

## Módulo 12 Datos masivos

*Mtro. Omar Mendoza González*



**DGTIC**

# Contenido

## 3. Machine Learning Distribuido

3.1 MLlib

3.2 Algoritmos de ML

3.3 Aprendizaje supervisado

3.4 Aprendizaje no supervisado



# **MLlib (Machine Learning Library)**

- MLlib es la librería de Machine Learning (ML) de Apache Spark
- Ayuda a administrar y simplificar muchas de las tareas para la construcción de modelos de aprendizaje automático, como la caracterización, pipeline, evaluar y ajustar el modelo.

# MLlib

- Spark hace que sea extremadamente fácil implementar algoritmos de ML y ejecutarlos de manera escalable a través de un cluster de máquinas.



# Herramientas MLlib

- ML Algorithms
  - Forman el núcleo de MLlib.
  - Se incluyen algoritmos de aprendizaje comunes como clasificación, regresión, agrupamiento y filtrado colaborativo.
  - MLlib estandariza las API para facilitar la combinación de múltiples algoritmos en una sola canalización

# Herramientas MLib

- Featurization
  - Extracción de características.
  - Transformación de características
  - Selección de características
  - Reducción de dimensionalidad



# Herramientas MLlib

- Pipelines
  - Consiste en una secuencia de Etapas de Pipeline (Transformadores y Estimadores) que se ejecutarán en un orden específico.
- Model Tuning
  - Entrenar un modelo con el conjunto correcto de parámetros para lograr el mejor rendimiento para cumplir con el objeto definido en el primer paso del proceso de desarrollo de ML.

# Herramientas MLlib

- Persistence
  - Ayuda a guardar y cargar algoritmos, modelos y pipelines de ML
- Utilities
  - Álgebra lineal
  - Estadística
  - Manejo de datos

# Herramientas de MLlib

- **Algoritmos de ML**
  - Estadística Básica
  - Algoritmos de Clasificación y Regresión
  - Sistemas de Recomendación
  - Clustering
  - Gestión de Features
  - Optimización
  - Descomposición en valores singulares (SVD)
  - Análisis de los componentes principales (PCA)
  - Comprobación de hipótesis y cálculo de estadísticas de ejemplo

# Spark ML

- "*Spark ML*" no es un nombre oficial, pero ocasionalmente se usa para referirse a la API basada en MLlib DataFrame.
- Esto se debe principalmente al nombre del paquete org.apache.spark.ml
- MLlib incluye tanto la API basada en RDD como la API basada en DataFrame.
  - La API basada en RDD ahora está en modo de mantenimiento.
  - Pero ni la API ni MLlib en su conjunto están en desuso.

# MLlib, componentes Pipeline

- DataFrame
  - esta API de ML usa DataFrame de Spark SQL como un conjunto de datos de ML, que puede contener una variedad de tipos de datos.
- Transformador
  - Es un algoritmo que puede transformar un DataFrame en otro DataFrame.
  - Un modelo ML es un Transformer que transforma un DataFrame con *features* en un DataFrame con predicciones.
- Estimador
  - Es un algoritmo que se puede ajustar a un DataFrame para producir un transformador.
  - Un algoritmo de aprendizaje es un estimador que se entrena en un DataFrame y produce un modelo.

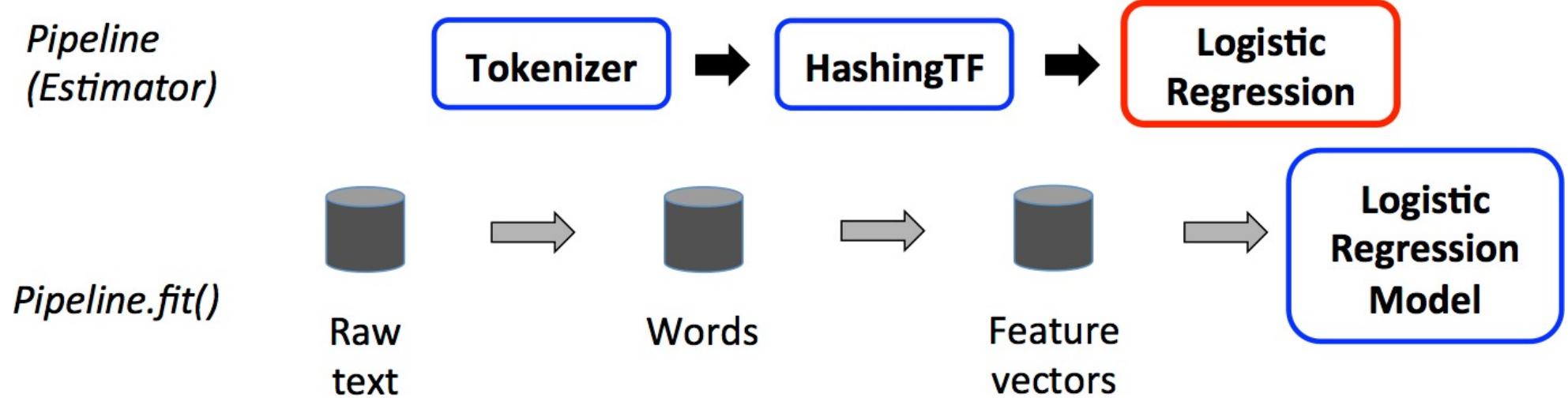
# MLlib, componentes Pipeline

- Pipeline
  - Encadena varios transformadores y estimadores para especificar un flujo de trabajo de ML
- Parámetro
  - Todos los transformadores y estimadores ahora comparten una API común para especificar parámetros.

# Mlib, componentes Pipeline

- Pipeline
  - Encadena varios transformadores y estimadores para especificar un flujo de trabajo de ML
- Parámetro
  - Todos los transformadores y estimadores ahora comparten una API común para especificar parámetros.
  - Hay dos formas principales de pasar parámetros a un algoritmo:
  - Establecer parámetros para una instancia.
  - Pase un ParamMap a fit() o transform()
  - ParamMap es un conjunto de pares (parámetro, valor)

# Mlib, componentes Pipeline



# Consideraciones del proyecto de ML

- Dependiente de datos
- Recursos informáticos disponibles
- Computación en una sola máquina vs computación distribuida
- Inferencia
  - requisitos de implementación

# Consideraciones del proyecto de ML

	Rendimiento	Latencia	Ejemplo
<b>Batch</b>	Alto	Horas a días	Predicción de abandono de clientes
<b>Streaming</b>	Medio	Segundos a minutos	Mantenimiento predictivo
<b>Real-time</b>	Bajo	Milisegundos	Detección de fraudes

# Contacto

Omar Mendoza González

*Profesor de carrera ICO FES Aragón*

[omarmendoza564@aragon.unam.mx](mailto:omarmendoza564@aragon.unam.mx)

# Referencias

- **Corea, Francesco, An Introduction to data : everything you need to know about AI, Big data and data science / Francesco Corea -- Cham, Switzerland : Springer, [2019].-- xv, 131 páginas : ilustraciones (Studies in Big data, 2197-6503 ; 50 )**
- **Casas Roma, Jordi, Big data : análisis de datos en entornos masivos / Jordi Casas Roma, Jordi Nin Guerrero, Francesc Julbe López -- Barcelona : Editorial UOC, 2019 287 páginas : ilustraciones (Tecnología ; 623 ).**
- **Caballero, Rafael, Big data con Python recolección, almacenamiento y proceso / Rafael Caballero Adrián Riesco Enrique Martín: Universidad Complutense de Madrid Editorial AlfaOmega, 2019 282 páginas**
- **Rioux, Jonathan, Data Analysis with Python and PySpark / Jonathan Rioux: Editorial Manning Publications, 2020 259 páginas**
- **Singh, Pramod, Machine Learning with PySpark: With Natural Language Processing and Recommender Systems / Pramod Singh: Editorial Apress, 2019 233 páginas**