spark_colab_ejercicios

November 14, 2025

1 Apache Spark en Google Colab

Ejercicios de WordCount, DataFrame API y MLlib (clasificación)

[ ]:
```
#configuración en google colab de spark y pyspark
from google.colab import drive
drive.mount('/content/gdrive')
```

[ ]:
```
!apt-get install openjdk-17-jdk-headless -qq > /dev/null
!wget -q https://downloads.apache.org/spark/spark-4.0.1/spark-4.0.1-bin-hadoop3.
 ↪tgz
!tar xf spark-4.0.1-bin-hadoop3.tgz
!pip install -q findspark
!pip install -q pyspark
import os
os.environ["JAVA_HOME"] = "/usr/lib/jvm/java-17-openjdk-amd64"
os.environ["SPARK_HOME"] = "/content/spark-4.0.1-bin-hadoop3"
import findspark
findspark.init()
```

1.1 Ejemplo 1: WordCount con RDD

[ ]:
```
from pyspark import SparkContext
sc = SparkContext.getOrCreate()

text = sc.textFile("gdrive/MyDrive/st0263-252/bigdata/datasets/gutenberg-small/
 ↪*.txt")
# Simular archivo de texto
# text = sc.parallelize(["Hola Spark Hola Big Data", "Spark es rápido y␣
 ↪poderoso"])
counts = text.flatMap(lambda x: x.split(" ")) \
             .map(lambda x: (x, 1)) \
             .reduceByKey(lambda a, b: a + b)
counts.collect()
```

1.2 Ejemplo 2: Análisis con DataFrame API

[ ]:

```
from pyspark.sql import SparkSession
spark = SparkSession.builder.getOrCreate()

# Simular DataFrame de ventas
data = [("martillo", 12000), ("taladro", 45000), ("martillo", 15000)]
columns = ["producto", "valor"]
df = spark.createDataFrame(data, columns)
df.groupBy("producto").sum("valor").show()
```

1.3 Ejemplo 3: Clasificación con MLlib

[ ]:

```
from pyspark.ml.feature import VectorAssembler
from pyspark.ml.classification import LogisticRegression

df = spark.read.csv("gdrive/MyDrive/st0263-252/bigdata/datasets/clientes.csv",␣
 ↪header=True, inferSchema=True)

assembler = VectorAssembler(inputCols=["edad", "ingresos"],␣
 ↪outputCol="features")
data = assembler.transform(df).select("features", df["comprador"].
 ↪alias("label"))
train, test = data.randomSplit([0.8, 0.2], seed=42)
lr = LogisticRegression()
model = lr.fit(train)
model.transform(test).select("features", "label", "prediction").show()
```

1.4 Ejemplo 4: Spark GraphX

[ ]:

```
!pip install -q pyspark
!pyspark --packages graphframes:graphframes:0.8.3-spark3.5-s_2.12
```

[ ]:

```
from pyspark.sql import SparkSession

spark = SparkSession.builder \
    .appName("GraphFrames PageRank") \
    .config("spark.jars.packages", "graphframes:graphframes:0.8.3-spark3.5-s_2.
 ↪12") \
    .getOrCreate()
```

[ ]:

```
from graphframes import GraphFrame
from pyspark.sql import DataFrame

# DataFrame de vértices
vertices = spark.createDataFrame([("1", "A"), ("2", "B"), ("3", "C"), ("4",␣
 ↪"D") ], ["id", "name"])
```

```
# DataFrame de edges
edges = spark.createDataFrame([("1", "2"),    ("2", "3"),    ("3", "4"),   ␣
 ↪("4", "1")], ["src", "dst"])

# grafo
g = GraphFrame(vertices, edges)
```

[ ]:

```
# algoritmo de PageRank
results = g.pageRank(resetProbability=0.15, maxIter=10)

# resultados de PageRank
results.vertices.select("id", "name", "pagerank").show()
```