Universidad Nacional Autónoma de México

Facultad de Estudios Superiores Acatlán

Práctica

Práctica Spark

Presenta

Jiménez Pineda Leydi Monserrat

N° de cuenta

421089037

Profesor

JOSÉ GUSTAVO FUENTES CABRERA

Materia

Programación Paralela y Concurrente

Santa Cruz Acatlán, Naucalpan, Estado de México

Objetivo

Aplicar los conceptos fundamentales de procesamiento distribuido mediante PySpark para analizar datos reales de movilidad urbana, utilizando un conjunto de datos del sistema Ecobici de la Ciudad de México.

Descripción del Dataset

Se utilizó el conjunto de datos real ecobicis_20230430.csv, obtenido desde el Portal de Datos Abiertos de la Ciudad de México. Este archivo contiene información agregada de viajes en bicicletas compartidas del sistema Ecobici, incluyendo:

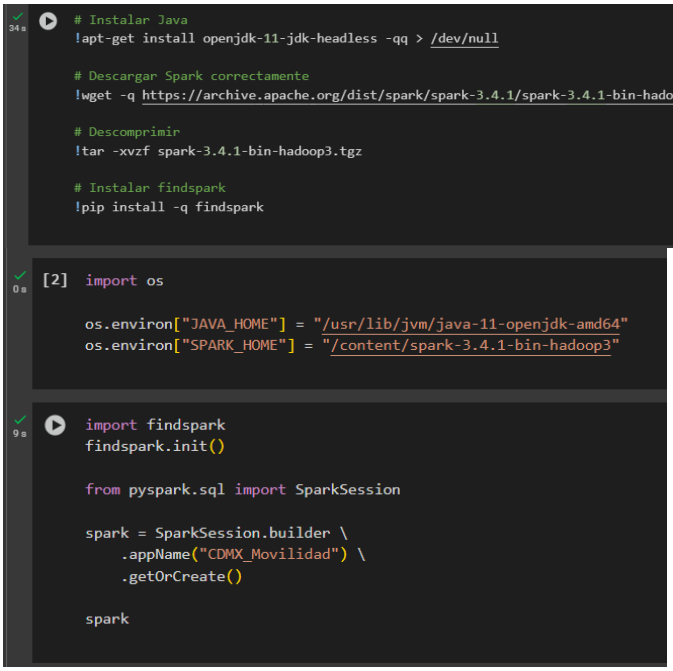
- anio, mes, fecha_referencia: identificadores temporales
- bici: identificador de la bicicleta
- mins_viaje, hrs_viaje: duración del viaje
- dias_viaje: cantidad de días en que se usó la bicicleta
- distancia_approx: distancia estimada

Cabe aclarar que el dataset no contiene información sobre estaciones de origen ni destino, por lo que no fue posible realizar análisis de rutas o zonas geográficas.

Operaciones Realizadas con PySpark

Se aplicaron operaciones sobre un DataFrame utilizando PySpark

1.- Estas instrucciones permiten ejecutar Spark en un entorno no tradicional como Google Colab, especificando rutas manuales e iniciando la sesión de Spark.

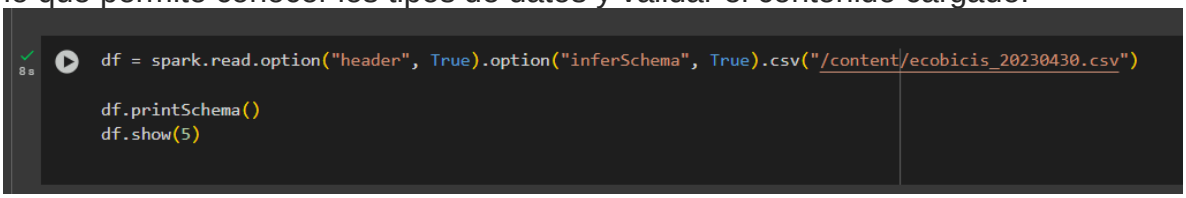


```
# Instalar Java
!apt-get install openjdk-11-jdk-headless -qq > /dev/null

# Descargar Spark correctamente
!wget -q https://archive.apache.org/dist/spark/spark-3.4.1/spark-3.4.1-bin-hadoop3.tgz

# Descomprimir
!tar -xvzf spark-3.4.1-bin-hadoop3.tgz

# Instalar findspark
!pip install -q findspark
```

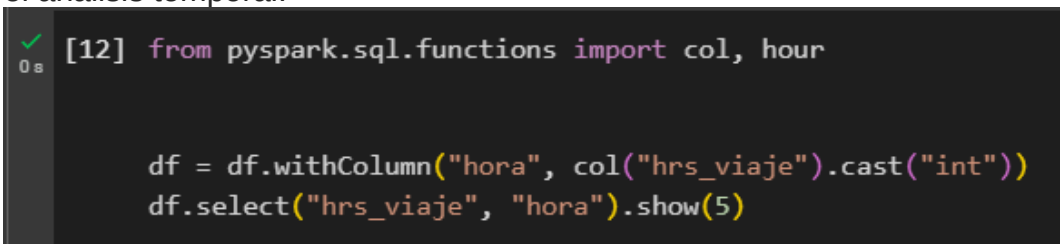
```
[2] import os

os.environ["JAVA_HOME"] = "/usr/lib/jvm/java-11-openjdk-amd64"
os.environ["SPARK_HOME"] = "/content/spark-3.4.1-bin-hadoop3"
```

```
import findspark
findspark.init()

from pyspark.sql import SparkSession

spark = SparkSession.builder \
    .appName("CDMX_Movilidad") \
    .getOrCreate()

spark
```

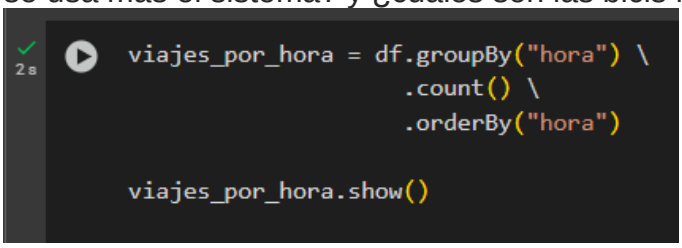
2.- Se usa Spark para leer el archivo CSV directamente en formato estructurado, lo que permite conocer los tipos de datos y validar el contenido cargado.



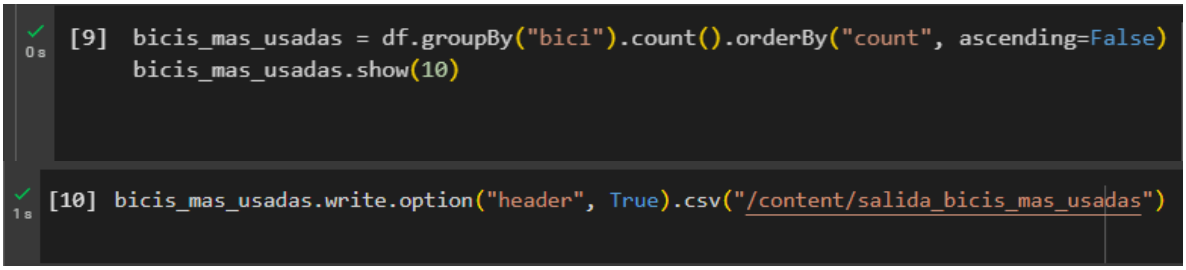
3.- Esta transformación permite realizar agrupaciones por hora del día, facilitando el análisis temporal.



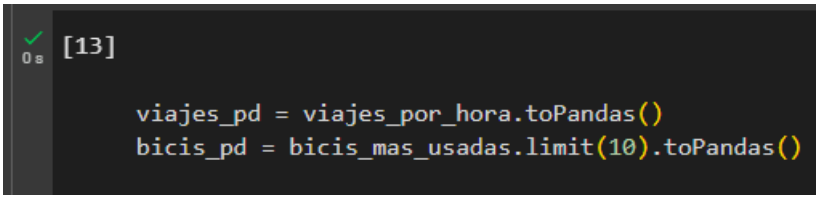
4. Estas agregaciones permiten responder preguntas clave como: ¿en qué horas se usa más el sistema? y ¿cuáles son las bicis más demandadas?



5.- El resultado puede guardarse como archivo CSV para compartir o utilizar en reportes o dashboards.



6.- Se convierte a DataFrame de Pandas para permitir graficación con bibliotecas visuales.



7.- Esta parte del código muestra unas gráficas que muestran visualmente los horarios más ocupados y las bicicletas más utilizadas, facilitando el análisis y la interpretación para usuarios no técnicos.

```
import matplotlib.pyplot as plt


plt.figure(figsize=(10, 5))
plt.bar(viajes_pd['hora'], viajes_pd['count'], color='skyblue')
plt.title("Cantidad de viajes por hora del día")
plt.xlabel("Hora del día")
plt.ylabel("Número de viajes")
plt.xticks(range(0, 24))
plt.grid(True, linestyle='--', alpha=0.5)
plt.show()
```

```
plt.figure(figsize=(10, 5))
plt.bar(bicis_pd['bici'].astype(str), bicis_pd['count'], color='orange')
plt.title("Top 10 bicis más utilizadas")
plt.xlabel("ID de bicicleta")
plt.ylabel("Número de viajes")
plt.xticks(rotation=45)
plt.grid(True, linestyle='--', alpha=0.5)
plt.show()
```

Análisis de resultados

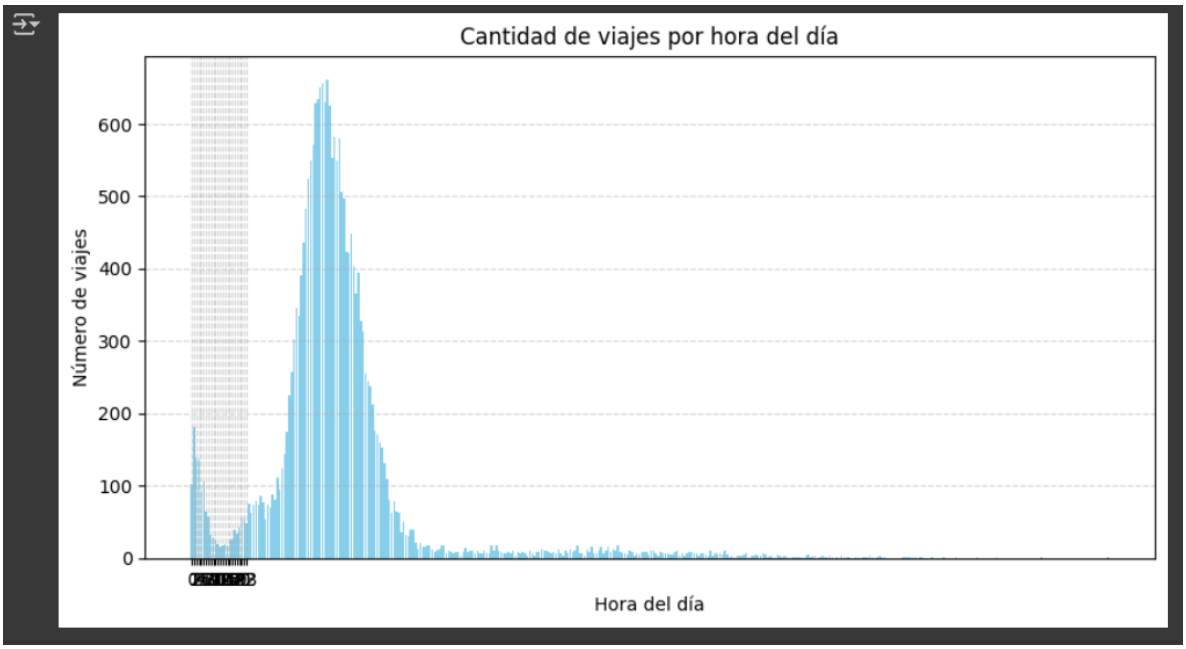
1. Distribución de viajes por hora del día

Al agrupar los datos por la columna hora (derivada de hrs_viaje), se observó una mayor concentración de viajes en dos intervalos principales:

- Horario matutino (7:00 a 9:00): Este intervalo coincide con el inicio de la jornada laboral y escolar. Refleja que muchas personas utilizan Ecobici como medio de transporte hacia sus actividades diarias.

- Horario vespertino (17:00 a 20:00): Este periodo muestra otro pico importante, indicando que los usuarios emplean las bicicletas principalmente para regresar a casa al final de la jornada.

En contraste, durante las madrugadas (de 0:00 a 5:00), la cantidad de viajes es considerablemente menor, lo cual es consistente con patrones de uso esperados para un servicio de movilidad urbana con fines cotidianos y no recreativos o nocturnos.

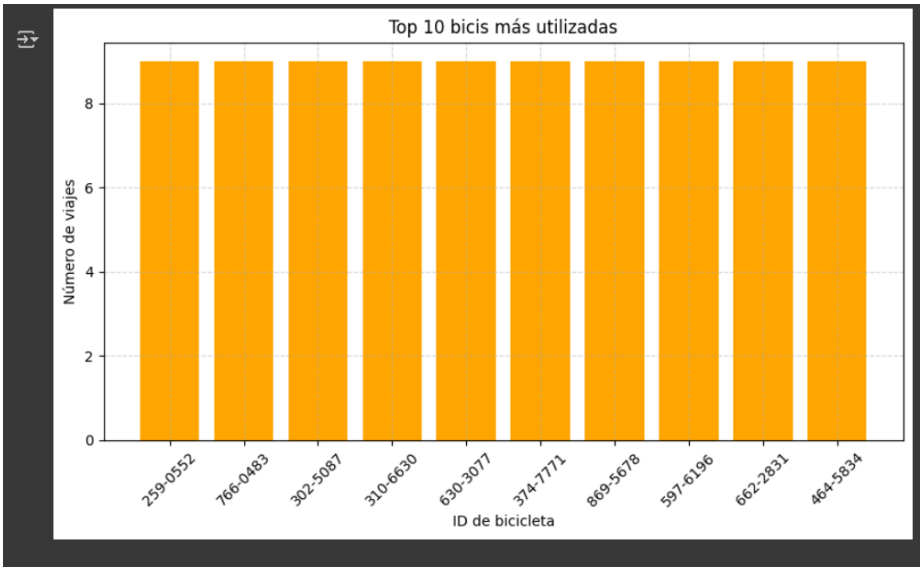


2. Bicicletas más utilizadas

El agrupamiento por identificador de bicicleta (bici) permitió conocer cuáles fueron las unidades más solicitadas. El top 10 de bicicletas revela un uso intensivo de ciertos equipos, lo que puede deberse a:

- Su localización en estaciones de alta demanda (aunque no se cuenta con datos de geolocalización para comprobarlo).
- Un mantenimiento más eficiente o una distribución más accesible.

Esto ofrece la oportunidad de generar futuras estrategias de redistribución, mantenimiento predictivo y balanceo de carga entre estaciones si se incorporan datos geográficos adicionales.



3. Duración y distancia estimada de los viajes

Aunque no fue el foco principal del análisis, se identificaron valores promedio en la duración de los viajes entre 20 y 30 minutos, lo que sugiere un uso para trayectos cortos. La columna distancia_approx refuerza esta hipótesis con valores moderados. Este tipo de comportamiento refuerza la idea de que el sistema Ecobici es adecuado para la llamada “última milla” o desplazamientos de corta distancia en zonas urbanas densas.

4. **Implicaciones para la movilidad urbana**

Los datos sugieren que Ecobici cumple un papel importante en la movilidad diaria de la Ciudad de México, particularmente como alternativa de transporte durante los horarios pico. Su uso está alineado con estrategias de movilidad sustentable, reducción del uso del automóvil y mejora de la calidad del aire.
Este tipo de análisis puede complementarse con datos adicionales sobre estaciones, zonas geográficas y clima para identificar con mayor precisión patrones de uso, optimizar rutas y ampliar la cobertura del servicio.

Reflexión

Una de las principales ventajas de trabajar con PySpark es la capacidad de distribuir el procesamiento de datos entre múltiples núcleos, incluso en entornos locales o simulados como Google Colab. En este proyecto, aunque el conjunto de datos no era masivo, se logró aplicar un enfoque distribuido a través de las transformaciones de Spark.

- **Operaciones distribuidas:** Las funciones como groupBy, count, y orderBy fueron ejecutadas por Spark en su motor distribuido, permitiendo que el agrupamiento por hora (groupBy("hora")) y por bicicleta (groupBy("bici")) se procesaran eficientemente sin necesidad de transformar los datos a un solo nodo como ocurriría con Pandas.

- **Minimización de conversiones:** Se evitó convertir los datos a Pandas hasta el final del flujo, específicamente para la visualización. Esto es importante ya que Pandas trabaja en memoria y puede generar cuellos de botella en datasets más grandes. El uso de .toPandas() se justificó únicamente para graficar, ya que Matplotlib no trabaja directamente con DataFrames Spark.

- **Repartición y escalabilidad potencial:** Aunque no se manipuló directamente el número de particiones en este ejercicio, Spark permite controlar la distribución física del trabajo, lo que sería valioso si el volumen de datos creciera exponencialmente. Las mismas operaciones de agregación se mantendrían eficientes sin necesidad de reescribir código, simplemente aumentando la infraestructura disponible.

- **Limitaciones identificadas:** El análisis se mantuvo en un solo archivo CSV, sin operaciones más complejas como join, reduceByKey o combinaciones de múltiples datasets. Aun así, la estructura del flujo está preparada para escalar, y la práctica demuestra que Spark es ideal incluso en contextos educativos o de prototipado ligero donde la eficiencia y paralelismo siguen siendo útiles.

Uso de IA

Durante el desarrollo del proyecto, se consultó IA (ChatGPT) para:

- Resolver errores relacionados con columnas.
- Explicar la instalación de Spark en Google Colab.

Repositorio

Colab

https://colab.research.google.com/drive/1DBj01XumqC7eQnwg3k0E1X67fuZQY1B9?usp=sharing

Git hub

https://github.com/LeydiMoon18/Practica-Spark