Taxis

April 1, 2020

1 TLC

1.1 Nicolás Patalagua

1.1.1 Infraestructura para Big Data - Universidad Sergio Arboleda

The New York City Taxi and Limousine Commission (TLC), created in 1971, is the agency responsible for licensing and regulating New York City's Medallion (Yellow) taxi cabs, for-hire vehicles (community-based liveries, black cars and luxury limousines), commuter vans, and paratransit vehicles. The Commission's Board consists of nine members, eight of whom are unsalaried Commissioners. The salaried Chair/ Commissioner presides over regularly scheduled public commission meetings and is the head of the agency, which maintains a staff of approximately 600 TLC employees.

*Over 200,000 TLC licensees complete approximately 1,000,000 trips each day. To operate for hire, drivers must first undergo a background check, have a safe driving record, and complete 24 hours of driver training. TLC-licensed vehicles are inspected for safety and emissions at TLC's Woodside Inspection Facility.

More info: https://www1.nyc.gov/site/tlc/about/about-tlc.page

[12]:
```
from pyspark.sql import SparkSession
from pyspark.sql import functions as F
spark = SparkSession.builder.master("local").getOrCreate()
```

[10]:
```
ObjTaxi=spark.read.csv("Taxis.csv",header=True)
```

[22]:
```
ObjZone=spark.read.csv("Zone.csv",header=True)
```

Formas de Pago

[6]:
```
ObjTaxi20= ObjTaxi.select('payment_type').distinct().show()
```

```
+------------+
|payment_type|
+------------+
|           3|
|           1|
|           4|
```

```
|           2|
+------------+
```

Taxi con mayor número de viajes

[7]:
```
ObjTaxi30=ObjTaxi.groupBy("VendorID").agg(F.count("VendorID").
 ↪alias("viajes_max"))
ObjTaxi31=ObjTaxi30.select("VendorID","viajes_max").agg(F.max("VendorID").
 ↪alias("VendorID"), F.max("viajes_max"))
ObjTaxi31.show()
```

```
+--------+---------------+
|VendorID|max(viajes_max)|
+--------+---------------+
|       4|        4382892|
+--------+---------------+
```

Número de viajes por dia en el mes de junio de 2019

[8]:
```
ObjTaxi40 = ObjTaxi.groupBy("tpep_pickup_datetime").agg(F.
 ↪count("tpep_pickup_datetime").alias("max pickup"))
ObjTaxi41 = ObjTaxi.groupBy("tpep_dropoff_datetime").agg(F.
 ↪count("tpep_dropoff_datetime").alias("max dropoff"))
ObjTaxi40.show(5)
ObjTaxi41.show(5)
```

```
+--------------------+----------+
|tpep_pickup_datetime|max pickup|
+--------------------+----------+
| 2019-06-01 00:29:17|         3|
| 2019-06-01 00:07:12|         1|
| 2019-06-01 00:52:54|         5|
| 2019-06-01 00:08:46|         3|
| 2019-06-01 00:40:46|         1|
+--------------------+----------+
only showing top 5 rows

+---------------------+-----------+
|tpep_dropoff_datetime|max dropoff|
+---------------------+-----------+
|  2019-06-01 00:22:34|          2|
|  2019-06-01 00:57:29|          4|
|  2019-06-01 01:03:00|          5|
|  2019-06-01 00:05:36|          1|
|  2019-06-01 00:29:17|          4|
+---------------------+-----------+
```

```
only showing top 5 rows
```

Área donde se recoge mayor número de pasajeros

[9]:
```
ObjZone50 = ObjZone.groupBy("Zone").agg(F.count("Zone").alias("max_pas"))
ObjZone51 = ObjZone50.select("Zone","max_pas").agg(F.max("Zone").alias("zone"),␣
 ↪F.max("max_pas"))
ObjZone51.show()
```

```
+--------------+------------+
|          zone|max(max_pas)|
+--------------+------------+
|Yorkville West|           3|
+--------------+------------+
```

Número de viajes que se dirigieron al "Bronx"

[13]:
```
ObjZone60 = ObjZone.where("`Borough` like 'Bronx%'").select("Borough",␣
 ↪"LocationID")
ObjZone61 = ObjZone60.groupBy("Borough").agg(F.count("Borough").
 ↪alias("num_trips"))
ObjZone61.show()
```

```
+-------+---------+
|Borough|num_trips|
+-------+---------+
|  Bronx|       43|
+-------+---------+
```

Número promedio de personas por viaje que se dirigen al aeropuerto JFK

[24]:
```
ObjZone70=ObjZone.where("`Zone` like 'JFK_Airport%'").select("service_zone",␣
 ↪"LocationID","Borough","Zone")
ObjTaxiZone70=ObjTaxi.join(ObjZone70, ObjTaxi.PULocationID == ObjZone70.
 ↪LocationID)
ObjTaxiZone71=ObjTaxiZone70.groupby("Zone").agg(F.avg("VendorID").
 ↪alias("prom_pass"))
ObjTaxiZone71.show()
```

```
+-----------+------------------+
|       Zone|         prom_pass|
+-----------+------------------+
|JFK Airport|1.6908959629637494|
+-----------+------------------+
```

Distancia y Costo promedio de tomar un taxi del Aeropuerto JFK a Manhattan Valley

[27]:
```
ObjZone80=ObjZone.where("`Zone` like 'JFK_Airport%'").select("service_zone",␣
 ↪"LocationID","Borough","Zone")
ObjZone81=ObjZone.where("`Zone` like 'Manhattan_Valley%'").
 ↪select("service_zone", "LocationID","Borough","Zone")
ObjTaxiZone80=ObjTaxi.join(ObjZone80, ObjTaxi.PULocationID == ObjZone80.
 ↪LocationID)
ObjTaxiZone81=ObjTaxiZone80.where("`DOLocationID` like '151%'").select("Zone",␣
 ↪"trip_distance","fare_amount","PULocationID","DOLocationID")
ObjTaxiZone82=ObjTaxiZone81.groupBy("Zone").agg(F.avg("trip_distance"),F.
 ↪avg("fare_amount"))
ObjTaxiZone82.show()
```

```
+-----------+------------------+-----------------+
|       Zone|avg(trip_distance)| avg(fare_amount)|
+-----------+------------------+-----------------+
|JFK Airport| 20.18786912751678|52.09825503355705|
+-----------+------------------+-----------------+
```

Recorrido más frencuente (entre qué zona y qué zona)

[26]:
```
ObjTaxiZone90= ObjTaxi.join(ObjZone, ObjTaxi.PULocationID == ObjZone.LocationID)
ObjTaxiZone91= ObjTaxiZone90.groupBy("Zone","PULocationID","DOLocationID").
 ↪agg(F.count("trip_distance").alias("num_viajes"))
ObjTaxiZone92= ObjTaxiZone91.groupBy("Zone","PULocationID","DOLocationID").
 ↪agg(F.max("num_viajes").alias("max_viajes"))
ObjTaxiZone93= ObjTaxiZone92.select('Zone','max_viajes').agg(F.
 ↪max("max_viajes"), F.max("zone").alias("Zona"))
ObjTaxiZone93.show()
```

```
+---------------+--------------+
|max(max_viajes)|          Zona|
+---------------+--------------+
|          47368|Yorkville West|
+---------------+--------------+
```