# Data Ingestion and Processing:
# Hive Group Case Study

- By Mukundan & Anindo 6th January 2020

## Problem Statement :

The New York City Taxi & Limousine Commission (TLC) has provided a dataset of trips made by the taxis in the New York City. The detailed trip-level data is more than just a vast list of taxi pickup and drop off coordinates.

The records include fields capturing pick-up and drop-off dates/times, pick-up and drop-off locations (location coordinates of the starting and ending points), trip distances, itemized fares, rate types, payment types, driver-reported passenger counts etc.

The data used was collected and provided to the NYC Taxi and Limousine Commission (TLC) by technology providers authorized under the Taxicab & Livery Passenger Enhancement Programs (TPEP/LPEP).

## Objective :

- The purpose of this dataset is to get a better understanding of the taxi system so that the city of New York can improve the efficiency of in-city commutes. Several exploratory questions can be asked about the travelling experience for passengers.

- We have to consider the data of yellow taxis for November and December of the year 2017.

## Data Dictionary :

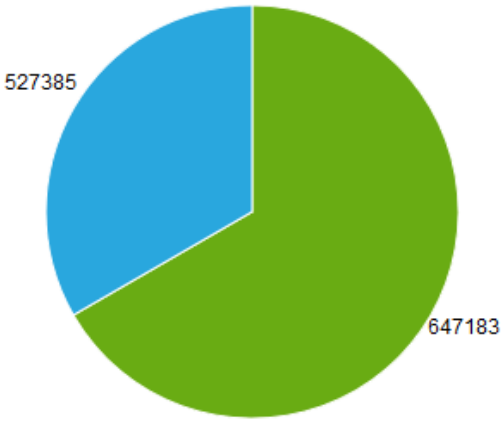
| Sl No | Field Name | Data Description |
|---|---|---|
| 1 | vendorid | A code indicating the TPEP provider that provided the record. 1= Creative Mobile Technologies, LLC; 2= VeriFone Inc. |
| 2 | tpep_pickup_timestamp | The date and time when the meter was engaged. |
| 3 | tpep_dropoff_timestamp | The date and time when the meter was disengaged. |
| 4 | passenger_count | The number of passengers in the vehicle. This is a driver-entered value. |
| 5 | trip_distance | The elapsed trip distance in miles reported by the taximeter. |
| 6 | 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. |
| 7 | store_forward_flag | This flag indicates whether the trip record was held in vehicle memory before sending to the vendor, aka store & forward, because the vehicle did not have a connection to the server. Y= store and forward trip N= not a store and forward trip. |
| 8 | pickup_location | TLC Taxi Zone in which the taximeter was engaged. |
| 9 | dropoff_location | TLC Taxi Zone in which the taximeter was disengaged. |
| 10 | 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. |
| 11 | fare_charge | The time-and-distance fare calculated by the meter. |
| 12 | extra_charge | Miscellaneous extras and surcharges. Currently, this only includes the $0.50 and $1 rush hour and overnight charges. |
| 13 | mta_tax_charge | $0.50 MTA tax that is automatically triggered based on the metered rate in use. |
| 14 | tip_amount\| Tip amount | This field is automatically populated for credit card tips. Cash tips are not included. |
| 15 | tolls_charge | Total amount of all tolls paid in trip. |
| 16 | improvement_surcharge | $0.30 improvement surcharge assessed trips at the flag drop. The improvement surcharge began being levied in 2015. |
| 17 | total_charge | The total amount charged to passengers. It does not include cash tips. |

## Basic Data Quality Checks

**Question 1:**

***How many records has each TPEP* provider provided? Write a query that summarises the number of records of each provider. (*Taxi-Passenger Experience Enhancement Program Provider):***

| Vendor_ID | Vendor_Type | No_Records |
|---|---|---|
| 1 | Creative Mobile Technologies | 527,385 |
| 2 | VeriFone Inc. | 647,183 |
| Total Records | | 1,174,568 |




***Question 2:***

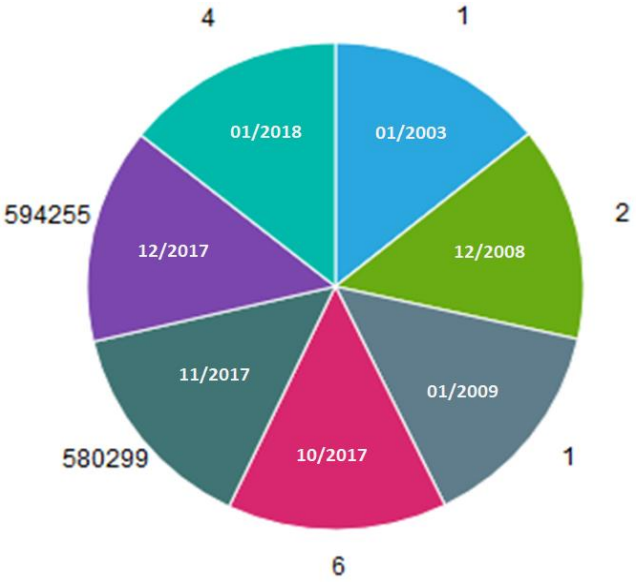
***The data provided is for months November and December only. Check whether the data is consistent, and if not, identify the data quality issues. Mention all data quality issues in comments.***

***Answer***

- Understand the datetime column as timestamp and get the Month and Year from datetime column.
- We have both tpep_pickup_timestamp and tpep_dropoff_timestamps are available we will set tpep_pickup_timestamp as our reference column as it is the first point of contact with passenger. Only trips that registered a tpep_pickup_timestamp and tpep_dropoff_timestamp during November and December 2017 will be considered.
- This implies that only trips that have been started and completed between November to December 2017 will be considered for our analysis.

**PICK-UP YEAR & TIME :**

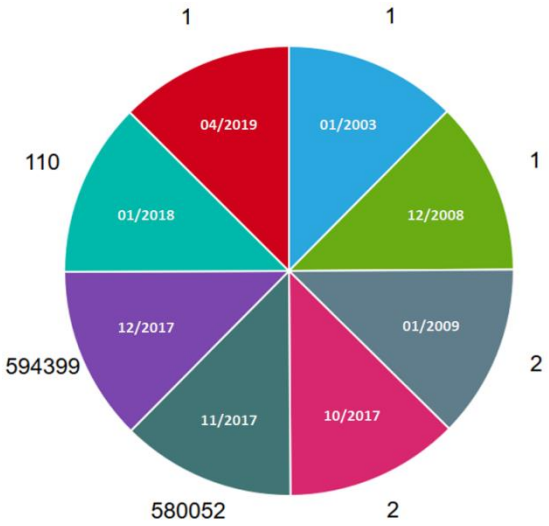
| pickup_year | pickup_month | no_records |
|---|---|---|
| 2003 | 1 | 1 |
| 2008 | 12 | 2 |
| 2009 | 1 | 1 |
| 2017 | 10 | 6 |
| 2017 | 11 | 580,299 |
| 2017 | 12 | 594,255 |
| 2018 | 1 | 4 |
| Total Records | | 1,174,568 |




The tpep_pickup_timestamp has the year range from 2003 to 2018. Since our task is to observ the trips for Nov to Dec 2017. There are 14 non-adhering records based on tpep_pickup_timestamp.

**DROP- OFF YEAR & TIME:**

| dropoff_year | dropoff_month | no_records |
|---|---|---|
| 2003 | 1 | 1 |
| 2008 | 12 | 1 |
| 2009 | 1 | 2 |
| 2017 | 10 | 2 |
| 2017 | 11 | 580,052 |
| 2017 | 12 | 594,399 |
| 2018 | 1 | 110 |
| 2019 | 4 | 1 |
| Total Records | | 1,174,568 |

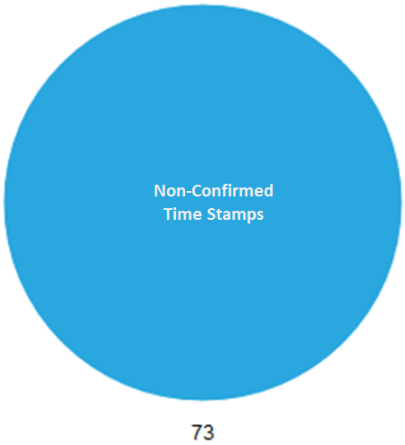



The tpep_dropoff_datetime has the year range from 2003 to 2019. Since our task is to observe the trips for Nov to Dec 2017.

There are 117 non-adhering records based on tpep_dropoff_datetime.

**NON-CONFIRMED TIMESTAMPS:**

Review the data for pickup_timestamp is after the dropoff_timestamp.



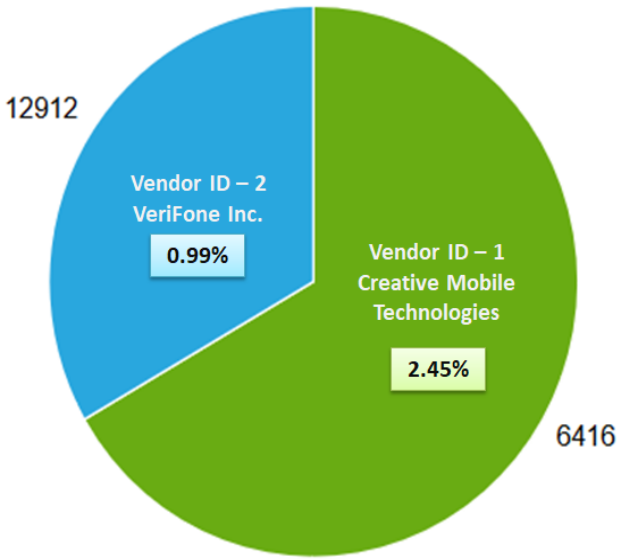

The results reveal that there are 73 records with tpep_pickup_timestamp after the tpep_dropoff_timestamp.

## Question 3:

***You might have encountered unusual or erroneous rows in the dataset. Can you conclude which vendor is doing a bad job in providing the records using different columns of the dataset? Summarise your conclusions based on every column where these errors are present. Ex: There are unusual passenger count, i.e. 0 which is unusual.***

***Answer***

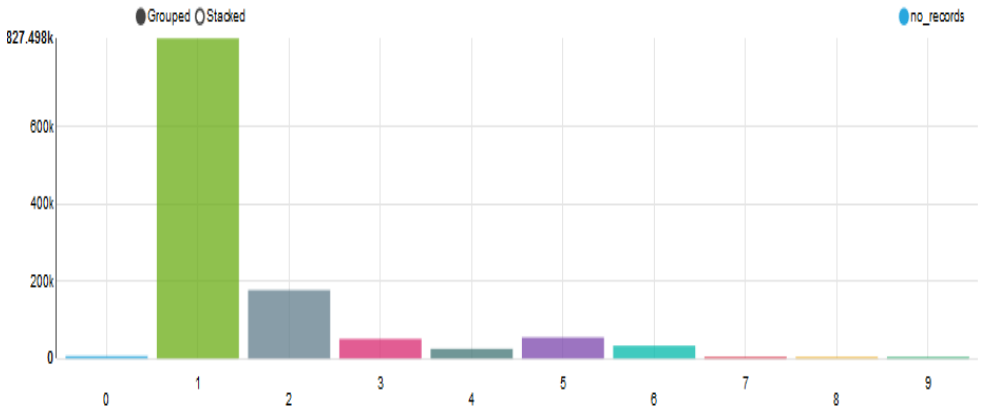
### NON-CONFIRMED RECORDS:




| Vendor _ID | Vendor_Type | No _Records | nonconf _records | % nonconf _records |
|---|---|---|---|---|
| 1 | Creative Mobile Technologies | 527,385 | 12,912 | 2.45% |
| 2 | VeriFone Inc. | 647,183 | 6,416 | 0.99% |
| Total Records | | 1,174,568 | 19,328 | |

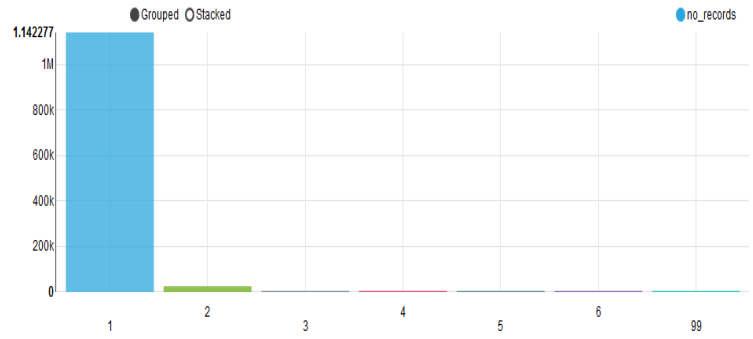
### PASSENGER COUNT :

| No_of_PAX | No_Records |
|---|---|
| 0 | 6,824 |
| 1 | 827,498 |
| 2 | 176,872 |
| 3 | 50,693 |
| 4 | 24,951 |
| 5 | 54,568 |
| 6 | 33,146 |
| 7 | 12 |
| 8 | 3 |
| 9 | 1 |
| Total Records | 1,174,568 |




- *The passenger_count values range between 0 to 9 clearly there are some data quality issues in this attribute.*
- *Trips cannot be registed and paid for with 0 passengers [These are due to some refunds or abnormailties] and a taxi cannot accomodate 9 passengers.*
- *Therefore we must set some limitations to this parameter. The maximum passengers allowed in a yellow taxicab by law is four (4) in a four (4) passenger taxicab or five (5) passengers in a five (5) passenger taxicab, except that an additional passenger must be accepted if such passenger is under the age of seven (7) and is held on the lap of an adult passenger seated in the rear. Source: http://www.nyc.gov/html/tlc/html/faq/faq_pass.shtml*
- *Therefore only passenger_count between 1-6 will be treated as valid records.*

## RATE_CODE PARAMETER :

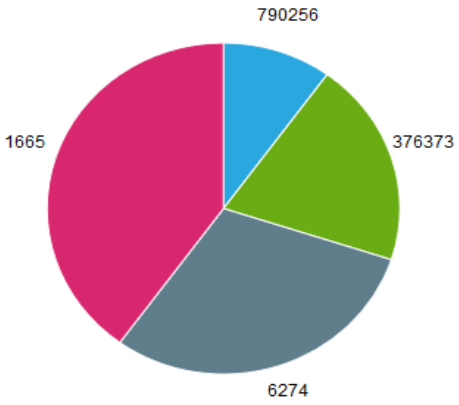



| Sl # | Rate_ Code | No_ Records | Rate_Encoding |
|---|---|---|---|
| 1 | 1 | 1,142,277 | Standard rate |
| 2 | 2 | 25,338 | JFK |
| 3 | 3 | 2,562 | Newark |
| 4 | 4 | 586 | Nassau or Westchester |
| 5 | 5 | 3,793 | Negotiated fare |
| 6 | 6 | 3 | Group ride |
| 7 | 99 | 9 | NA |
| Total Records | | 1,174,568 | |

- From the above result there are 7 distinct rate codes while the data dictionary limits it to 6 distinct codes between 1-6.
- The 9 records under rate_code 99 will be treated as non-conforming

## PAYMENT_TYPE PARAMETER :

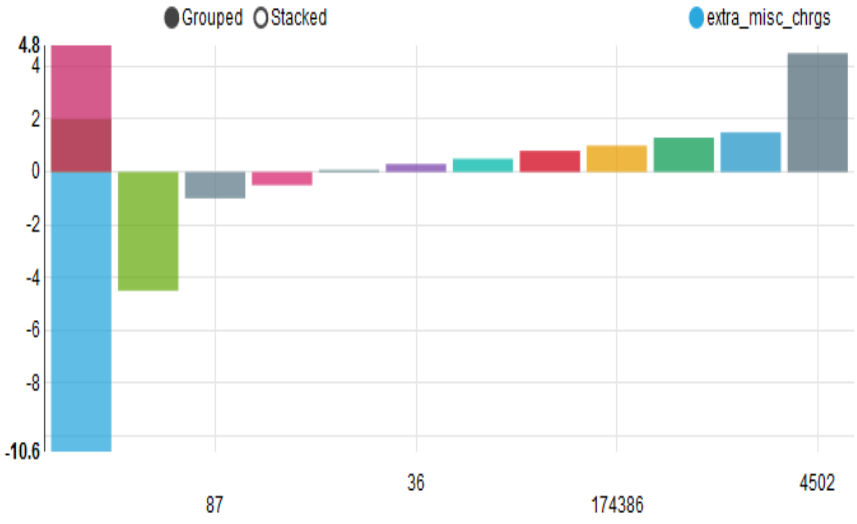
| Sl No | Payment _ Type | No _Records | Payment _By |
|---|---|---|---|
| 1 | 1 | 790,256 | Credit card |
| 2 | 2 | 376,373 | Cash |
| 3 | 3 | 6,274 | No charge |
| 4 | 4 | 1,665 | Dispute |
| Total Records | | 1,174,568 | |




There are 4 distinct payment_types that are in agreement with the data-disctionary.

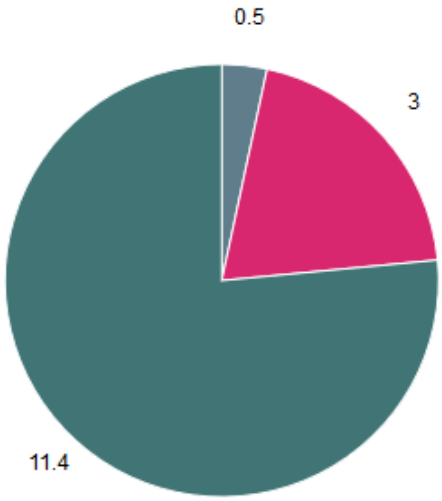
## EXTRA_CHARGES ATTRIBUTE :

| Sl No | Extra _Misc _Chrgs | No _Records |
|---|---|---|
| 1 | $10.60 | 1 |
| 2 | $4.50 | 5 |
| 3 | $1.00 | 87 |
| 4 | $0.50 | 193 |
| 5 | $0.00 | 631,872 |
| 6 | $0.30 | 36 |
| 7 | $0.50 | 363,454 |
| 8 | $0.80 | 15 |
| 9 | $1.00 | 174,386 |
| 10 | $1.30 | 13 |
| 11 | $1.50 | 2 |
| 12 | $2.00 | 1 |
| 13 | $4.50 | 4,502 |
| 14 | $4.80 | 1 |
| Total # | $0.10 | 1,174,568 |




- There are 14 distinct extra_charge values in the dataset Ranging between -$10.6 and $4.8. However, the extra_charge is a surcharge that can only take up $0.5 and $1 during rush hour and traffic, otherwise it is $0. Therefore, all other values will be treated as non-conformities.

## MTA TAX ATTRIBUTE :

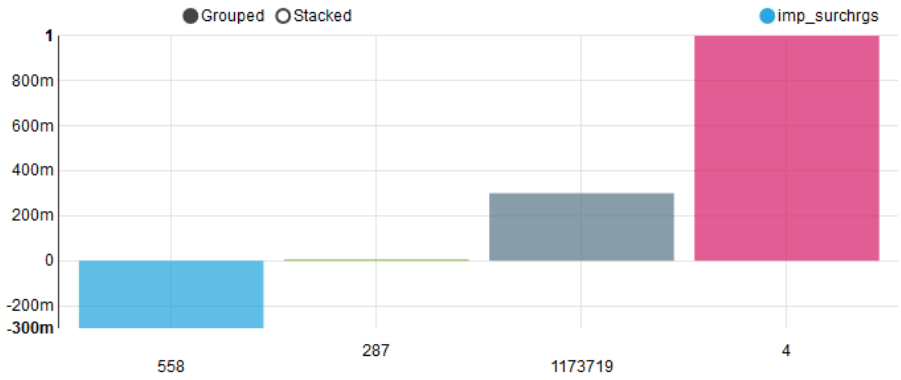



| Sl No | MTA _TAX | No _Records |
|---|---|---|
| 1 | $0.50 | 544 |
| 2 | $0.00 | 5,197 |
| 3 | $0.50 | 1,168,823 |
| 4 | $3.00 | 3 |
| 5 | $11.40 | 1 |
| Total # | $14.40 | 1,174,568 |

- *There are 5 distinct mta_tax_charge values in the dataset Ranging between -$0.5 and $11.4.*
- *The data dictionary specified that mta_tax_charge of $0.5 is triggered based on metered rate in use.*
- *Therefore, it can only take up two values $0 or $0.5 all other values will be treated as non-conformities.*

## IMPROVEMENT_SURCHARGE ATTRIBUTE :

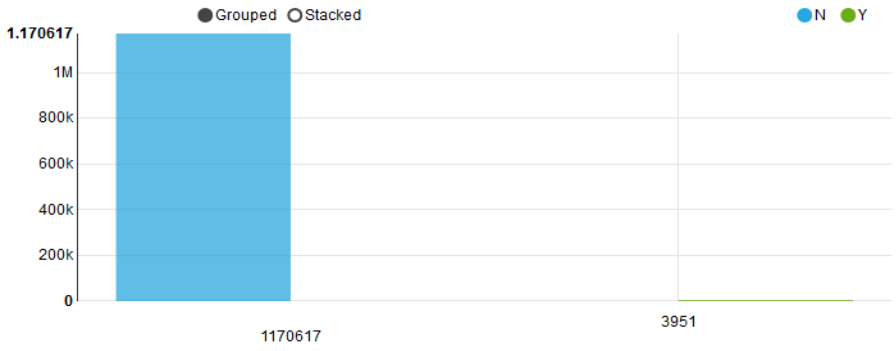
| Sl No | Imp_ SurChrgs | No _Records |
|---|---|---|
| 1 | $0.30 | 558 |
| 2 | $0.00 | 287 |
| 3 | $0.30 | 1,173,719 |
| 4 | $1.00 | 4 |
| Total # | $1.00 | 1,174,568 |




- *There are 4 distinct values of improvement_surcharge Rangeing between -$0.3 and $1.*
- *The improvement_surcharge of $0.3 began being levied on assessed trips at flagdrop this means that the improvement_surcharge can only take up $0 or $0.3 All other values of improvement_surcharge will be treated as non-conformity.*

## STORE_FORWARD_FLAG PARAMETER :

| Sl No | str_and _fwd_flag | No _Records |
|---|---|---|
| 1 | N | 1,170,617 |
| 2 | Y | 3,951 |
| Total # | | 1,174,568 |




- *There are only 2 store_forward_flag parameter values [Y and N] which is inline with the specified limits with 0.34% of the total records being stored and the passed to the servers.*

## EXPLORATORY DATA ANALYSIS

### EDA of Trip Details from Data_NYCTaxiFare :

***Observations :*** *After performing the EDA based on Trip Details*

| Descriptions | Trip Values |
|---|---|
| no_of_rcrds | 1,174,568 |
| no_of_tpep_vendors | 2 |
| oldest_pickup_timestamp | 1/1/2003 |
| recent_pickup_timestamp | 1/1/2018 |
| oldest_dropoff_timestamp | 1/1/2003 |
| recent_dropoff_timestamp | 24/4/2019 |
| min_pax_pertrip | 0 |
| max_pax_pertrip | 9 |
| avg_pax_pertrip | 1.621837135 |
| min_trip_distance | 0 |
| max_trip_distance | 126.41 |
| avg_trip_distance | 2.871185006 |
| no_of_rate_codes | 7 |
| str_and_fwd_flag_types | 2 |
| no_of_pickup_zones | 246 |
| no_of_dropoff_zones | 260 |
| no_of_payment_types | 4 |

1. *There are a total of 117,4568 records in the dataset*
2. *There are 2 TPEP vendors*
3. *The tpep_pickup_timestamps ranges between 1st Jana 2003 to 01 Jana 2018. This is a nonconformity.*
4. *The tpep_drop_timestamps ranges between 1st January 2003 to 24 April 2019. This is a nonconformity*
5. *The passenger per trip count ranges between Min 0 to Max 9 withan Avg of 1.62 passengers per trip*
6. *The trip distances range between 0 to 126.41 miles. A trip of 0 miles should not be charged and 126.41 miles seems like a outlier. However we will retain it. Average distance per trip is at 2.87 miles.*
7. *There are 7 distinct rate_codes in the dataset when the data_dictionary limits it to 6. This is a nonconformity.*
8. *There are 246 logged pickup_locations and 260 logged dropoff_locations.*
9. *There are 4 distinct payment_type in the datase.*

### EDA of FARE Details from Data_NYCTaxiFare :

***Observations :*** *After performing the EDA based on FARE Details*

| Descriptions | Fare Values |
|---|---|
| min_fare_chrg | -200 |
| max_fare_chrg | 650 |
| avg_fare_chrg | 12.99541063 |
| min_extra_chrg | -10.6 |
| max_extra_chrg | 4.8 |
| avg_extra_chrge | 0.320292141 |
| types_of_mta_tax_chrg | 5 |
| min_mta_tax_chrg | -0.5 |
| max_mta_tax_chrg | 11.4 |
| avg_mta_tax_chrg | 0.497340214 |
| min_tip_amt | -1.16 |
| max_tip_amt | 450 |
| avg_tip_amt | 1.85312543 |
| min_toll_chrg | -5.76 |
| max_toll_chrg | 895.89 |
| avg_toll_chrg | 0.327426884 |
| types_of_surcharge | 4 |
| min_surchrg | -0.3 |
| max_surchrg | 1 |
| avg_surchrg | 0.299644039 |
| min_total_chrg | -200.8 |
| max_total_chrg | 928.19 |
| avg_total_chrg | 16.29586697 |

***1.*** The fare_charge attribute Range: -$200 and $650 | Average: $12.99. The trips with fare_charges <= 0 will be treated as Nonconformities.

***2.*** The extra_charge attribute Range: -$10.6 and $4.8 | Average: $0.32. The extra_charge is a surcharge that can only take up $0.5 and $1 during rush hour and traffic, otherwise it is $0. other values will be treated as non-conformities.

***3.*** The mta_tax_charge attribute Range: -$0.5 and $11.4 | Average: $0.497. There are 5 distinct values of mta_tax_charge.

***4.*** The tip_amount attribute Range: -$1.16 and $450 | Average: $1.85. Tip tip_amounts are automatically populated for credit card paid trips but cash tips are not recorded.

However, a negative tip amount is peculiar [refund of trip or abnormality] therefore all records with tip amount<0 will be treated as non-conforming.

***5.*** The tolls_charge arribute Range: -$5.76 and $895.89 | Average: $0.327. Negative toll charges seem peculiar and may indicate a refund transaction or abnormality. Therefore, all records with tolls_charge <0 will be treated as a non-conformity.

***6.*** The improvement_surcharge attribute Range: -$0.3 and $1 | Average: $0.299. The dataset has 5 distinct improvement_surcharges.The improvement_surcharge of $0.3 began being levied on assessed trips at flagdrop this means that the improvement_surcharge can only take up $0 or $0.3.

***7.*** The total_charge attribute Range: -$200.8 and $928.19 | Average: $16.29.The negative total_charges may be logged due to refunds or disputed trips. This is an abnormality and will not be considered. Only records with total_charge >0 will be considered for our analysis.

## ORC TABLE CREATION

*The Next Step after Performing the basic quality checks and EDA is to set the Hive Parameters before creating the ORC table.*

```
SET hive.exec.dynamic.partition = true;
SET hive.exec.dynamic.partition.mode = nonstrict;
SET hive.exec.max.dynamic.partitions=100000;
SET hive.exec.max.dynamic.partitions.pernode=100000;
SET hive.execution.engine=mr;
```

- *Create the ORC_NYCTaxiFare :*
- *Populate the ORC_NYCTaxiFare PARTITION(mnth, m_day)*

*Now the partition is been set to perform further analysis*

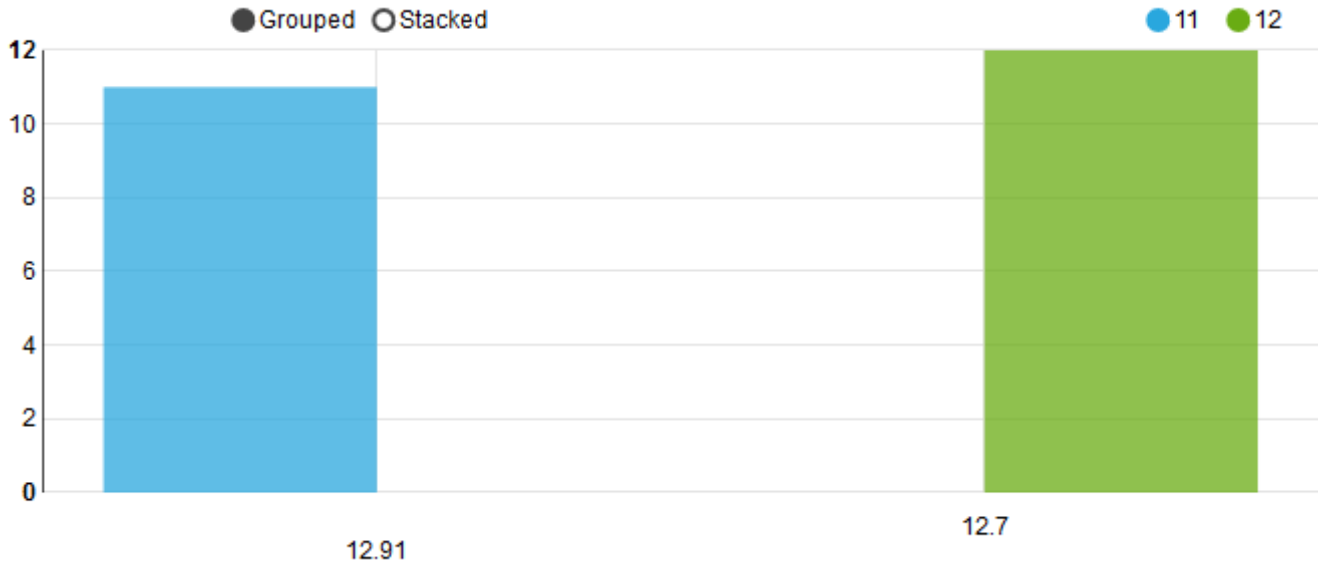
## *ANALYSIS – I*

**Q1. Compare the overall average fare per trip for November and December.**

***Answer :***

**Group the table by month and average fare_charge**

| Sl No | Month _of_year | Avg _fare_chrg |
|---|---|---|
| 1 | 11 | 12.91 |
| 2 | 12 | 12.70 |
| Avg Fare Chrgs | | 12.81 |




***Observations :***

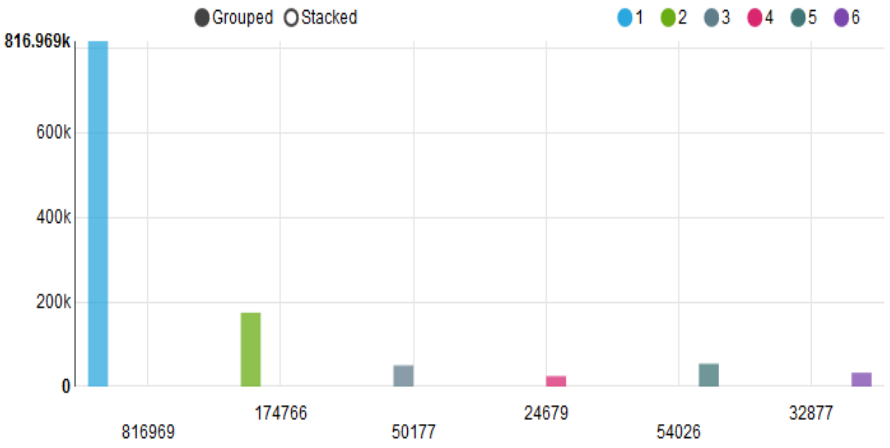
- *November Average fare_charge: $12.91*
- *December Average fare_charge: $12.70*
- *Therefore the Average fare_charge recorded during November is 1.22% higher than the average fare_charge recorded in December.*

Q2. Explore the 'number of passengers per trip' - how many trips are made by each level of 'Passenger_count'? Do most people travel solo or with other people?

***Answer :***

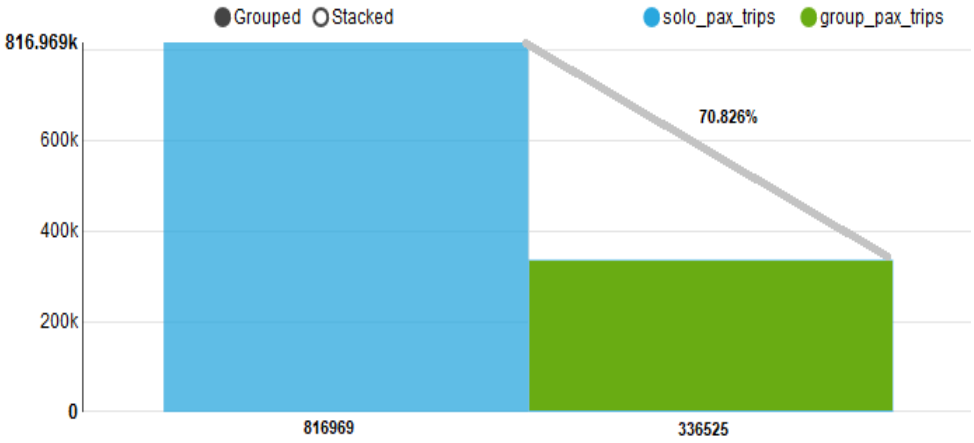
*No of trips made by each level of passenger_count*

| Sl No | no_of_pax | no_records |
|---|---|---|
| 1 | 1 | 816,969 |
| 2 | 2 | 174,766 |
| 3 | 3 | 50,177 |
| 4 | 4 | 24,679 |
| 5 | 5 | 54,026 |
| 6 | 6 | 32,877 |




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

| Trip Type | Values |
|---|---|
| solo_pax_trips | 816,969 |
| group_pax_trips | 336,525 |
| solo_trips_ pct_total_trips | 70.826% |




***Observations :***

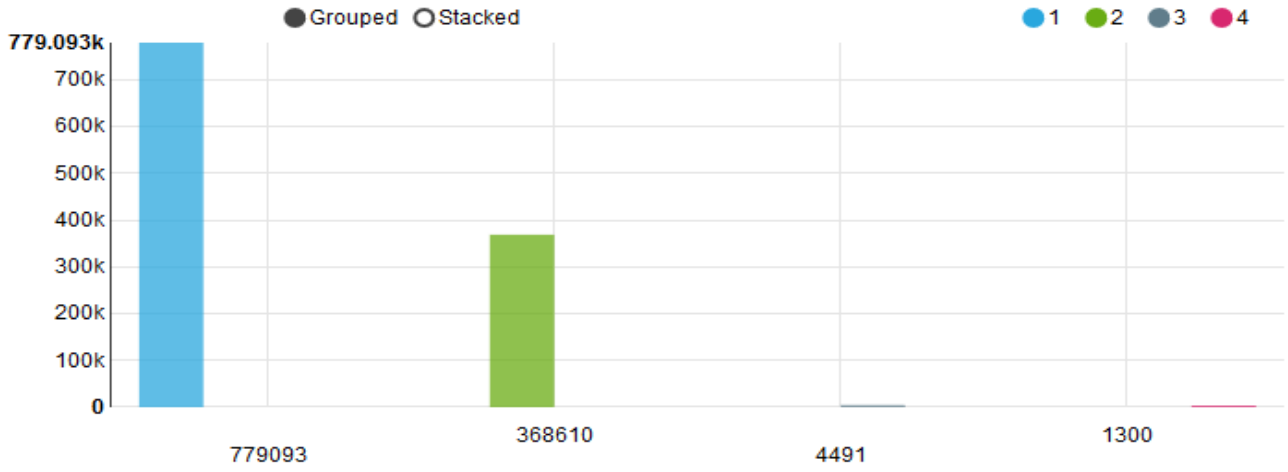
- *No. Solo PAX Trips : 816,969*
- *No. Group PAX Trips : 336,525*
- *% of trips with Solo PAX w.r.t Total No. of trips : 70.826%*
- *From the results it is clear that in 70.826% of all trips, people prefer to travel Solo.*

Q3. *Which is the most preferred mode of payment?*

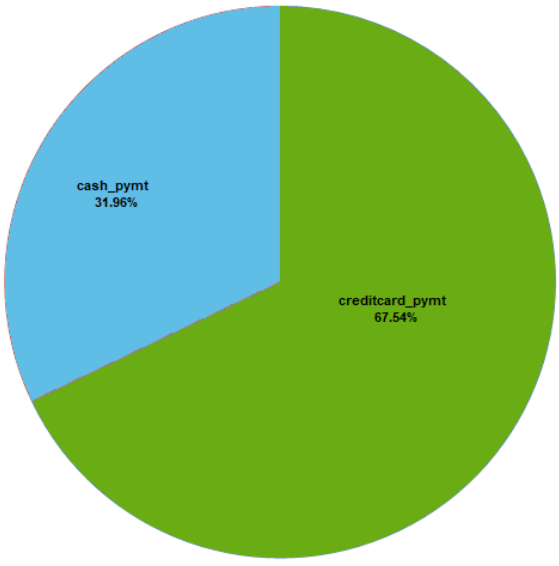
***Answer :***

*Group the table by Payment Type and No. of records:*

| Sl No | Payment _ Type | No _Records | Payment _By | % by Payment Type |
|---|---|---|---|---|
| 1 | 1 | 779,093 | Credit Card | 67.54% |
| 2 | 2 | 368,610 | Cash | 31.96% |
| 3 | 3 | 4,491 | No charge | 0.39% |
| 4 | 4 | 1,300 | Dispute | 0.11% |




| Payment Description | Values |
|---|---|
| creditcard_pymt | 779,093 |
| cash_pymt | 368,610 |
| total_number_ trips | 1,153,494 |
| pct_ paidby_creditcard | 67.54% |
| pct_paidby _cash | 31.96% |



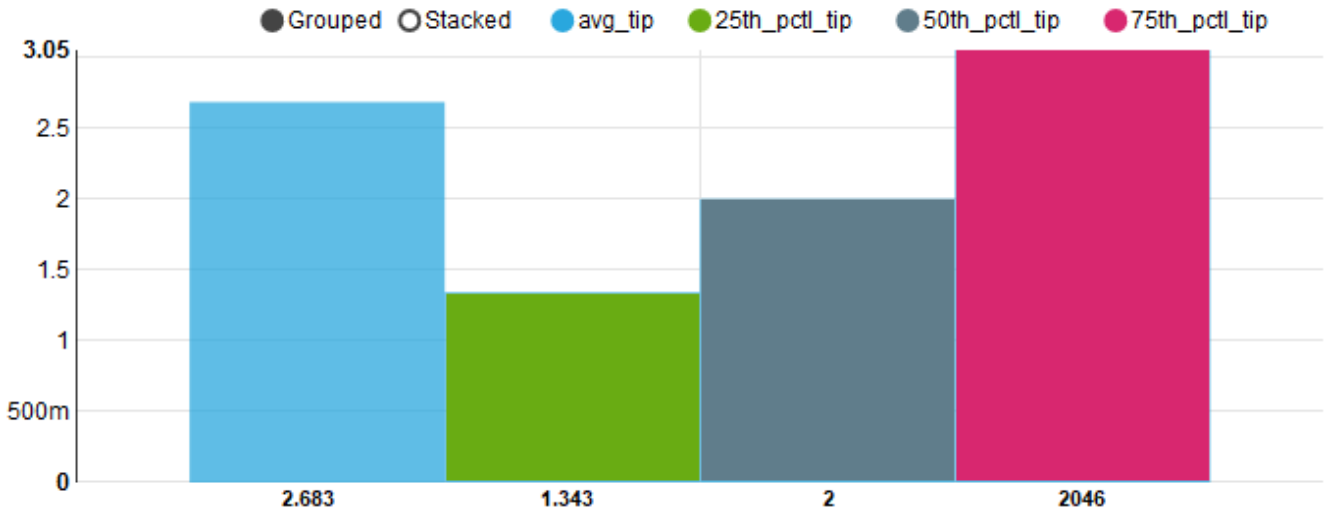

***Observations :***

- Payment Type (PYMT) 1 is Credit Card Payment
  Total Trips by Credit Card Payment = 779,093 which consitutes around 67.54% of the Trip
- Payment Type (PYMT) 2 is Cash Payment
  Total Trips by Cash Payment = 368,610 which consitutes around 31.96% of the Trip
- Credit Card is the preferred payment method

**Q4.** What is the average tip paid per trip? Compare the average tip with the 25th, 50th and 75th percentiles and comment whether the 'average tip'

***Answer :***

- *Earlier analysis the tip_amount recorded for cash is 0.*
- *We need to remove these fields before we compute the central tendency as these records are synonymous to missing records.*
- *Therefore we will remove all records where payment_type=2 [Cash Payments]*




| Tips Description | Values |
|---|---|
| avg_tip | 2.683 |
| 25th_pctl_tip | 1.343 |
| 50th_pctl_tip | 2 |
| 75th_pctl_tip | 3.05 |
| dist_tip_amt | 2,046 |

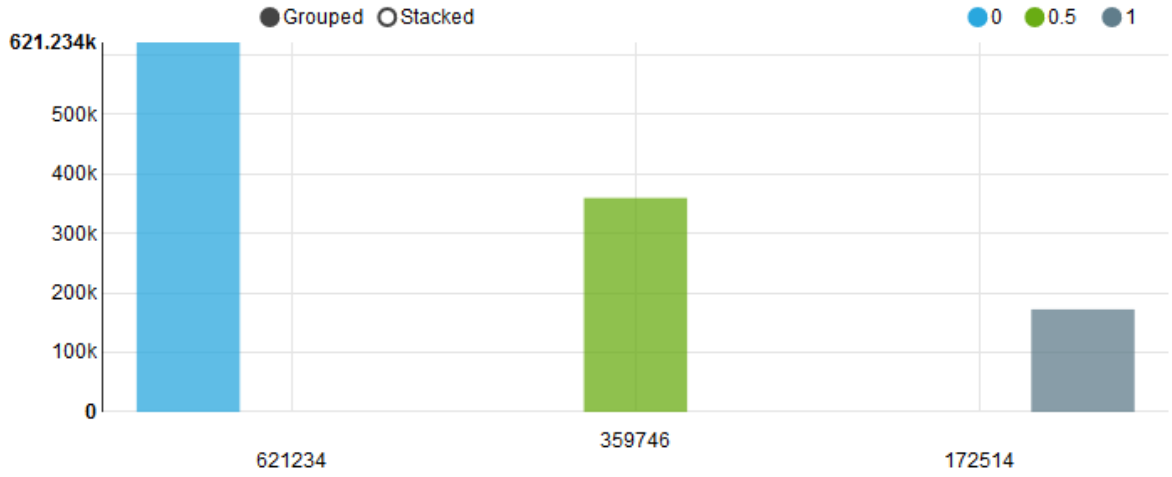
***Observations :***

- *Here, since tip_amount is stored as double data type we have to use percentile_approx() instead of percentile().*
- *From the documentation: percentile_approx(DOUBLE col, p [, B]) .Returns an approximate pth percentile of a numeric column (including floating point types) in the group. The B parameter controls approximation accuracy at the cost of memory.*
- *Higher values yield better approximations, and the default is 10,000. When the number of distinct values in col is smaller than B, this gives an exact percentile value.*
- *Since the number of distinct tip amounts 2,046<10,000 percentile_approx() returns the exact percentile value.*
- *There is $2.683 difference of the Average_Tip*

Q5. *Explore the 'Extra' (charge) variable - what fraction of total trips have an extra charge is levied?*

***Answer :***

*Group extra_charge by No of records*



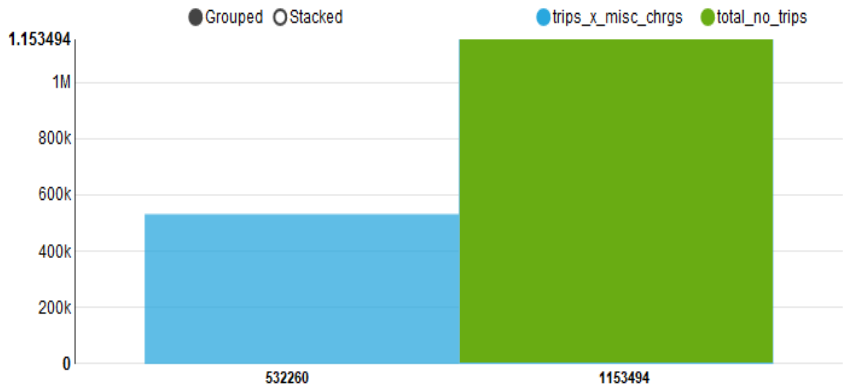

| Sl No | x_misc _chrgs | no_records |
|---|---|---|
| 1 | 0 | 621,234 |
| 2 | 0.500 | 359,746 |
| 3 | 1 | 172,514 |

***Observations :***

- *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.*

| Trip Descriptions | Values |
|---|---|
| trips_x_misc_chrgs | 532,260 |
| total_no_trips | 1,153,494 |
| fraction_trips_extra_chrgs | 0.46143 |




***Observations :***

- *No of Trips for which the Extra_Misc_Charge was levied: 532,260*
- *Total Number of Trips: 1,153,494*
- *Fraction of trips for which the Extra_Misc_Charge was levied: 0.46143 [or 46.143%].*

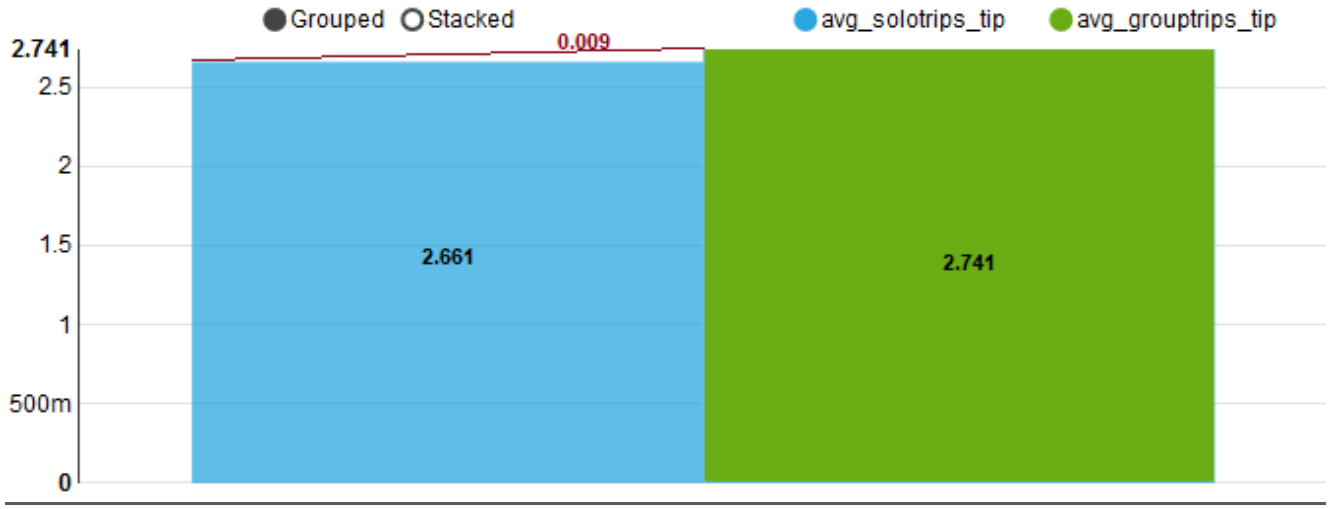
## *ANALYSIS – II*

Q5. *Explore the 'Extra' (charge) variable - what fraction of total trips have an extra charge is levied?*

*Answer :*

*Correlation between the tip_amount and number of passengers.*

| Trip Types | Values |
|---|---|
| corr_paxcnt_vs_tipamt | 0.009 |
| avg_solotrips_tip | 2.661 |
| avg_grouptrips_tip | 2.741 |



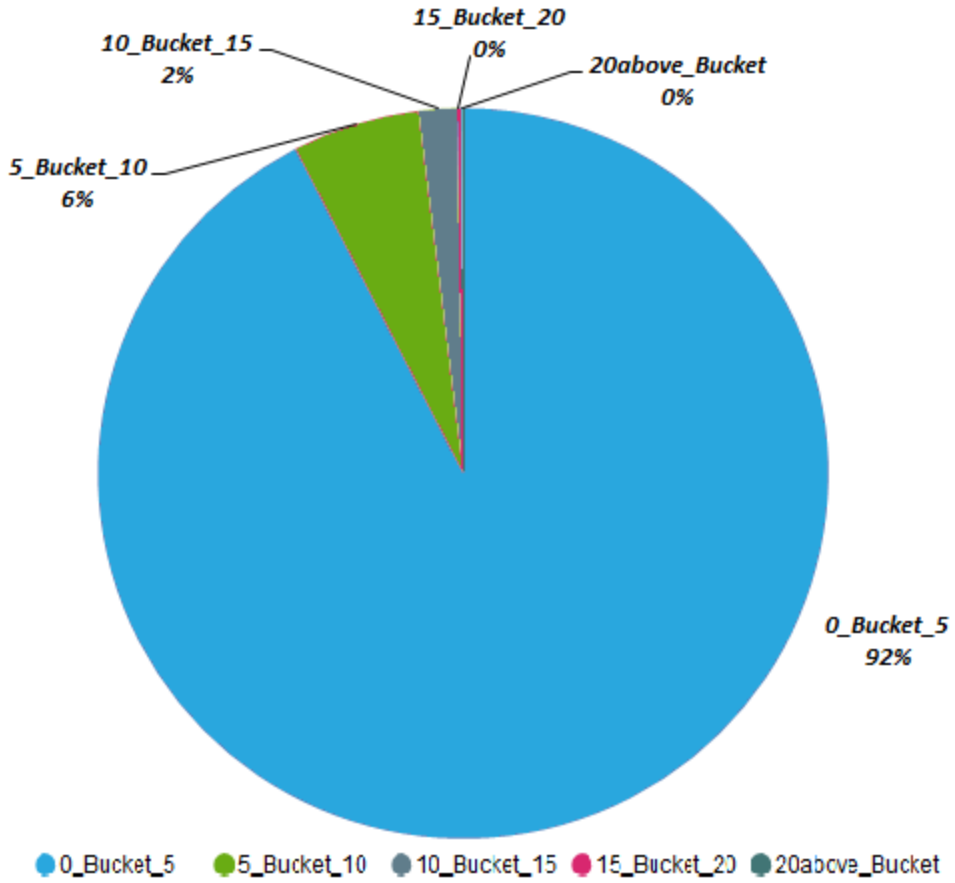

*Observations :*

- *Correlation between Passenger Count and Tip_Amount: +0.009*
- *Avg. Tip for Solo Trips : $2.661*
- *Avg. Tip for Group Trips: $2.741*
- *There is a weak +ve correlation between Passenger Count and Tip_Amount hence Average Tip are consistent with the obtained correlation value.*
- *Passengers traveling in group gives higher tip amount.*

*Q2. Segregate the data into five segments of 'tip paid': [0-5), [5-10), [10-15) , [15-20) and >=20. Calculate the percentage share of each bucket (i.e. the fraction of trips falling in each bucket).*

***Answer :***

| Sl No | tip_bucket | No _records | Total _no _records | tip_bucket _fraction _overall |
|---|---|---|---|---|
| 1 | 0_Bucket_5 | 1,065,876 | 1,153,494 | 0.92404 |
| 2 | 5_Bucket_10 | 65,032 | 1,153,494 | 0.05638 |
| 3 | 10_Bucket_15 | 19,410 | 1,153,494 | 0.01683 |
| 4 | 15_Bucket_20 | 2,160 | 1,153,494 | 0.00187 |
| 5 | 20above_Bucket | 1,016 | 1,153,494 | 0.00088 |




***Observations :***

- *These results are expected as the tip_amount is logged as $0 for all Cash paid trips where (payment_type=2), which constitutes to about 32% of all records in the dataset. Therefore if a total objective view is required over.*

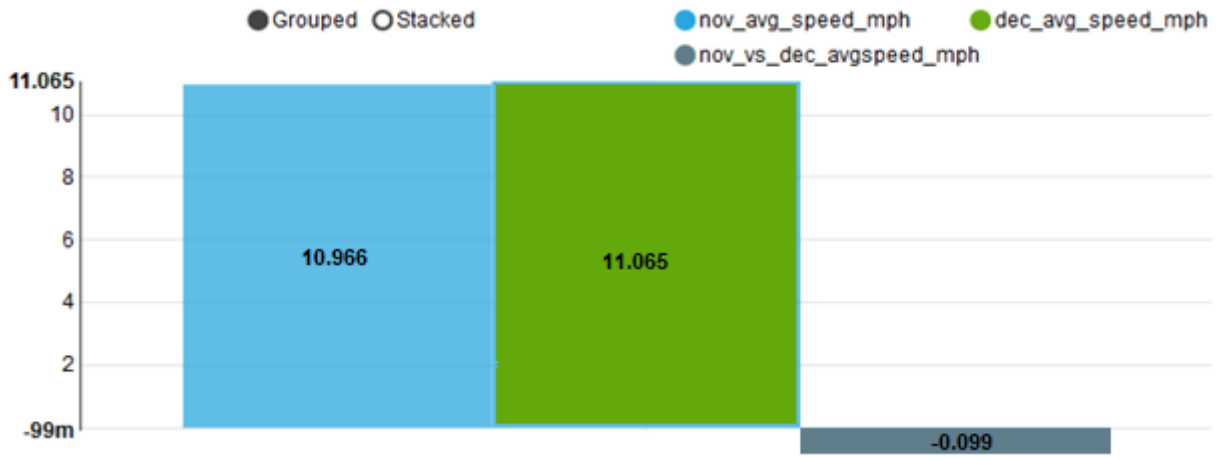
*Q3. Which month has a greater average 'speed' - November or December? Note that the variable 'speed' will have to be derived from other metrics. -Hint: You have columns for distance and time.*

***Answer :***

*Lets Compare the Average Speed of Taxi's for November 2017 and December 2017 by taking the columns distance and time.*

*The unix time stamp is a way to track time as a running total of seconds. This count starts at the Unix Epoch on January 1st, 1970 at UTC. Therefore, the unix time stamp is merely the number of seconds between a particular date and the Unix Epoch. It should also be pointed out (thanks to the comments from visitors to this site) that this point in time technically does not change no matter where you are located on the globe. This is very useful to computer systems for tracking and sorting dated information in dynamic and distributed applications both online and client side.*
[Ref : https://www.unixtimestamp.com/index.php]

| nov_avg_speed_mph | dec_avg_speed_mph | nov_vs_dec_avgspeed_mphh |
|---|---|---|
| 10.966 | 11.065 | -0.099 |



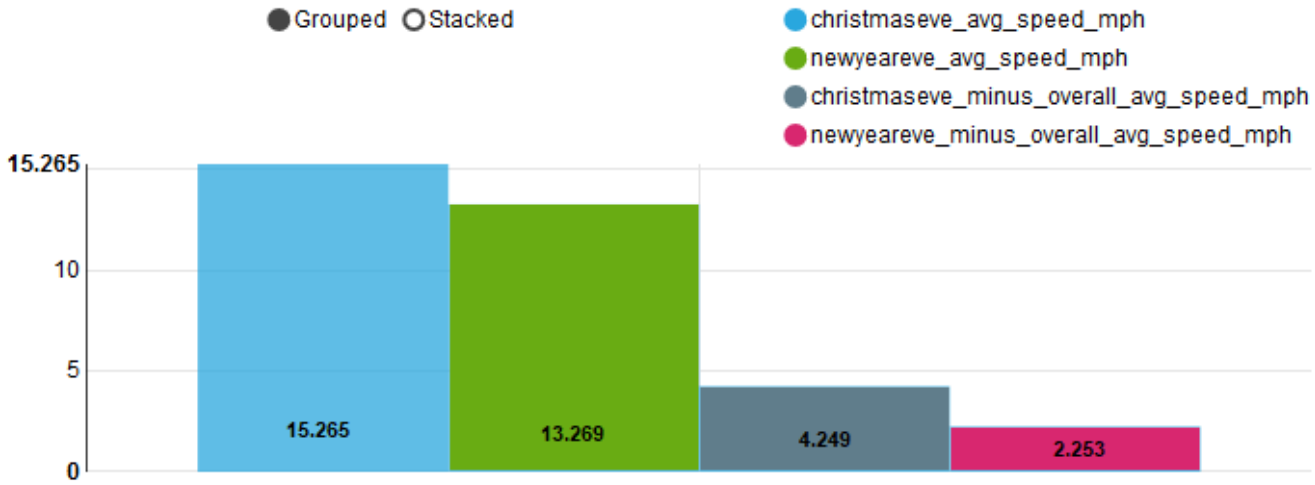

***Observations :***

- *November Month Average Speed: 10.966 MPH*
- *December Month Average Speed: 11.065 MPH*
- *Average Speed of November - Average Speed of December: -0.099*
- *The Average Speed of taxis in December is greater than their Average Speed in November.*

*Q4. Analyse the average speed of the most happening days of the year, i.e. 31st December (New year's eve) and 25th December (Christmas) and compare it with the overall average.*

***Answer :***

*Lets Compare the Average Speed of Taxi's for November 2017 and December 2017 by taking the columns distance and time.*




***Observations :***

| | |
|---|---|
| **overall_average_speed_mph** ***(Overall Average Speed for November and December Combined)*** | **11.016** |
| | |
| **christmaseve_avg_speed_mph** ***(Average Speed on Christmas Eve)*** | **15.265** |
| **christmaseve_minus_overall_avg_speed_mph** | **4.249** |
| | |
| **newyeareve_minus_overall_avg_speed_mph** ***(Average Speed on New Year Eve)*** | **2.253** |
| **newyeareve_avg_speed_mph** | **13.269** |