The NYC Yellow Taxi dataset, provided by the New York City Taxi and Limousine Commission (TLC), tracks every trip taken by yellow taxis in the city. For 2021 and 2022, it includes details like where and when trips occurred, how far they went, how much they cost, and how people paid. This data is super helpful for understanding taxi usage patterns. In this Phase 2, I am discovering and understanding the data for 2021 and 2022. This document explains the data source, size, structure, useful columns, my exploration process, and how it supports my project

**Where to Find the Data**

* **Source:** The data is provided by the New York City Taxi and Limousine Commission (TLC) as Parquet files.
* Location: Available at <https://data.cityofnewyork.us/Transportation/2021-Yellow-Taxi-Trip-Data/m6nq-qud6/about_data>
* Files: Monthly files like yellow\_tripdata\_2021-01.parquet to yellow\_tripdata\_2021-12.parquet (2021) and yellow\_tripdata\_2022-01.parquet to yellow\_tripdata\_2022-12.parquet (2022)

**Format and Coverage**

* Format: Parquet, which is good for big data and works with PySpark or Pandas.
* Coverage: Includes all yellow taxi trips for 2021 and 2022 in New York City.

**Size and Row Count**

* **2021:**
* **Total Size:** About 508mb in Parquet
* **Row Count:** 30.9 million(309Lakh) rows for 2021 year)
* **2022:**
* **Total Size:** Similar, 813.2mb in Parquet
* **Row Count:** 39.7 million(307Lakh) rows for 2022 year).

**Column Descriptions:-**

1. VendorID (Integer)

A number that tells which taxi company recorded the trip.

2.tpep\_pickup\_datetime (Timestamp / DateTime)

The exact date and time when the taxi ride started.

3.tpep\_dropoff\_datetime (Timestamp / DateTime)

The exact date and time when the taxi ride ended.

4.passenger\_count (Integer)

The number of people who were in the taxi during the trip.

5.trip\_distance (Float)

How far the taxi traveled during the ride (in miles).

6.RatecodeID (Integer)

A number showing which fare rule was used (like standard fare, airport rate, etc.).

7.store\_and\_fwd\_flag (String / Character)

Shows if the trip data was saved first (because of no internet) and sent later.

"Y" = Yes, saved and sent later

"N" = No, sent right away

8.PULocationID (Integer)

A code for the area where the taxi picked up the passenger.

9.DOLocationID (Integer)

A code for the area where the taxi dropped off the passenger.

10.payment\_type (Integer)

Shows how the passenger paid for the ride (like card, cash, etc.).

11.fare\_amount (Float)

The main fare for the ride, based on distance and time.

12.extra (Float)

Any extra charges, like for rush hour or night rides.

13.mta\_tax (Float)

A flat $0.50 tax that’s added to most taxi rides.

14.tip\_amount (Float)

The tip the passenger gave (only shown if they paid by card).

15.tolls\_amount (Float)

Any tolls the taxi paid during the trip (like bridge or tunnel fees).

16.improvement\_surcharge (Float)

A fixed fee of $0.30 added to every ride to improve taxi services. This started in 2015.

17.total\_amount (Float)

The full amount the passenger had to pay, including fare, tax, tips, and other charges.

18.congestion\_surcharge (Float)

Extra fee added for rides in busy traffic areas to help reduce congestion.

In 2022 year dataset one new column include

airport\_fee (Float)

Extra charge added when a taxi picks up a passenger from JFK or LaGuardia airport.

**Understanding:-**

**Usefulness for Analysis:-**

* Very Useful (7 Columns): tpep\_pickup\_datetime, tpep\_dropoff\_datetime, trip\_distance, PULocationID, DOLocationID, fare\_amount, total\_amount. These are the backbone for time, location, and financial studies.
* Useful (4 Columns): passenger\_count, payment\_type, tip\_amount, congestion\_surcharge. These add depth to passenger and payment analysis.
* Somewhat Useful (3 Columns): RatecodeID, extra, tolls\_amount. These are relevant for specific cases like fare types or toll routes.
* Not Very Useful (4 Columns): VendorID, store\_and\_fwd\_flag, mta\_tax, improvement\_surcharge. These have limited variation or fixed values, reducing their impact.

**Time Analysis**

* **tpep\_pickup\_datetime (Floating Timestamp):** The date and time when the taxi meter starts (e.g., "2021-01-01 00:30:10"). This can show yearly trends (e.g., more trips in December) or daily patterns (e.g., rush hours).
* **tpep\_dropoff\_datetime (Floating Timestamp):** The date and time when the meter stops (e.g., "2021-01-01 01:00:10"). Combined with pickup time, it can reveal trip durations or end-time trends.

**Location Insights**

* **PULocationID (Number):** A code for the TLC Taxi Zone where the trip begins (e.g., 100). It identifies pickup areas and can map NYC’s busiest zones using zone data.
* **DOLocationID (Number):** A code for the TLC Taxi Zone where the trip ends (e.g., 200). It identifies drop-off areas and helps track destination hotspots.

-NYC Taxi Zones: PULocationID and DOLocationID are codes (1 to 263) linked to TLC Taxi Zones

(I have download this file .csv lookup file).

**Discovery**

* Accessing the 1-to-263 zone data reveals the geographic scope of the 2021 dataset, enhancing my exploration.

**Financial Study**

* **fare\_amount (Number):** The base fare calculated by the meter (e.g., $10.50). It can estimate revenue per trip.
* **total\_amount (Number):** The total charge to passengers (e.g., $15.30), including extras but excluding cash tips. It provides a broader revenue pattern.