



# IMPLEMENTACIÓN DE ALGORITMOS

14/III/2023




# Contenidos de la sesión



1. Inferencia Online
2. Datos reales de Twitter
3. Inferencia en Streaming
4. ML Automatizado
5. Esenciales curso

# Tipos de inferencia

		OFFLINE	ONLINE	Inferencia en tiempo real
BATCH	ON DEMAND	 Predicciones en batch	Streaming	
		Microservicios y API	ML Automatizado	Inferencia en diferido



# Inferencia online



**Objetivo.** Obtener respuestas al momento.

Para ello vamos a construir un microservicio.

**Def.** Los **microservicios** son un enfoque arquitectónico y organizativo para el desarrollo de software donde el software está compuesto por pequeños servicios independientes que se comunican a través de API bien definidas.

# Pasamos a la práctica

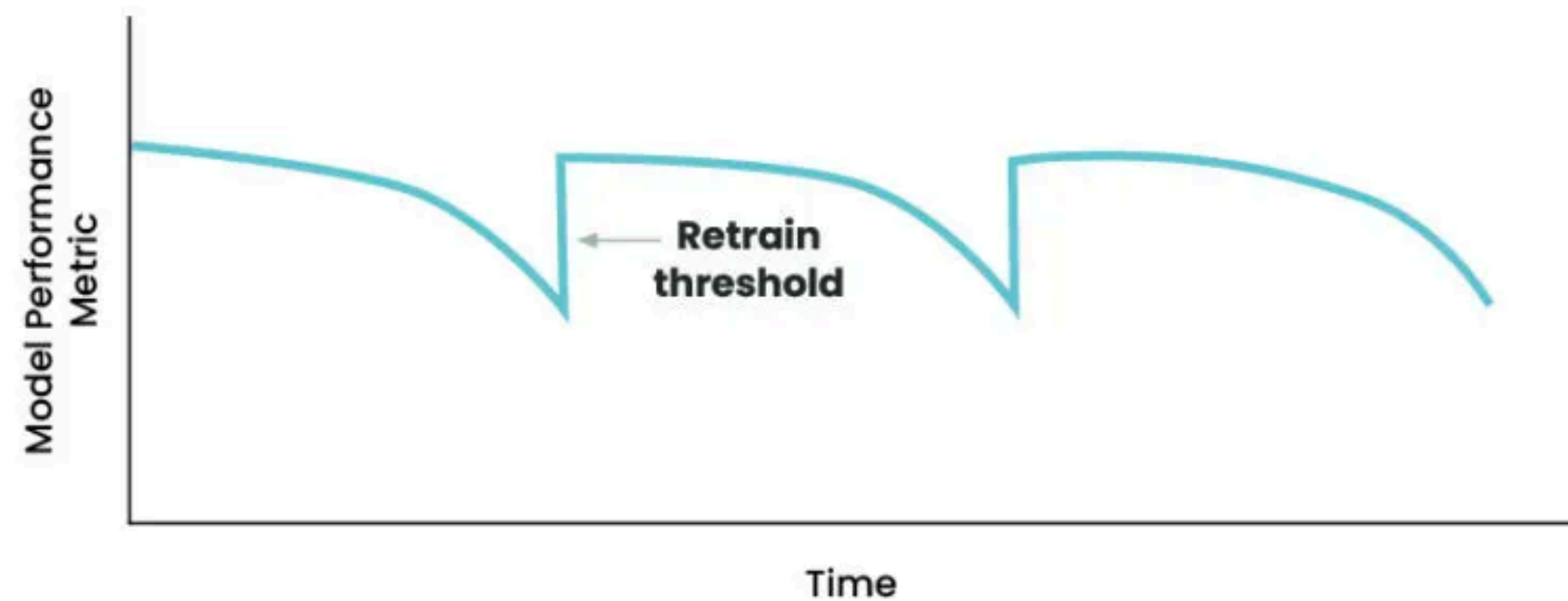
[Enlace a la práctica](#)





# Conceptos clave inferencia online

- Nos permite disponibilidad inmediata del modelo.
- Ideal para realizar testeos y pruebas iniciales.
- Permite hacer un seguimiento del modelo de manera que podemos ver a medida que pasa el tiempo cómo se comporta y si la calidad del modelo se está degradando.

# ¿Cómo funciona el reentrenamiento?



# Tipos de inferencia

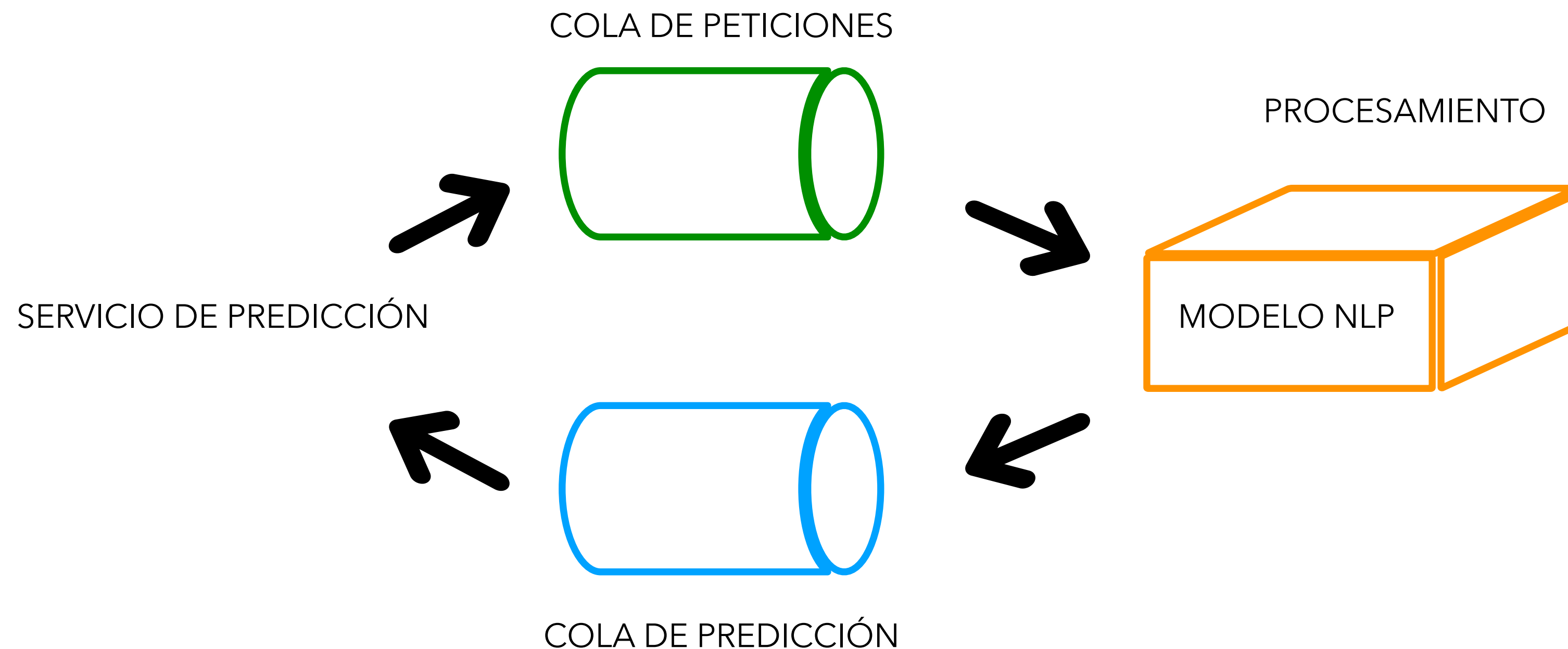
		Inferencia en diferido	Inferencia en tiempo real
BATCH	ON DEMAND	 Predicciones en batch	Streaming
		 Microservicios y API	ML Automatizado



# Concepto de streaming

**Objetivo.** Obtener respuestas al momento de manera constante.

La idea de la arquitectura que vamos a implementar es:



# Concepto de streaming

La diferencia con batch es que aquí vamos a estar trabajando con información en tiempo real.

En este caso vamos a trabajar con tweets que se están publicando en el momento.

Vamos a diseñar una arquitectura capaz de recoger tweets en tiempo real, enviarlos a nuestro modelo y obtener las predicciones del análisis de sentimiento.



# Pasamos a la práctica




[Enlace a la práctica](#)



# Conceptos clave inferencia streaming

- Permite procesar en tiempo real.
- Ideal para problemas que requieren acciones rápidas o problemas de monitoreo.
- Se puede escalar a las necesidades de cada momento.
- Requiere recursos disponibles todo el tiempo.

# Tipos de inferencia

		Inferencia en diferido	Inferencia en tiempo real
BATCH	ON DEMAND	 Predicciones en batch	 Streaming
		 Microservicios y API	ML Automatizado



# ML Automático



**Objetivo.** Tener predicciones en el mismo nivel que el batch pero planificado.

Además se establecen reglas para el reentrenamiento del modelo.

Permite una ejecución totalmente desatendida y es el fin último de las metodologías MLOPS.

Es una manera automatizada de hacer inferencia a nivel Batch, entrenar modelos y realizar el seguimiento de la calidad del modelo, todo de la manera más automatizada posible.

# Ejemplo Práctico ML Automatizado

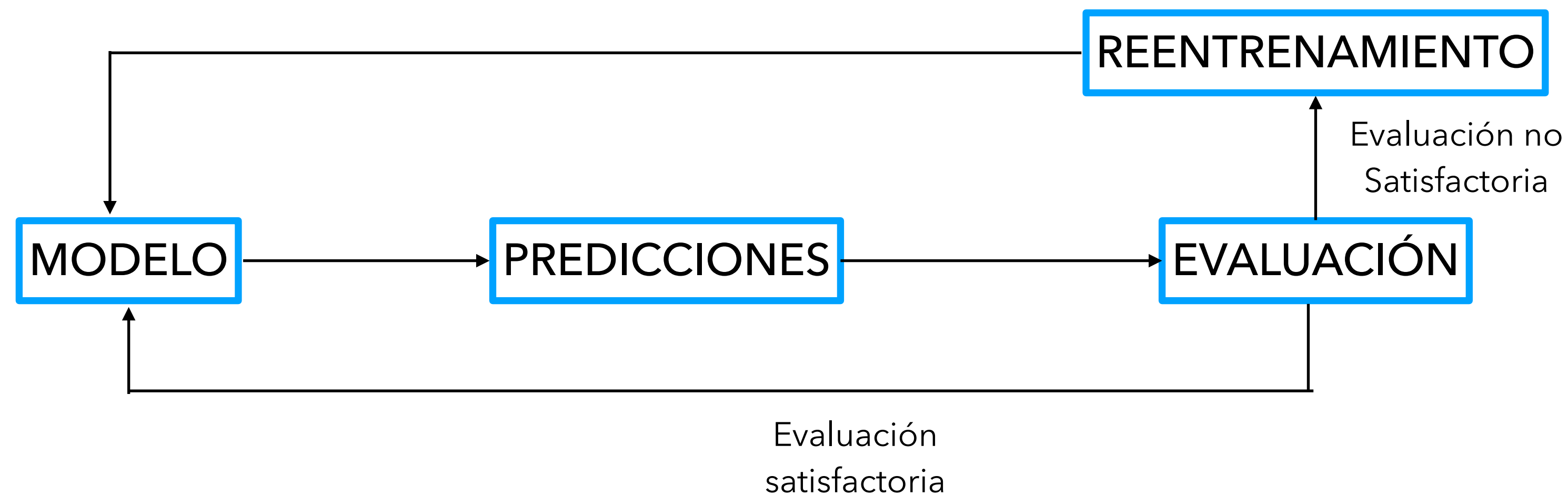
Supongamos que tenemos un modelo para la propensión de compra de un producto. Por ejemplo, un ordenador en Amazon.

Cada semana ejecutamos nuestro modelo y localizamos a las personas más propensas a comprar dicho ordenador.

A estas personas les enviamos publicidad para intentar que se lo compren.





Una parte comprará y otra no.

Nuestro modelo tendrá una tasa de acierto base y cuando caiga de esa tasa podemos reentrenar con todos los nuevos registros.





# Tipos de inferencia

		Inferencia en diferido	Inferencia en tiempo real
BATCH	ON DEMAND	 Predicciones en batch	 Streaming
		 Microservicios y API	 ML Automatizado



# Puntos clave del curso

1. **Apache Beam** es un framework para la construcción de pipelines para procesamiento en paralelo.
2. Apache Beam funciona en distintos lenguajes (Python, Java...) y permite una **gran versatilidad**: cualquier código de Python se puede ejecutar en Apache Beam.
3. Una vez construido un modelo debemos dar una serie de pasos para disponibilizarlo para su uso, a esto lo llamaremos **despliegue del modelo** o **puesta en producción**.
4. El **MLOPS** es una metodología que busca agilizar el paso a producción de los modelos.
5. El tipo de inferencia debe adecuarse al problema que estemos buscando resolver; **no existen ingenierías mejores o peores**, la elección viene únicamente dada por la explotación del modelo.
6. Actualmente la inferencia en **batch** es la más frecuente en la industria aunque poco a poco van apareciendo más casos de inferencia en **streaming**.
7. Las tecnologías en la **nube** nos permiten desplegar **arquitecturas a medida** para nuestros modelos.
8. Es importante en la medida de las posibilidades realizar las **pruebas en local** y no lanzar los trabajos a la nube hasta que sepamos que nuestro código no tiene errores para evitar incurrir en costes innecesarios (tanto monetarios como de tiempo).



# ¡Muchas gracias!