

# Implementación algoritmos





# Contenido de la Masterclass

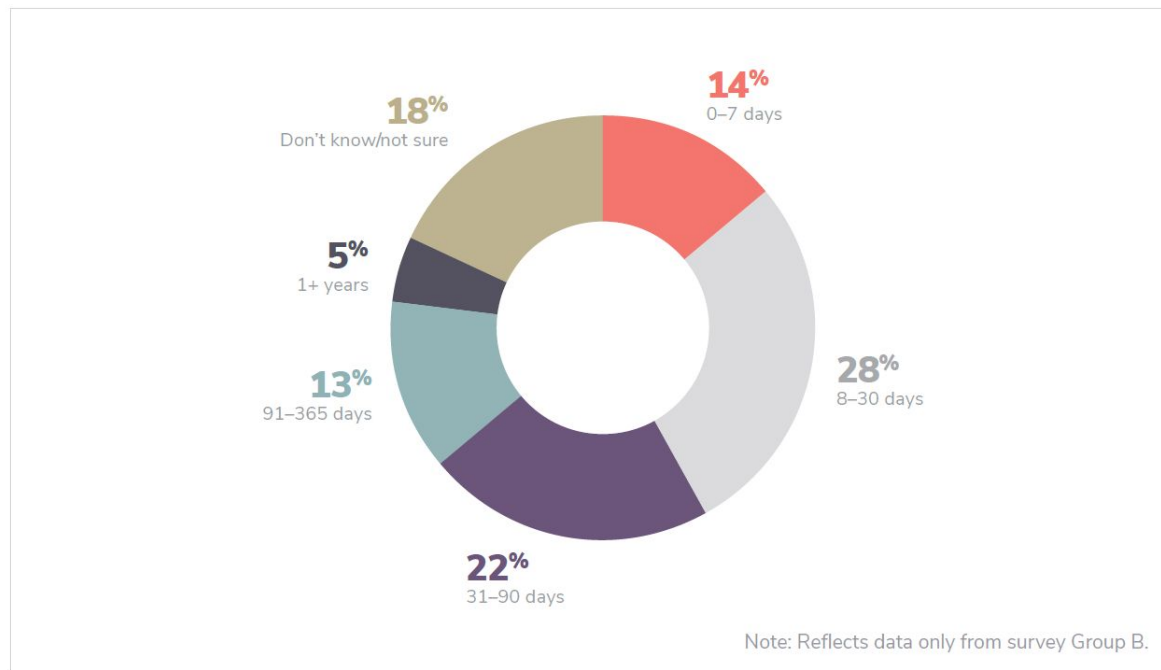
- MLOPS
- Tipos de inferencia
- Cloud
- Apache Beam
- Google Cloud
- Microservicios
- ML Automatizado



# Introducción

# Un problema real

## Machine learning model deployment timeline



# Un problema real

Tecnología

Esto es lo que gasta OpenAI para mantener a ChatGPT funcionando: más de 700.000 dólares al día

<https://www.eleconomista.es/tecnologia/noticias/12243633/04/23/esto-es-lo-que-gasta-openai-para-mantener-a-chatgpt-funcionando-mas-de-700000-dolares-al-dia-.html>



# Importancia de un buen despliegue

- Ritmo acelerado de innovación
- Cada vez más empresas están incorporando IA
- Experiencia de usuario
- Eficiencia operativa y reducción de costes para las empresas

# MLOps





# MLOps



- Fusión de Machine Learning (ML) y operaciones (Ops).
- Práctica que busca la eficiencia en la producción de modelos de ML.
- Garantiza un flujo continuo desde el desarrollo hasta la producción.
- Agiliza la implementación de modelos.
- Asegura modelos más robustos y confiables en producción.
- Facilita la colaboración entre científicos de datos y equipos de operaciones.



# Objetivo de MLOps

- **Automatización:** Integración de datos, entrenamiento, testing implementación y supervisión
- **Supervisión:** Monitorizar eficacia, precisión y rendimiento del modelo
- **Adaptabilidad:** Permitir que los modelos se entrenen según necesidades
- **Colaboración:** Mejorar la colaboración entre equipos de desarrollo
- **Trazabilidad:** Permitir que las operaciones del modelo se puedan trazar completamente



# Tipos de inferencia



# Inferencia



- Usar un modelo entrenado para predecir nuevos datos
- El modelo muestra su valor
- Proporciona información en base a su entrenamiento



# Inferencia en Batch

- Inferencia en conjuntos de datos completos
- **Ventajas** = Más eficiente en términos computacionales y más fácil de implementar
- **Desventajas** = No es en tiempo real, latencias importantes en la recopilación de datos

# Inferencia Online

- Inferencia en tiempo real de cuando van llegando los datos
- **Ventajas** = Proporciona resultados en tiempo real y permite tomar decisiones rápidas
- **Desventajas** = Más complejo, más costoso y requiere de infraestructura robusta y escalable.

# Inferencia Online (streaming)



- Tratamiento y respuesta a un flujo ininterrumpido de datos en tiempo real o casi en tiempo real.
- A diferencia de la inferencia online discreta, la streaming opera sobre datos que fluyen constantemente.
- Respuestas y análisis se producen casi inmediatamente a medida que los datos llegan.
- Monitoreo en tiempo real, análisis de tendencias en vivo, sistemas de alerta, procesamiento de eventos complejos.

# Batch VS Online

Consideraciones clave:

¿Es un problema que requiere una respuesta inmediata o se puede esperar?

¿Estamos tratando con enormes volúmenes de datos que serían más eficientes en un sistema en batch?

¿Tenemos los recursos para manejar la inferencia en streaming en términos de infraestructura y costos?

¿El usuario final espera una respuesta inmediata o está dispuesto a esperar?

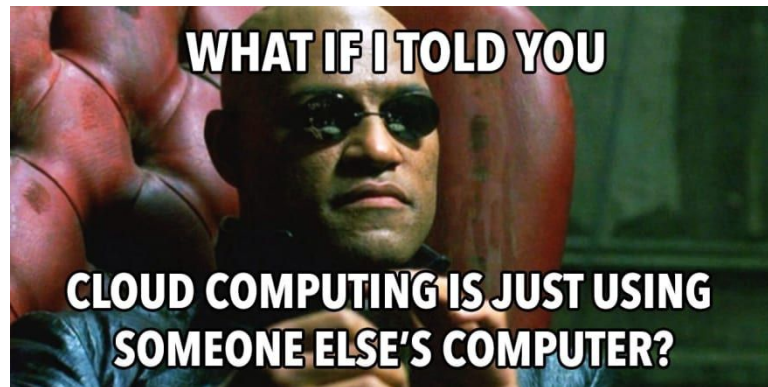


# Cloud



# Computación Cloud

La computación en la nube se refiere a la entrega de servicios informáticos a través de Internet (la nube), incluyendo almacenamiento, procesamiento, bases de datos, redes, software, análisis e inteligencia artificial





# Ventajas

- Escalabilidad
- Flexibilidad
- Reducción de costos (Pay as you go)
- Around the world
- Seguridad, mantenimiento y actualizaciones a cargo del proveedor

# Herramientas





# Apache Beam

- Apache Beam es un modelo unificado y open-source para definir pipelines de procesamiento.
- Puede manejar tanto procesamientos en batch como en streaming.
- Destaca por su versatilidad y eficiencia en escenarios con alto paralelismo y grandes volúmenes de datos.
- Permite definir un único pipeline para ambos modos de procesamiento, simplificando el desarrollo.



# Diferencia con scikit-learn

- **Apache Beam:** Diseñado para operaciones en paralelo de alto volumen.
- **scikit-learn:** Biblioteca de aprendizaje automático en Python.
- **Diferencia clave:** scikit-learn no está diseñado para procesamiento paralelo distribuido.

# Pipeline

Es la entidad principal en Beam y encapsula toda la tarea de procesamiento. Un pipeline se encarga de leer datos de entrada, aplicar diversas transformaciones y finalmente escribir los datos de salida. En términos simples, puede imaginarse como una "tubería" por la que fluyen los datos, siendo procesados en diferentes etapas.





# PCollection

Este es un concepto fundamental en Beam y representa un conjunto de datos distribuido. Es el formato en el que los datos son manejados dentro del pipeline, permitiendo que se realicen operaciones en paralelo sobre ellos.

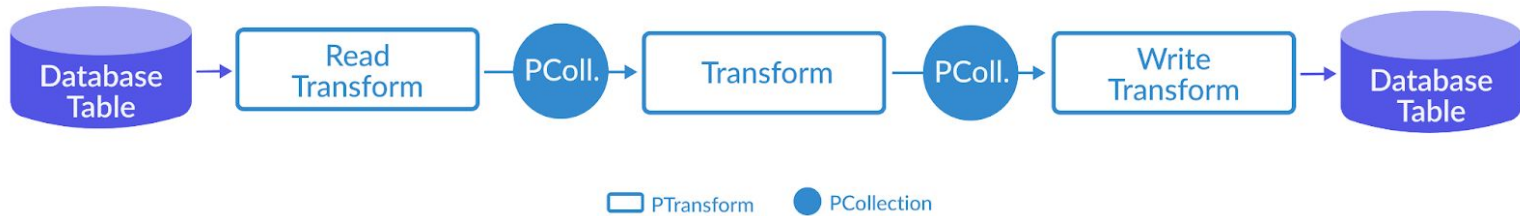




# PTransform

Representa una operación de procesamiento dentro de un pipeline. Puede ser una operación simple, como filtrar datos, o una compleja, como realizar un join entre dos PCollections.

# Flujo





# Dataflow



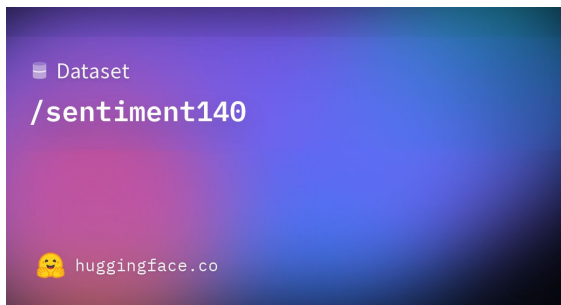
- Dataflow es un sistema unificado para procesamiento de datos en flujo y en lotes que es serverless, rápido y rentable.
- Permite desarrollo ágil de pipelines de datos en flujo con baja latencia.
- Libera a los equipos de gestionar clusters de servidores, eliminando la carga operativa de las tareas de ingeniería de datos.
- Autodimensionamiento de recursos y procesamiento por lotes optimizado para costos.
- Facilita la observación y diagnóstico de problemas en cada paso del pipeline de Dataflow.

# AI Platform



- AI Platform te permite entrenar modelos de aprendizaje automático a gran escala, alojar tu modelo entrenado en la nube y hacer predicciones sobre nuevos datos.
- Construye modelos con datos de cualquier tamaño usando infraestructura de entrenamiento distribuido.
- Integración con Cloud Dataflow, Cloud Storage y Cloud Datalab.

# LAB1





# Kahoot!



# Microservicios





# API

- Una API es un conjunto de reglas y definiciones que permite que dos aplicaciones se comuniquen entre sí.
- Oculta la complejidad interna y sólo expone funciones relevantes al consumidor.
- Permite que diferentes sistemas y aplicaciones interactúen y compartan datos.
- Las APIs pueden manejar grandes volúmenes de solicitudes y responder a las demandas cambiantes de los usuarios.

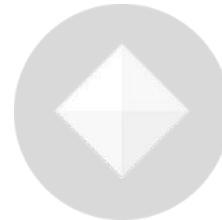
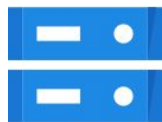
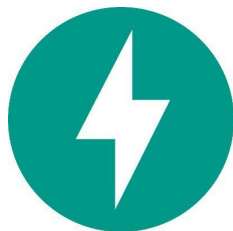
# Microservicios

- **Independencia:** Cada microservicio es autónomo y puede desarrollarse, implementarse y escalarse de forma independiente.
- **Foco en una tarea:** Cada servicio se centra en realizar una sola función o tarea específica.
- **Descentralización:** La gestión de datos y funciones es descentralizada, lo que permite una mayor flexibilidad y resistencia.
- **Interoperabilidad:** Los microservicios están diseñados para funcionar en diferentes plataformas y comunicarse con otros servicios.

# Serverless

- Se refiere a un modelo en el que los desarrolladores construyen y ejecutan aplicaciones sin preocuparse por la infraestructura subyacente.
- Solo pagas por la ejecución real de tu código, no por el tiempo de inactividad.
- Se escala automáticamente con el número de ejecuciones, desde unas pocas solicitudes por día hasta miles por segundo.
- Los proveedores manejan todas las tareas operativas, como el mantenimiento del servidor, la actualización y la seguridad.

# LAB2





localhost:8000



Túnel



Web

# LAB3





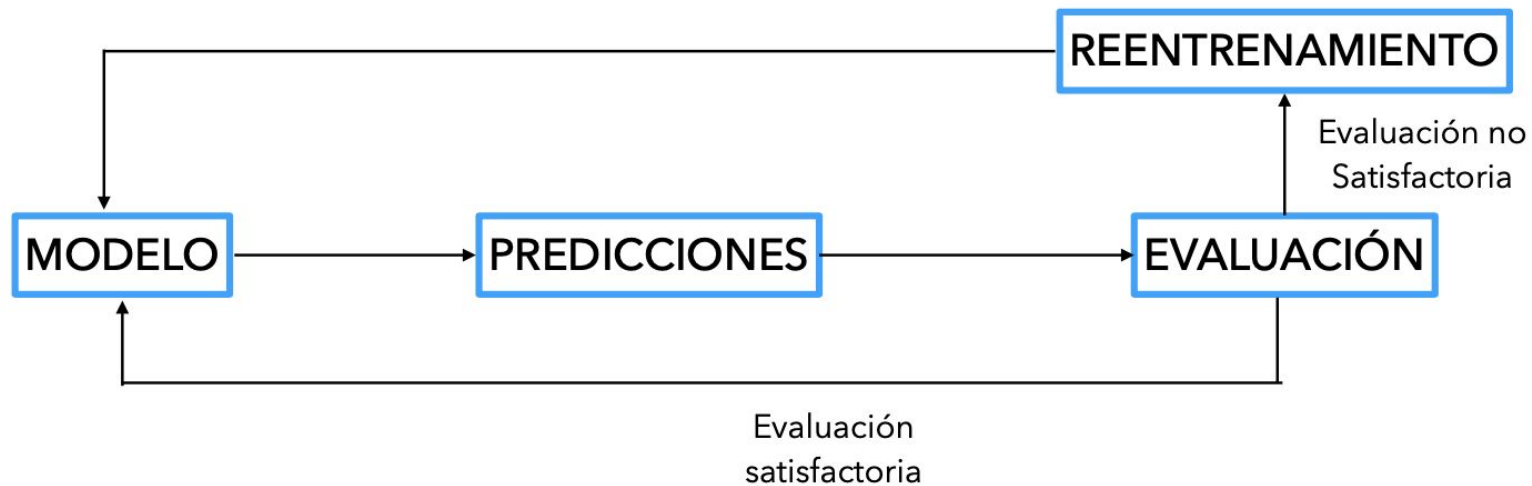
# ML Automatizado

# ML Automatizado

- 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/Streaming, entrenar modelos y realizar el seguimiento de la calidad del modelo, todo de la manera más automatizada posible.



# ML Automatizado





# FUTURO?



# FINAL BOSS



# KEEPCODING

## Tech School

Madrid | Barcelona | Bogotá

Eric Risco de la Torre

[erisco@icloud.com](mailto:erisco@icloud.com)

<https://www.linkedin.com/in/erisco-and/>