**Trabajo Final Grupo N°10**

Patricio Nicolas Argentino

Victor Alejandro Ordoñez

Docente

RIVADERA GUSTAVO RAMIRO

Universidad Católica de Salta

Facultad de Ingeniería

Ldor.Gral.San Matín (Jujuy) 2023

Trabajo Final Grupo N°10

# Introducción:

Aplicación de la tecnología Spark para abordar casos reales de gestión de información de consumo de materiales en el departamento de Abastecimiento de una empresa Industrial.

# Métodos

# Resultados

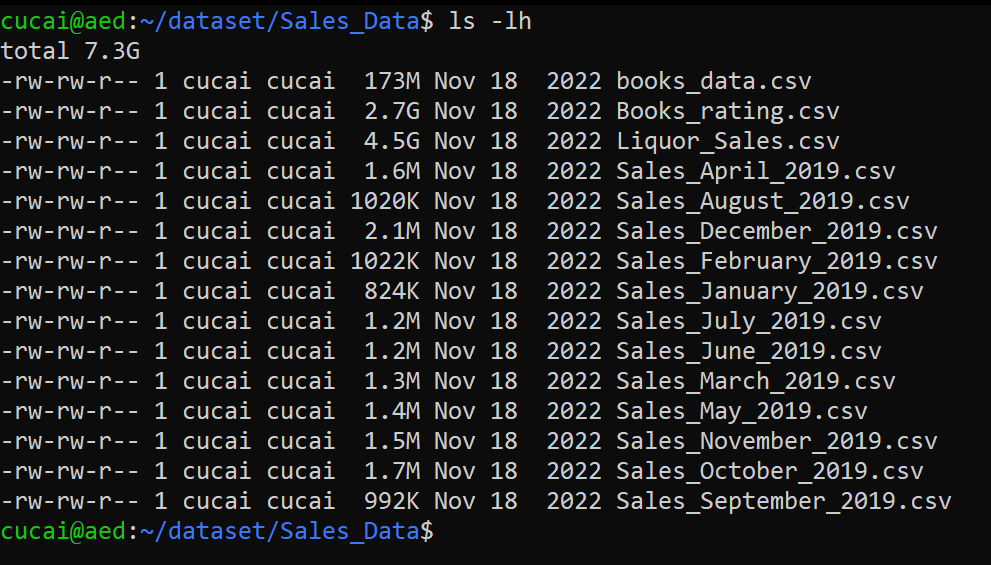
# Conclusiones

# Referencias

# Apéndices

Origen del Dataset

<https://www.kaggle.com/datasets/pigment/big-sales-data>

D

carpeta:

/home/cucai/dataset/Sales\_Data/

books\_data.csv

Books\_rating.csv

Liquor\_Sales.csv

Sales\_April\_2019.csv

Sales\_August\_2019.csv

Sales\_December\_2019.csv

Sales\_February\_2019.csv

Sales\_January\_2019.csv

Sales\_July\_2019.csv

Sales\_June\_2019.csv

Sales\_March\_2019.csv

Sales\_May\_2019.csv

Sales\_November\_2019.csv

Sales\_October\_2019.csv

Sales\_September\_2019.csv

Comandos

1. Inicio el master:  
   cucai@aed:~/bigdata/spark-3.5.0-bin-hadoop3/sbin$ **./start-master.sh**
2. Accedo a: <http://190.182.223.172:8080/> - localhost:8080
3. Inicio un wroker a partir de la url proporcionada por el master  
   cucai@aed:~/bigdata/spark-3.5.0-bin-hadoop3/sbin$ **./start-worker.sh spark://aed:7077**
4. cucai@aed:~/bigdata/spark-3.5.0-bin-hadoop3/bin$ ./pyspark

**Como instalar spark**

**Cómo instalar pyspark**

Código:

# Librerias que se usaran

from pyspark.sql.functions import to\_date

from pyspark.sql.functions import regexp\_replace

# Importo información de materiales

materiales = spark.read.options(header='True', inferSchema='True',delimiter=';').csv('m.csv')

# Importo información de consumos desde 2018

consumos = spark.read.options(delimiter=';',header='False',inferSchema='True').csv("ConsumosMicro2023.txt")

# Reemplazo el nombre de las cabeceras

consumos = consumos.withColumnRenamed('\_c0', 'id').withColumnRenamed('\_c1','Fecha').withColumnRenamed('\_c2','codigo').withColumnRenamed('\_c3','mov').withColumnRenamed('\_c4','Almacen').withColumnRenamed('\_c6','Doc').withColumnRenamed('\_c8','Salida')

# Verifico la estructura del dataframe

consumos.printSchema()

# Elimino las columnas inecesarias

consumos = consumos.drop('\_c5')

consumos = consumos.drop('\_c7')

# Cambio el formato del campo Fecha de string a Date

consumos = consumos.withColumn("Fecha", to\_date("Fecha", "d/M/yyyy H:mm:ss"))

# Creo la columna Salidas de tipo float a partir de un campo de tipo string

consumos = consumos.withColumn("Salidas", regexp\_replace("Salida", ",", "."))

# Elimino la columna salida

consumos = consumos.drop('Salida')

# Convertir columnas númericas a string

consumos = consumos.withColumn('Material',consumos['codigo'].cast('string'))

consumos = consumos.withColumn('Reservas',consumos['Doc'].cast('string'))

consumos = consumos.drop('codigo')

consumos = consumos.drop('Doc')

from pyspark.sql.functions import sum

total = consumos.select(sum('Salidas')).collect()[0][0]

# Mostrar el total de la suma

print(total)

consumos.select(['codigo','Salidas']).filter("codigo='1017142'").show()

# RESUMO LA INFORMACION DE LOS CONSUMOS

df = consumos.groupBy(['Fecha', 'codigo']).agg({'Salidas': 'sum'})

df.write.csv("output.csv", header=True)