Máster en Big Data y Data Science:
Ciencia e Ingeniería de Datos

UNIVERSIDAD AUTONOMA DE MADRID

ECOSISTEMA SPARK

Nº de Créditos: **3 ECTS**
Segundo Semestre
Primer Curso

EQUIPO DOCENTE

Villegas, Paulo
Escuela Politécnica Superior
UAM
Coordinador

Del Cacho, Carlos
Jobandtalent

Martínez Muñoz, Gonzalo
Escuela Politécnica Superior
UAM

Pulido Cañabate, Estrella
Escuela Politécnica Superior
UAM

OBJETIVOS

- Resolver problemas utilizando el paradigma de computación en paralelo de Apache Spark.
- Manejar APIs en Spark en distintos lenguajes de programación.
- Crear soluciones en Apache Spark que utilicen datos estructurados, métodos de aprendizaje automático y/o fuentes de datos de *streaming*.
- Resolver algoritmos sobre grafos en Spark.

PROGRAMA DETALLADO

- Fundamentos de Spark
  - Introducción: arquitectura y organización
  - Datos en Spark: *Resilient Distributed Datasets* (RDDs)
  - Flujo de un programa Spark
  - Entrada y salida de datos
  - Transformaciones
  - Persistencia
  - Acciones
  - Variables compartidas: *broadcast* y acumuladores
- Spark SQL
  - Introducción a DataFrames
  - Fuentes de datos: Hive, JDBC/ODBC, Parquet, etc.
  - API de DataFrames
- Lenguajes y APIs en Spark
  - APIs ofrecidas por Spark: Scala, Java, Python, R
  - SparkR: paralelización de data frames de R
  - Dataset, API unificada, SparkSession (Spark 2.0)
- Procesamiento de grafos vía Spark
  - Operadores sobre grafos
  - Librerías clásicas: GraphX
  - Algoritmos de grafos
  - Nuevas librerías
- MLlib
  - Aprendizaje supervisado: clasificación y regresión
  - Aprendizaje no supervisado
  - Creación de pipelines de aprendizaje automático
  - Contrastes de hipótesis
  - Sistemas de recomendación (ALS)
  - Reglas asociativas
- *Streaming* en Spark
  - Spark Streaming clásico: *Discretized Streams* (DStreams)
  - Entrada y salida
  - Operaciones con DStreams
  - Mantenimiento de estado. Ventanas
  - Tolerancia a fallos. Checkpoints. Ajustes
  - Streaming estructurado

BIBLIOGRAFÍA

1. Matei Zaharia et al, "***Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing***." In Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation, pp. 2-2. USENIX Association, 2012.
2. Matei Zaharia et al. "***Spark: Cluster Computing with Working Sets***." HotCloud 10 (2010): 10-10
3. Michael Armbrust et al. "***Spark SQL: Relational data processing in Spark***" Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. ACM, 2015.
4. Xiangrui Meng et al. "***Mllib: Machine Learning in Apache Spark.***" JMLR 17.34 (2016): 1-7.
5. Matei Zaharia et al, "***Discretized Streams: Fault-Tolerant Streaming Computation at Scale***", Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles. ACM, 2013.
6. Xin, Reynold S., et al. "***Graphx: A Resilient Distributed Graph System on Spark.***" First International Workshop on Graph Data Management Experiences and Systems. ACM, 2013.
7. Apache Spark, **Spark Overview & Documentation**, http://spark.apache.org/docs/latest/
8. Zaharia, M. et al, "***Learning Spark",*** O'Reilly Media, 2015.

MÉTODOS DOCENTES

- Lección magistral
- Resolución de problemas
- Prácticas de laboratorio
- Estudio de casos

MÉTODOS DE EVALUACIÓN

- Asistencia a clase: **10%**
- Evaluación continua: **40%**
- Examen final: **50%**