CASE STUDY: NYC Taxi Data Analysis

This case study is cantered on the concepts of Ingesting and Analysing Big Data on the APACHE-HIVE platform.

The dataset provided contains the detailed trip level data of trips made by taxis in New York City. Analysis is focused on the yellow taxis.

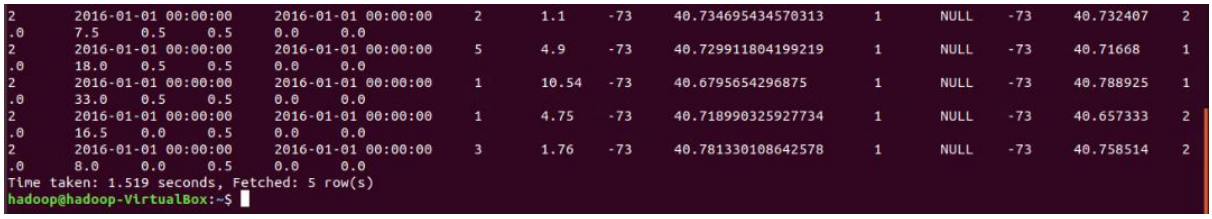
https://drive.google.com/drive/folders/1YGQBagh9X__Y65THEzuZBfiy3-lV2JBE?usp=share_link

Data Description:

- vendorid--A code indicating the TPEP provider that provided the record. 1= Creative Mobile Technologies, LLC; 2= VeriFone Inc.
- tpep_pickup_timestamp--The date and time when the meter was engaged.
- tpep_dropoff_timestamp--The date and time when the meter was disengaged.
- passenger_count--The number of passengers in the vehicle. This is a driver-entered value.
- trip_distance--The elapsed trip distance in miles reported by the taximeter.
- rate_code--The final rate code in effect at the end of the trip. 1= Standard rate 2=JFK 3=Newark 4=Nassau or Westchester 5=Negotiated fare 6=Group ride.
- store_forward_flag--This flag indicates whether the trip record was held in vehicle memory before sending to the vendor, aka store and forward, because the vehicle did not have a connection to the server. Y= store and forward trip N= not a store and forward trip.
- pickup_location--TLC Taxi Zone in which the taximeter was engaged.
- dropoff_location--TLC Taxi Zone in which the taximeter was disengaged.
- payment_type--A numeric code signifying how the passenger paid for the trip. 1= Credit card 2= Cash 3= No charge 4= Dispute 5= Unknown 6= Voided trip.
- fare_charge--The time-and-distance fare calculated by the meter.
- extra_charge--Miscellaneous extras and surcharges. Currently, this only includes the $0.50 and $1 rush hour and overnight charges.
- mta_tax_charge--$0.50 MTA tax that is automatically triggered based on the metered rate in use.
- tip_amount--Tip amount – This field is automatically populated for credit card tips. Cash tips are not included.
- tolls_charge--Total amount of all tolls paid in trip.
- improvement_surcharge--$0.30 improvement surcharge assessed trips at the flag drop. The improvement surcharge began being levied in 2015.
- total_charge--The total amount charged to passengers. It does not include cash tips.

USE Hive Queries to

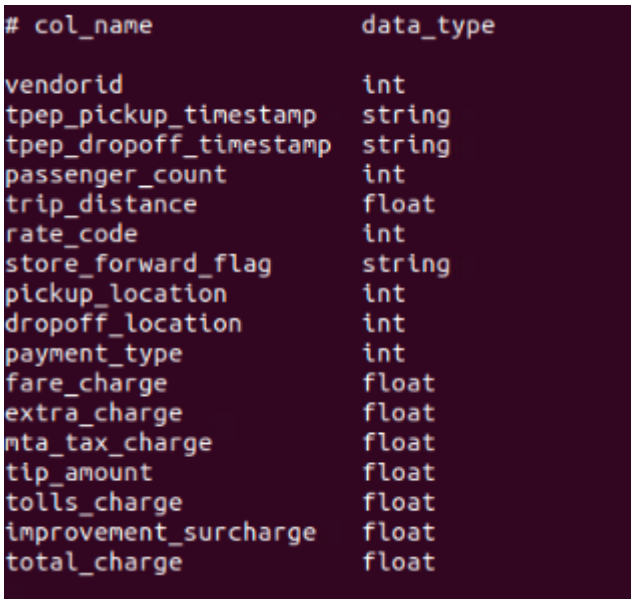
a) `Creating the intial data table titled nyc_taxifare to be used for Preliminary Analysis.`



b) Check if data table has been loaded successfully into the HIVE table.

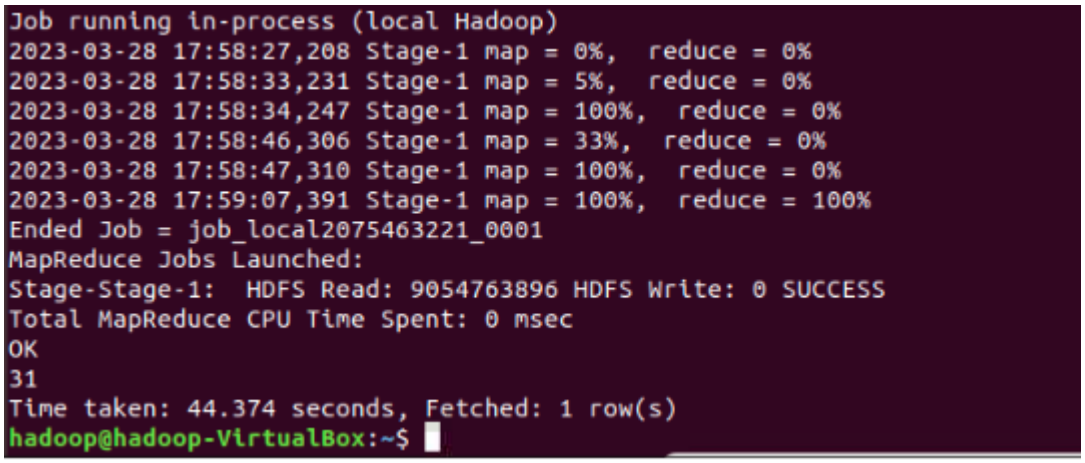
Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	hadoop	supergroup	1.59 GB	28/3/2023, 5:47:58 pm	1	128 MB	yellow_tripdata_2016-01.csv

c) Write a query that summarises the number of records of each provider.



d) Let's check if there are any records in which the pickup_timestamp is after the dropoff_timestamp. (use unix_timestamp).

Yes, there are 31 entries with the following conditions



e) EDA of components associated with Trip Details from nyc_taxifare.

f) EDA of components associated with Fare Details from nyc_taxifare.

g) Since passenger_count is an attribute registered by the driver it can be a source of Erroneous data. Write a query to analyse passenger_count.

```
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 19307105760 HDFS Write: 0 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
1       7726984 4.784183842367437       40.11393062197821
8       26      5.11692307688869        39.18467683058519
0       520     2.0803846094148377      36.43877091040978
7       22      2.6031818312000143      38.896760420365766
6       369155  2.8989140875661294      40.461123040755446
5       601079  2.985044777529876       40.39552056194798
4       210641  14.618104309498728      40.15268323445938
3       436431  2.9716486681615124      40.2201289826746
2       1561977 4.1537685762643894      40.16373631190907
9       23      2.657391301882656       40.773996601934016
Time taken: 25.938 seconds, Fetched: 10 row(s)
hadoop@hadoop-VirtualBox:~$
```

h) Checking the rate_code parameter. Write a query to show records of each rate_code.

```
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 19307105760 HDFS Write: 0 SUCCESS
Stage-Stage-2:  HDFS Read: 3417447288 HDFS Write: 0 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
1       10626315
2       225019
3       16822
4       4696
5       33688
6       102
99      216
Time taken: 17.547 seconds, Fetched: 7 row(s)
hadoop@hadoop-VirtualBox:~$
```

```
Total MapReduce CPU Time Spent: 0 msec
OK
1       2016-01-08 22:14:32     2016-01-08 22:14:32     1       0.0     0.0     0.0     99      N       0.0     0.0     3       50.0    0
.0      0.0     0.0     0.0     0.0     50.0
1       2016-01-08 22:15:10     2016-01-08 22:15:10     1       0.0     0.0     0.0     99      N       0.0     0.0     1       50.0    0
.0      0.0     0.0     0.0     0.0     50.0
1       2016-01-08 22:57:06     2016-01-08 23:33:55     1       18.3    -73.789665      40.643665       99      Y       -73.98562       4
0.761215        2       52.0    0.0     0.5     0.0     5.54    0.3     58.34
1       2016-01-08 23:32:12     2016-01-08 23:43:08     1       2.3     -73.99738       40.74184        99      Y       -73.988464      4
0.764484        1       10.0    0.5     0.5     1.5     0.0     0.3     12.8
1       2016-01-09 03:42:37     2016-01-09 03:53:28     1       3.6     -73.9869        40.75048        99      Y       -73.95271       4
0.787506        1       12.5    0.5     0.5     2.76    0.0     0.3     16.56
1       2016-01-09 11:03:06     2016-01-09 11:03:06     1       0.0     0.0     0.0     99      N       0.0     0.0     1       6.8     0
.0      0.0     0.0     0.0     0.0     6.8
1       2016-01-09 17:28:28     2016-01-09 17:39:36     1       2.5     -73.98221       40.773617       99      Y       -73.95952       4
0.777115        1       10.5    0.0     0.5     2.26    0.0     0.3     13.56
1       2016-01-09 18:58:29     2016-01-09 18:58:29     1       0.0     0.0     0.0     99      N       0.0     0.0     1       91.0    0
.0      0.0     0.0     0.0     0.0     91.0
1       2016-01-09 21:19:44     2016-01-09 21:38:28     1       4.7     -73.97722       40.78781        99      Y       -74.00225       4
0.73462 1       18.0    0.5     0.5     3.86    0.0     0.3     23.16
1       2016-01-09 22:41:50     2016-01-09 22:49:02     1       1.2     -74.00052       40.72738        99      Y       -74.00668       4
0.71579 1       7.0     0.5     0.5     1.66    0.0     0.3     9.96
Time taken: 9.03 seconds, Fetched: 10 row(s)
hadoop@hadoop-VirtualBox:~$
```

i) Checking the payment_type parameter and analyse the queries.

```
OK
1       7181476
2       3673651
3       38319
4       13411
5       1
Time taken: 17.416 seconds, Fetched: 5 row(s)
hadoop@hadoop-VirtualBox:~$
```

```
OK
1       1.90935207452889 12E7    0.23870831618405397
Time taken: 21.115 seconds, Fetched: 1 row(s)
hadoop@hadoop-VirtualBox:~$
```

j) Checking the extra_charges attribute and analyse the queries.

```
OK
0.31307568962538024
Time taken: 17.92 seconds, Fetched: 1 row(s)
hadoop@hadoop-VirtualBox:~$
```

k) Checking MTA tax attribute and analyse the count of each group of mta_tax_charge.

```
OK
0.5     10859581
0.0     43201
-0.5    4062
0.35    2
Screenshot
10.35   1
89.7    1
17.45   1
3.0     1
0.93    1
36.44   1
0.89    1
33.49   1
2.45    1
43.41   1
20.5    1
2.22    1
Time taken: 20.262 seconds, Fetched: 16 row(s)
hadoop@hadoop-VirtualBox:~$
```

l) Checking store_forward_flag parameter and analyse the count of each group of store_foreward_flag.

```
OK
N       10843625
Y       63233
Time taken: 17.375 seconds, Fetched: 2 row(s)
hadoop@hadoop-VirtualBox:~$
```

m) Checking if non-zero tip amount has been registed for cash payment trips.

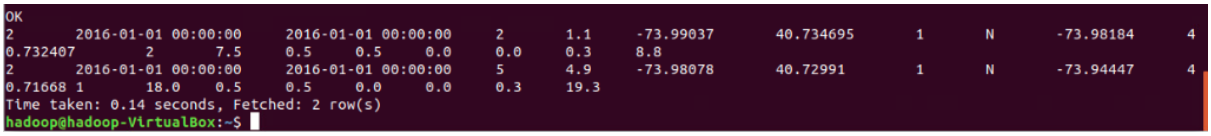
```
OK
77
Time taken: 17.086 seconds, Fetched: 1 row(s)
hadoop@hadoop-VirtualBox:~$
```

n) Checking improvement_surcharge other than $0.30 has been recorded.

Yes, recorded. Here is the count of number of transactions

```
OK
5819
Time taken: 17.408 seconds, Fetched: 1 row(s)
hadoop@hadoop-VirtualBox:~$
```

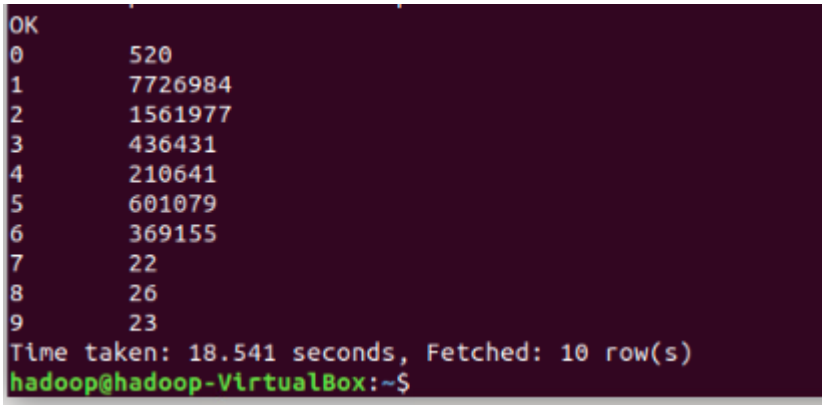
o) Creating the orc_taxifare table of ORC table format.



p) Compare the average fare_charge for November and December.

```
-- p)
SELECT
    AVG(fare_amount) AS avg_fare_charge,
    CASE
        WHEN MONTH(tpep_pickup_timestamp) = 11 THEN 'November'
        WHEN MONTH(tpep_pickup_timestamp) = 12 THEN 'December'
    END AS month
FROM nyc_taxifare
WHERE MONTH(tpep_pickup_timestamp) IN (11, 12)
GROUP BY MONTH(tpep_pickup_timestamp);
```

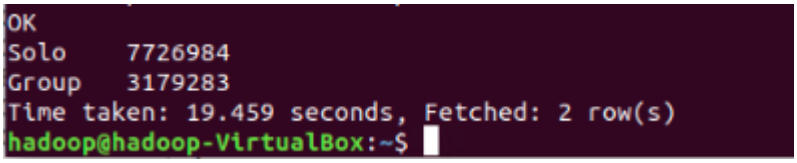
q) Explore the 'number of passengers per trip' - how many trips are made by each level of 'Passenger_count'?



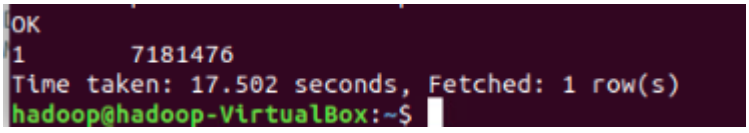
q.1: Do most people travel solo or with other people?
Ans: Yes, according to the data (passenger_count, number_of_trips), 77,26,984 people go to solo trips which is the highest number.

q.2: Let's have a look at how many trips are made by each level of passenger_count.
Ans: Refer the image snippet for reference.

r) Let's compare if the passengers prefer to travel solo [i.e, passenger_count=1] or in groups [i.e, passenger_count [2-6]]



s) Which is the most preferred mode of payment?



Ans: So according to this data set, the most preferred payment type is CREDIT CARD PAYMENT

t) What is the average tip paid? Compare the average tip with the 25th, 50th and 75th percentiles and comment whether the ‘average tip’ is a representative statistic (of the central tendency) of ‘tip amount paid’.

```
OK
2.658718172321249       1.0     2.0     3.0
Time taken: 20.1 seconds, Fetched: 1 row(s)
hadoop@hadoop-VirtualBox:~$
```

u) Explore the ‘Extra’ (charge) variable - what is the fraction of total trips where an extra charge is levied?
Let us observe the extra_charge attribute in a grouped table w.r.t number of records.

```
OK
No Extra Charge 5712207 0.5237
Extra Charge    5194651 0.4763
Time taken: 24.636 seconds, Fetched: 2 row(s)
hadoop@hadoop-VirtualBox:~$
```

v) The number of trips where the extra_charge was levied is marginally lower than the number of trips for which it was not.
Let us write a query to compare the Fraction of trips for which the extra_charge was levied.

```
OK
Extra Charge Levied     20      1.833708662934825E-6
Extra Charge Levied     3       2.7505629944022377E-7
Extra Charge Levied     1       9.1685433146741 26E-8
Extra Charge Levied     1       9.1685433146741 26E-8
Extra Charge Not Levied 1       9.1685433146741 26E-8
Extra Charge Not Levied 1       9.1685433146741 26E-8
Extra Charge Not Levied 1       9.1685433146741 26E-8
Extra Charge Levied     1       9.1685433146741 26E-8
Extra Charge Levied     6       5.501125988804475E-7
Extra Charge Levied     25      2.2921358286685313E-6
Extra Charge Not Levied 1       9.1685433146741 26E-8
Extra Charge Not Levied 1       9.1685433146741 26E-8
Extra Charge Levied     1       9.1685433146741 26E-8
Extra Charge Levied     2       1.8337086629348251E-7
Extra Charge Levied     1635787 0.14997783963080843
Extra Charge Levied     1       9.1685433146741 26E-8
Extra Charge Not Levied 513     4.7034627204278264E-5
Extra Charge Levied     1       9.1685433146741 26E-8
Extra Charge Levied     1       9.1685433146741 26E-8
Extra Charge Not Levied 1       9.1685433146741 26E-8
Extra Charge Not Levied 1       9.1685433146741 26E-8
Extra Charge Levied     1       9.1685433146741 26E-8
Extra Charge Levied     1       9.1685433146741 26E-8
Extra Charge Levied     44      4.0341590584566 15E-6
Extra Charge Levied     1       9.1685433146741 26E-8
Extra Charge Not Levied 1       9.1685433146741 26E-8
Extra Charge Levied     1       9.1685433146741 26E-8
Extra Charge Levied     2       1.8337086629348251E-7
Extra Charge Levied     3558725 0.3262832430751368
Extra Charge Levied     3       2.7505629944022377E-7
Extra Charge Levied     10      9.1685433146741 25E-7
Extra Charge Not Levied 5710200 0.523542160354522
Extra Charge Not Levied 1486    1.3624455365605 75E-4
Extra Charge Levied     1       9.1685433146741 26E-8
Extra Charge Levied     12      1.1002251977608 95E-6
Time taken: 19.453 seconds, Fetched: 35 row(s)
hadoop@hadoop-VirtualBox:~$
```

w) Create five buckets of ‘tip paid’: [0-5), [5-10), [10-15), [15-20) and >=20. Calculate the percentage share of each bucket.

```
set hive.exec.dynamic.partition=true;
set hive.exec.dynamic.partition.mode=nonstrict;
set hive.enforce.bucketing = true;

drop table if exists bucket_nyc_taxifare;

create external table if not exists bucket_nyc_taxifare(VendorID int,
tpep_pickup_datetime timestamp, tpep_dropoff_datetime timestamp,
passenger_count int, trip_distance int, pickup_longitude double, pickup_latitude double,
RatecodeID int, store_and_fwd_flag varchar(10), dropoff_longitude double,
dropoff_latitude double, payment_type int, fare_amount double,
extra double, mta_tax double, tip_amount double, tolls_amount double,
improvement_surcharge double, total_amount double,tip_grp varchar(20))
clustered by (tip_grp) into 5 buckets
row format delimited
fields terminated by ','
lines terminated by '\n';

insert overwrite table bucket_nyc_taxifare
select * from (select *,case when tip_amount >=0 and tip_amount <5 then "zerotofive"
when tip_amount >=5 and tip_amount <10 then "fivetoten"
when tip_amount >=10 and tip_amount <15 then "tentofive"
when tip_amount >=15 and tip_amount <20 then "fifteentotwenty"
when tip_amount >=20 then "morethantwenty" end as tip_grp from nyc_taxifare) as t1 ;
```

x) Which month has a greater average ‘speed’ - November or December?

```
OK
58.57720656281733       NULL
Time taken: 124.634 seconds, Fetched: 1 row(s)
hadoop@hadoop-VirtualBox:~$
```

y) Analyse the average speed of the most happening days of the year i.e., 31st December (New Year’s Eve) and 25th December (Christmas Eve) and compare it with the overall average.

```
Total MapReduce CPU Time Spent: 0 msec
OK
0.017071486652877766    12.895158275500817      11.37050364795488
Time taken: 109.447 seconds, Fetched: 1 row(s)
```

z) Create partitioning of table using total charge as total_charge category.

```
        Time taken to load dynamic partitions: 0.74 seconds
        Time taken for adding to write entity : 0.003 seconds
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 7346040252 HDFS Write: 7271019105 SUCCESS
Stage-Stage-5:  HDFS Read: 3422676485 HDFS Write: 3387197117 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
Time taken: 132.848 seconds
```