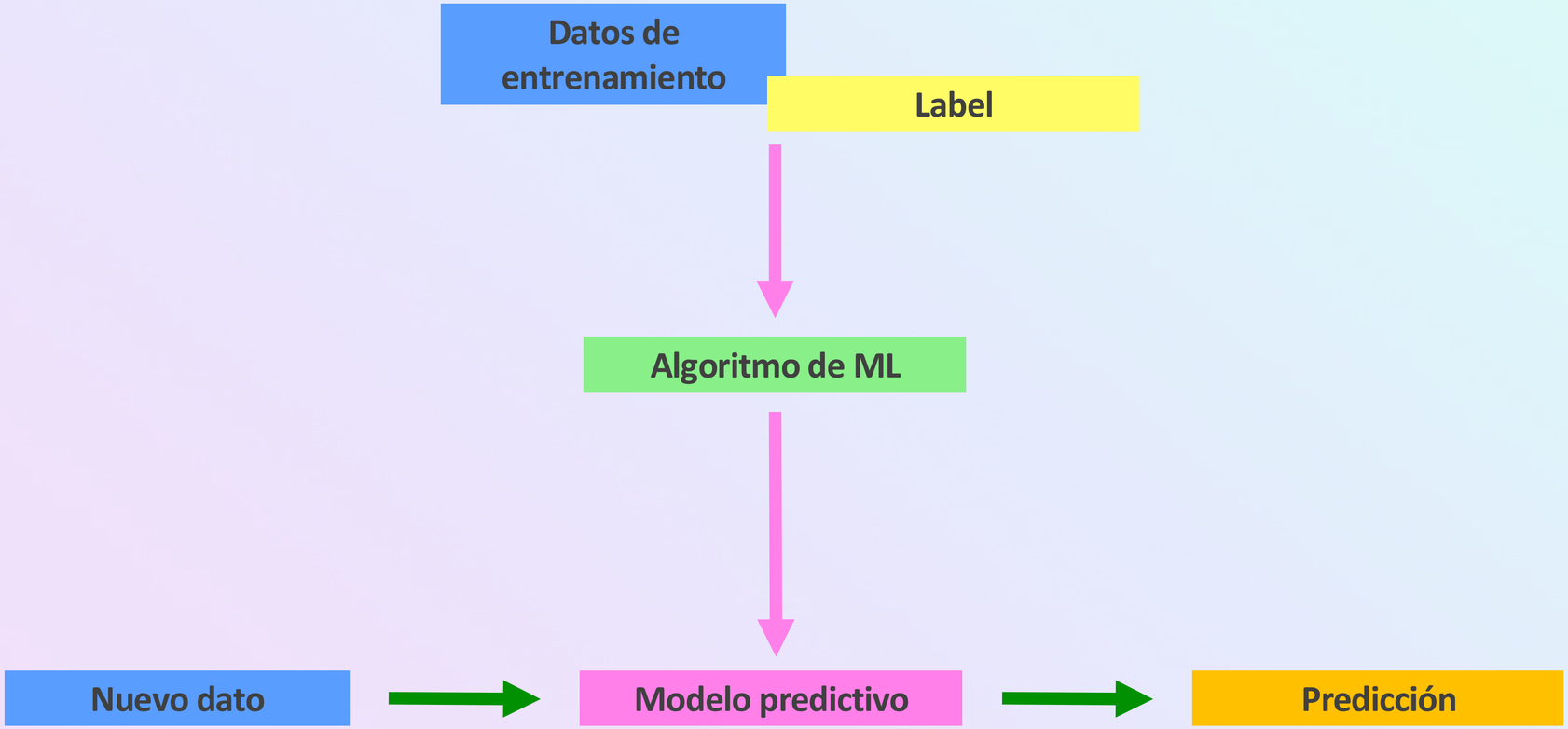
- **Aprendizaje supervisado**: Se refiere a un tipo de modelos de Machine Learning que se entrenan con un conjunto de ejemplos en los que los resultados de salida son conocidos.
- **Aprendizaje no supervisado**: Su objetivo es la extracción de información significativa sin la referencia de variables de salida conocidas, mediante la exploración de la estructura de los datos **no etiquetados**.
- **Aprendizaje profundo**: Es un subcampo de Machine Learning que utiliza una estructura jerárquica de redes neuronales artificiales, construidas de manera similar a la estructura neuronal del cerebro humano, con nodos de neuronas interconectadas.

APRENDIZAJE SUPERVISADO

- Los modelos aprenden de los resultados conocidos y realizan ajustes en sus parámetros internos para adaptarse a los datos de entrada.
- Una vez que el modelo ha sido entrenado adecuadamente y sus parámetros internos son coherentes con los datos de entrada y los resultados del conjunto de entrenamiento, el modelo podrá realizar predicciones precisas ante nuevos datos.

Image by vectorjuice

APRENDIZAJE SUPERVISADO

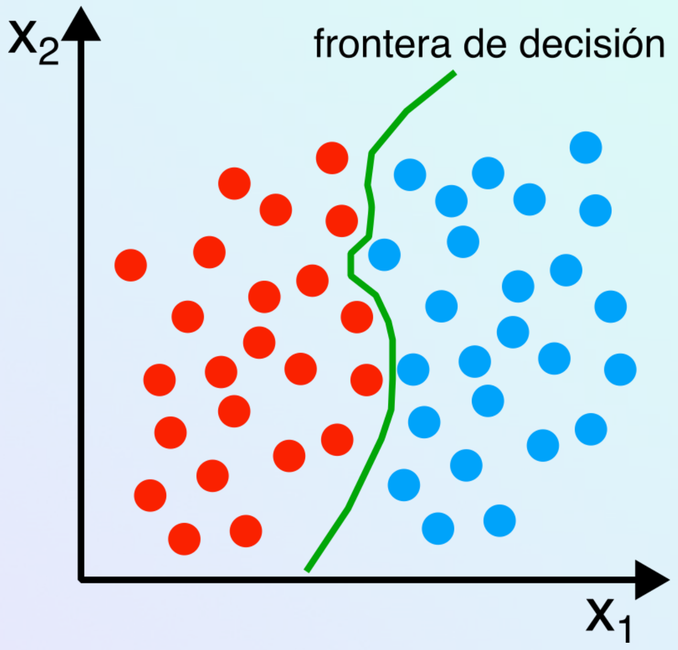


APRENDIZAJE SUPERVISADO

CLASIFICACIÓN

La clasificación es una subcategoría de aprendizaje supervisado en la que el objetivo es predecir clases categóricas (valores discretos, no ordenados, que indican pertenencia a grupos).

- Detección de SPAM: clasificación binaria (es spam o no es spam).
- Clasificación multiclase: múltiples clases, por ejemplo, la clasificación del nivel socioeconómico de una persona (alta, media o baja).

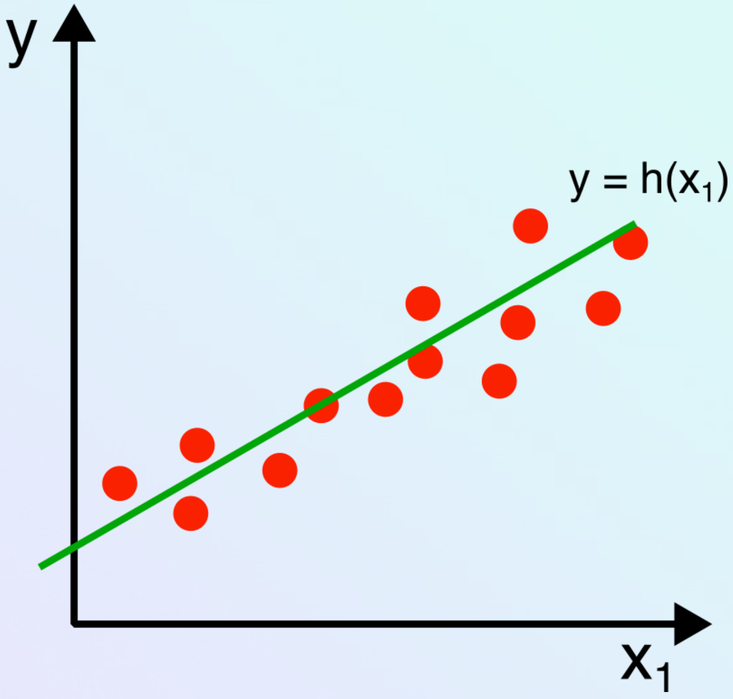


APRENDIZAJE SUPERVISADO

REGRESIÓN

En este tipo de aprendizaje tenemos un conjunto de variables predictoras (explicativas) y una variable de respuesta continua (resultado), y se tratará de encontrar una relación entre dichas variables que nos proporcione un **resultado continuo**.

- Regresión lineal: Dados X e y, se establece una **línea recta** que minimice la distancia entre los puntos de muestra y la línea ajustada. Luego, utilizaremos la fórmula obtenida de la regresión para predecir nuevos datos de salida.
- Un ejemplo típico es la regresión del precio de casas en venta en una ciudad, dado el barrio, la cantidad de habitaciones, etc.

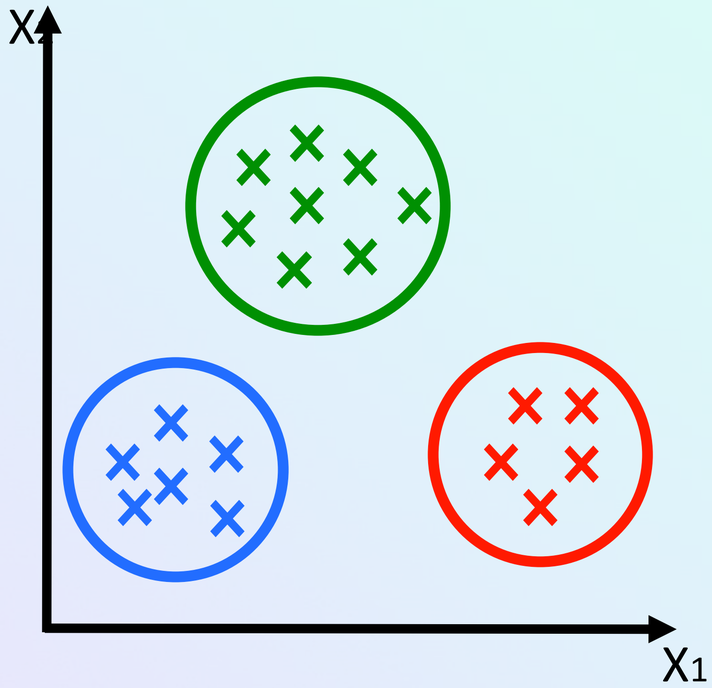


APRENDIZAJE NO SUPERVISADO

AGRUPAMIENTO O CLUSTERING

El agrupamiento es una técnica exploratoria de análisis de datos que se utiliza para organizar información en grupos con significado, sin tener conocimiento previo de su estructura.

- Cada grupo es un conjunto de objetos similares que se diferencia de los objetos de otros grupos.
- El objetivo es obtener un número de grupos con características similares.
- Un ejemplo de aplicación es analizar el comportamiento de múltiples cadenas de comida y determinar cómo se agrupan en el mercado para evaluar sus verdaderos competidores.

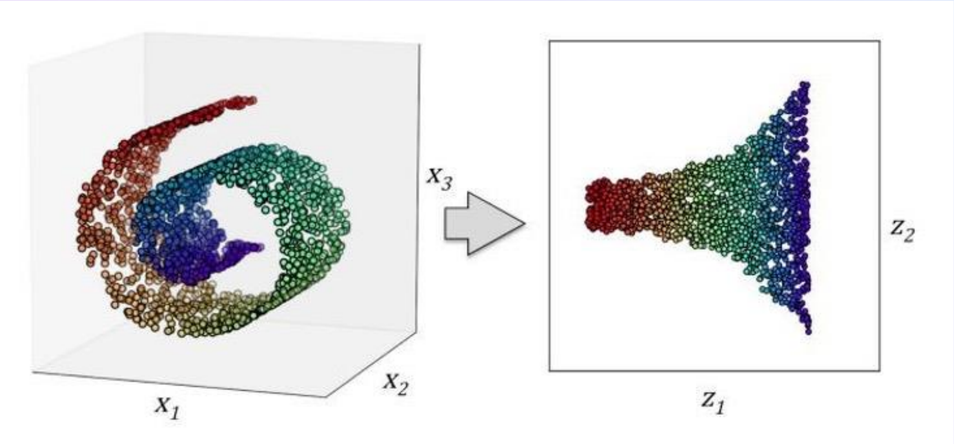


APRENDIZAJE NO SUPERVISADO

REDUCCIÓN DIMENSIONAL

La reducción dimensional funciona encontrando correlaciones entre las características, lo que implica que existe información redundante, ya que alguna característica puede explicarse parcialmente con otras (por ejemplo, puede existir dependencia lineal).

Estas técnicas **eliminan el ruido** de los datos (que también puede empeorar el desempeño del modelo) y **comprimen los datos** en un subespacio más reducido, **reteniendo la mayoría de la información relevante**.



APRENDIZAJE PROFUNDO

Esta arquitectura permite abordar el análisis de datos de forma **no lineal.**

La primera capa de la red neuronal toma datos en bruto como entrada, los procesa, extrae información y la transfiere a la siguiente capa como salida.

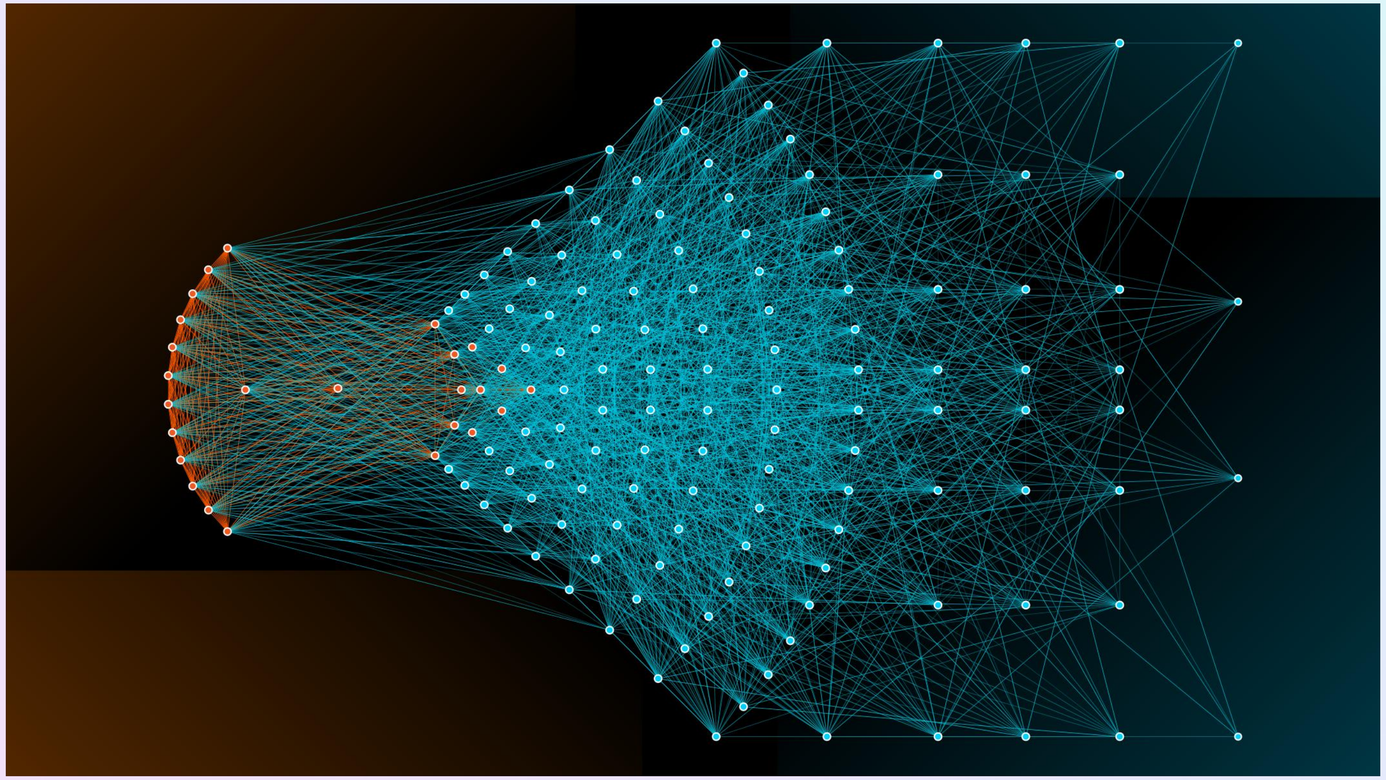
Este proceso se repite en las capas siguientes: cada capa procesa la información proporcionada por la capa anterior, y así sucesivamente hasta que los datos llegan a la capa final, donde se obtiene la predicción.

Esta predicción se compara con el resultado conocido, y mediante un **análisis inverso**, el modelo es capaz de aprender los factores que conducen a salidas adecuadas.

Es uno de los principales algoritmos utilizados en la creación de aplicaciones y programas para **reconocimiento de imágenes**.

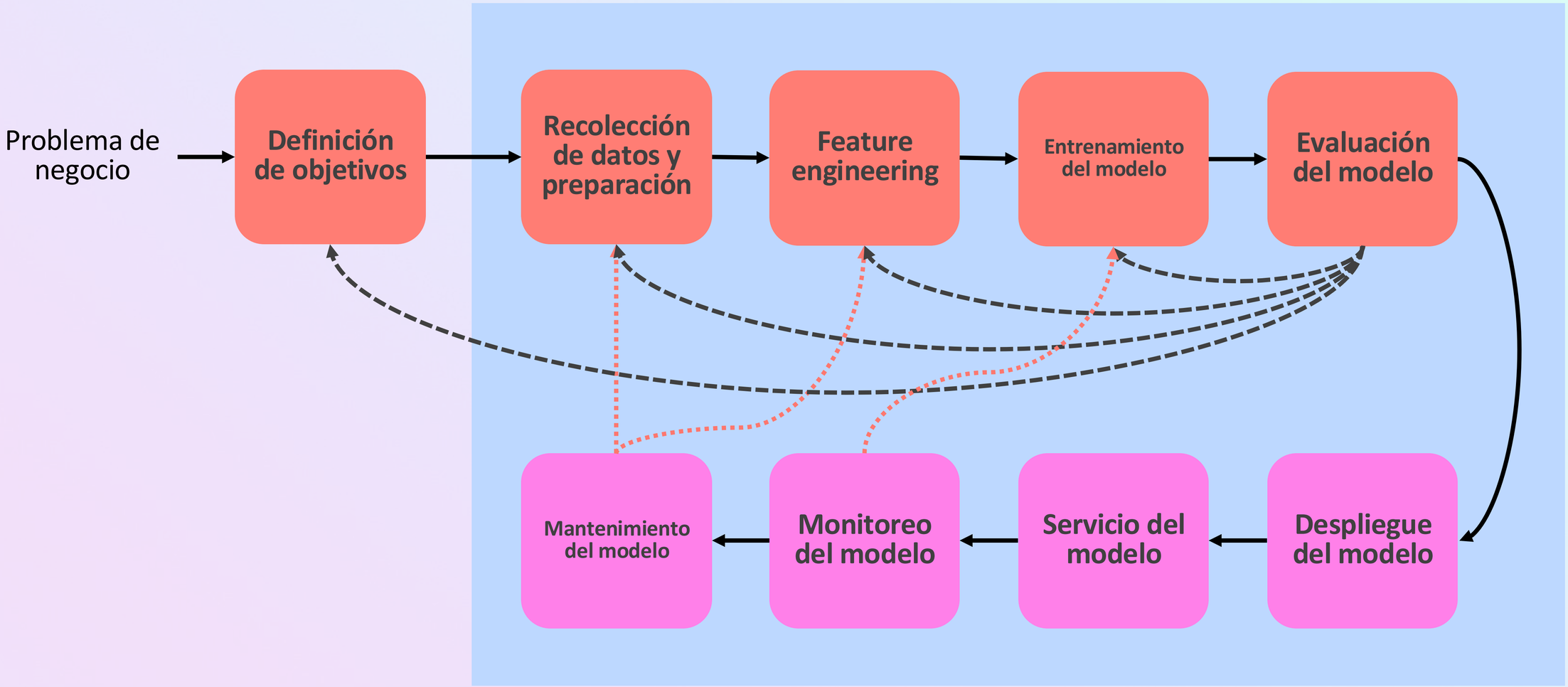
APRENDIZAJE PROFUNDO

Para obtener buenos resultados, necesitamos muchos datos, y en general el entrenamiento es costoso.

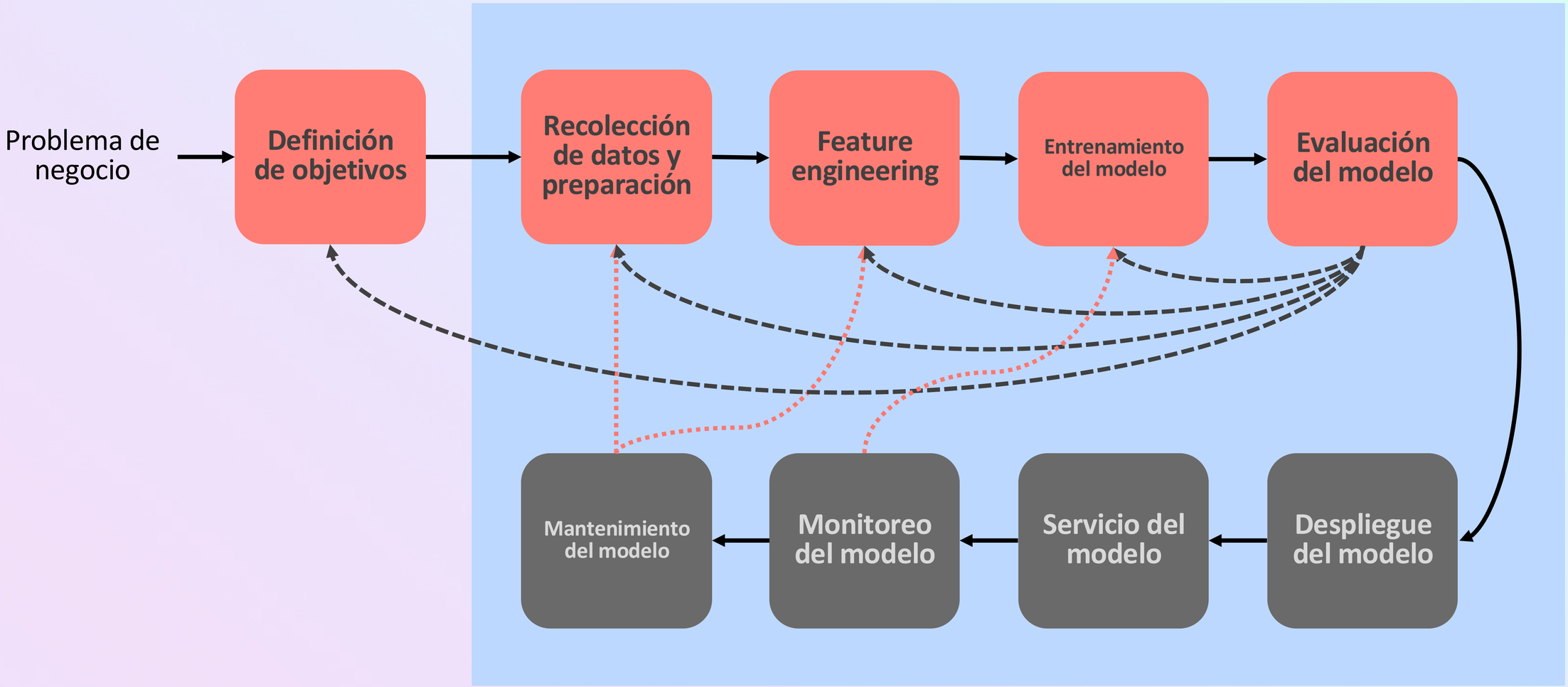


CICLO DE VIDA

CICLO DE VIDA DE UN PROYECTO DE APRENDIZAJE AUTOMÁTICO



CICLO DE VIDA DE UN PROYECTO DE APRENDIZAJE AUTOMÁTICO



ENTENDIENDO EL NEGOCIO

ENTENDIENDO EL NEGOCIO

Cuando se trabaja en un proyecto de **Aprendizaje Automático**, los científicos de datos suelen enfocarse en **métricas técnicas**, como:

Precisión, F1-score, latencia de inferencia, entre otras.

A menudo, los equipos invierten grandes recursos para mejorar una métrica del 94 % al 94,5 %... Pero la realidad es que, para la mayoría de las empresas, **esas mejoras no importan**, a menos que se traduzcan en **impacto comercial real**.

Para que un proyecto de ML tenga éxito, es clave **conectar las métricas técnicas con los resultados del negocio**.

Algunas preguntas que conviene plantearse son:

- ¿Qué indicadores de negocio busca mejorar el sistema de ML?
- ¿Más ingresos por publicidad?
- ¿Mayor número de usuarios activos?
- ¿Menor tasa de abandono?

ENTENDIENDO EL NEGOCIO

Es importante **vincular las métricas del modelo con las métricas del negocio**.
Una de las razones por las que la **predicción de las tasas de clics en anuncios** y la **detección de fraude** se encuentran entre los casos de uso más populares de ML en la actualidad es que **es fácil asignar el rendimiento de los modelos a las métricas de negocio**: cada aumento en la tasa de clics genera **ingresos publicitarios reales**, y cada transacción fraudulenta detenida representa un **ahorro de dinero real**.

Para obtener una respuesta definitiva a la pregunta de cómo las métricas de ML influyen en las métricas del negocio, a menudo es necesario **realizar experimentos**.
Muchas veces se utilizan pruebas como **experimentos A/B**, eligiendo el modelo que conduce a **mejores métricas comerciales**, independientemente de si este modelo tiene mejores métricas de ML.

ENTENDIENDO EL NEGOCIO

A veces, es difícil manejar las expectativas del negocio.

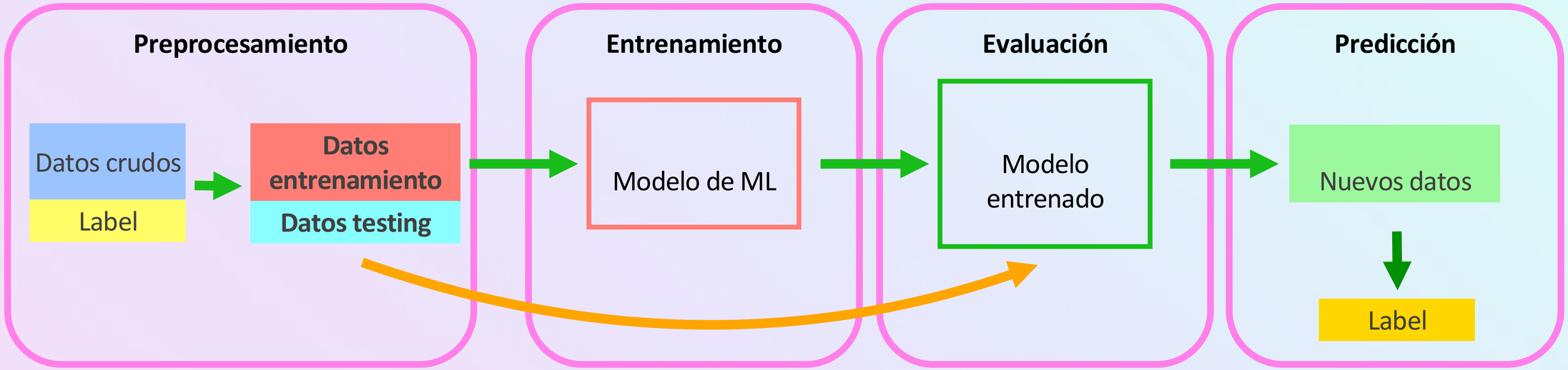
Debido al hype que genera Machine Learning, es necesario ser realista acerca de los beneficios que puede ofrecer un modelo.

Algunos negocios pueden tener la idea de que el Aprendizaje Automático puede transformar una solución mágicamente de la noche a la mañana.

- **Mágicamente: Posible**
- **De la noche a la mañana: No**

METODOLOGÍA PARA CONSTRUIR MODELOS

METODOLOGÍA PARA CONSTRUIR ALGORITMOS DE ML



METODOLOGÍA PARA CONSTRUIR ALGORITMOS DE ML

PREPROCESAMIENTO

Es el **paso más importante**. Recordar: *Garbage in, garbage out*..

Usualmente, los datos se presentan en **formatos no óptimos** (o incluso inadecuados) para ser procesados por el modelo.

Muchos algoritmos requieren que las **características estén en la misma escala** para optimizar su rendimiento, lo cual se logra frecuentemente aplicando técnicas de **normalización** o **estandarización** de los datos.

En algunos casos, las características seleccionadas pueden estar **correlacionadas** y, por tanto, resultar **redundantes** para extraer información significativa. En estos casos, será necesario utilizar técnicas de **reducción dimensional** para comprimir las características en subespacios de menor dimensión.

Realizar un **correcto análisis estadístico** es vital para garantizar que los datos estén preparados adecuadamente para el algoritmo que queremos utilizar.

.

METODOLOGÍA PARA CONSTRUIR ALGORITMOS DE ML

SELECCIÓN Y ENTRENAMIENTO DE MODELO

Es esencial **comparar los diferentes algoritmos** de un grupo para entrenar y seleccionar el de **mejor rendimiento**. Para ello, es necesario **elegir una métrica** que permita medir dicho rendimiento.

Cuando no tenemos nada para comparar inicialmente, elegimos un modelo sencillo llamado **baseline**, que servirá como referencia para evaluar el desarrollo de nuevos modelos. No siempre debe ser un modelo complejo.

Para asegurarnos de que nuestro modelo funcionará adecuadamente con **datos reales**, podemos usar **validación cruzada** (*Cross Validation*) antes de utilizar el conjunto de **datos de prueba** para la evaluación final del modelo.

En general, los **parámetros por defecto** de los algoritmos de **Machine Learning** proporcionados por las librerías no son los óptimos para nuestros datos, por lo que se utilizan técnicas de **optimización de hiperparámetros.**

METODOLOGÍA PARA CONSTRUIR ALGORITMOS DE ML

EVALUANDO Y PREDICIENDO CON DATOS NUEVOS

Una vez que hemos seleccionado y ajustado un modelo con nuestro conjunto de **datos de entrenamiento**, podemos usar los **datos de prueba** para estimar el **rendimiento del modelo** en datos nuevos. Esto nos permite hacer una **estimación del error de generalización** del modelo o evaluarlo utilizando alguna otra métrica.

Si el modelo cumple con nuestros objetivos, el siguiente paso es **ponerlo en producción** en el contexto en que se desea utilizar. En esta materia, **no abordaremos este paso**.