**Business Case Study**

1. Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset

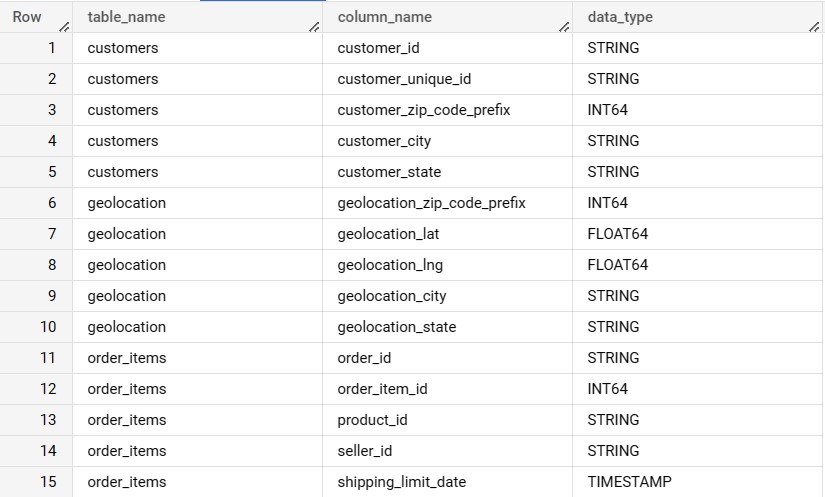
* Data type of columns of all tables

I have imported the dataset into big query and named the dataset as ‘Target’

SELECT table\_name,column\_name,data\_type

FROM Target.INFORMATION\_SCHEMA.COLUMNS

ORDER BY table\_name;



The above query will list all the 8 tables available in the dataset along with all the columns and its datatypes.

Looking at the dataset we see there are 3 tables related to orders ,1 table for reviews, 1 table for customers ,1 table for sellers ,1 table for payments and 1 table for location

* Time period for which the data is given?

select Min(cast(order\_purchase\_timestamp as date)) as Min\_Time,

Max(cast(order\_purchase\_timestamp as date)) as Max\_Time  from `Target.orders`



SELECT DATE\_DIFF(Max\_Time ,Min\_Time,Month) as Total\_Time FROM

(SELECT Min(cast(order\_purchase\_timestamp as date)) as Min\_Time ,

Max(cast(order\_purchase\_timestamp as date)) as Max\_Time  FROM `Target.orders`)



The results show the total time period for the data collected is for the duration of 25 months. These data were fetched from the ‘orders’ table which captures all the information related to order made by the customers between 4th September 2016 and 10th October 2018.

* Cities and States of customers ordered during the given period

with cte\_Orders as (

select customer\_id,order\_purchase\_timestamp,

Min(order\_purchase\_timestamp)as Min\_Time ,Max(order\_purchase\_timestamp) as Max\_time

from `Target.orders`

group by customer\_id , order\_purchase\_timestamp

)

select customer\_city, customer\_state from `Target.customers` as c

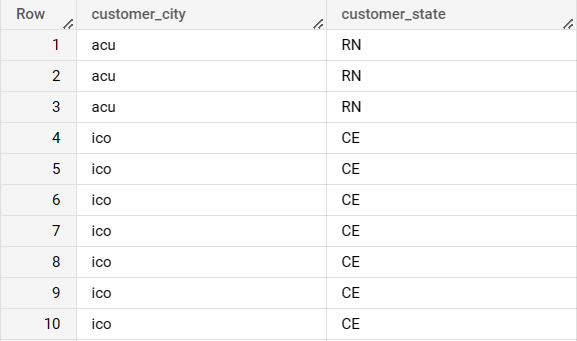
inner Join cte\_Orders o on

c.customer\_id = o.customer\_id

where o.order\_purchase\_timestamp between Min\_Time and Max\_Time;

The above query list all the cities and states of customers who have ordered between the period of 4thSeptember 2016 and 10th October 2018.

Sample output



2. In-depth Exploration:

* Is there a growing trend on e-commerce in Brazil?

select

extract(Year from order\_purchase\_timestamp)as purchase\_year,

FORMAT\_DATE("%h", DATE (order\_purchase\_timestamp))as purchase\_month,

count(order\_id)as Total\_orders

from `Target.orders`

group by purchase\_year,purchase\_month

order by purchase\_year,purchase\_month

The query above will list the total orders month on month from September 2016 to October 2018. The Total number of orders is 99441.

* In the beginning the orders were low from September to December 2016 and then the trend started showing growth from Feb 2017 and sales started increasing month on month
* The trend continued to see a rise in orders throughout 2017 and reached peak orders at 7544 during November 2017
* The sales stays consistent for most of the 2018 and recorded a 7000+ orders on two occasions (Jan 2018 and March 2018) and later started to see rapid decline in orders during the month of September and October which accounted for only 20 orders combined.
* The business saw 9% increase in the number of orders from 2017 to 2018.

Sample Output:



* What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)?

with cte\_time\_interval as(

Select time\_hour,Count(order\_id) as Total\_orders

from( select order\_id,

EXTRACT (HOUR from order\_purchase\_timestamp)as time\_hour

from `Target.orders`)

group by time\_hour

order By time\_hour

)

select Sum(Total\_orders) as no\_of\_orders,

case

when time\_hour >=0 and time\_hour <=6 then "Dawn"

when time\_hour >=7 and time\_hour <=12 then "Morning"

when time\_hour >=13 and time\_hour <=18 then "Afternoon"

else "Night"

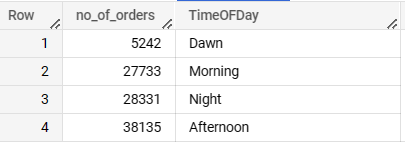
end as TimeOfDay

from cte\_time\_interval

group by TimeOfDay

order by no\_of\_orders

Sample Output:



Note: I have considered the timings for Dawn, Morning, Afternoon and Night as mentioned during the intro session

* The number of orders can change based on how the timings has been categorized into different intervals of the day.
* Here we can see from the snapshot of the result that maximum orders were placed during the afternoon and minimum orders were placed during the dawn.
* For Morning and Night, the orders were comparable to each other.
* If we look into hourly orders then the minimum orders were received at 5am and maximum orders were received at 4pm.

3. Evolution of E-commerce orders in the Brazil region:

* Get month on month orders by states

Select c.customer\_state,

extract(Year from order\_purchase\_timestamp)as purchase\_year,

extract(Month from order\_purchase\_timestamp)as purchase\_month,

FORMAT\_DATE("%B", DATE (order\_purchase\_timestamp))as month\_name,

count(o.order\_id) as Total\_orders\_per\_state,

from `Target.orders` as o

left join `Target.customers` as c

  on o.customer\_id= c.customer\_id

group by

    c.customer\_state,month\_name,purchase\_month,purchase\_year

order by c.customer\_state,purchase\_year,purchase\_month

* The sales for State SP,RJ have seen a steady rise in their orders month over month
* Some of the States have not seen any growth in their orders and still remains in single digits.

Sample Output: 

* Distribution of customers across the states in Brazil

1. select customer\_state, count(customer\_id) as Customers from `Target.customers` group by customer\_state

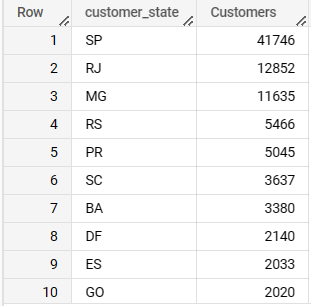
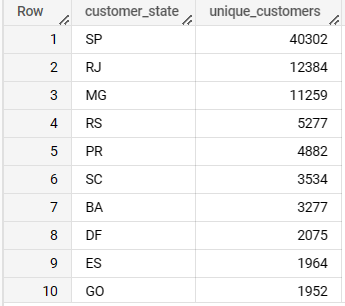
order by Customers desc;

1. select customer\_state, count(distinct customer\_unique\_id)as unique\_customers from `Target.customers`

group by customer\_state

order by unique\_customers desc;

Sample Output: a Sample Output: b

* State SP has the highest customer distribution across all the states in Brazil.
* State RR has the lowest customer distribution with only 46 customers.
* While RJ and MG states have the 2nd and 3rd highest customer base respectively.

4. Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.

* Get % increase in cost of orders from 2017 to 2018 (include months between Jan to Aug only) - You can use “payment\_value” column in payments table

with cte\_extract as (

select order\_id,extract(Year from order\_purchase\_timestamp)as purchase\_year,

extract(Month from order\_purchase\_timestamp)as purchase\_month

from `Target.orders`

),

cte\_filterByMonth as (

select purchase\_year,

round(sum(p.payment\_value),2) as total\_revenue

from cte\_extract o left join `Target.payments` p

on o.order\_id = p.order\_id

where purchase\_month>=1 and purchase\_month<=8

group by purchase\_year

order by purchase\_year desc

),

cte\_previousYearRevenue as (

select purchase\_year,total\_revenue,lag(total\_revenue) over(order by purchase\_year) as prev\_year\_revenue from cte\_filterByMonth

)

select purchase\_year,total\_revenue, ifnull(prev\_year\_revenue,0) as last\_year\_revenue,

ifnull(round(((total\_revenue - prev\_year\_revenue)/ prev\_year\_revenue \*100),2),0) as revenue\_increase\_in\_percent

from cte\_previousYearRevenue



Note: In the last\_year\_revenue column for Row 1 the value is considered zero even it was not, Since the total cost of orders was considered between 2017 to 2018

* The Revenue in 2018 has more than doubled from the previous year for the same period between Jan – Aug.
* Mean & Sum of price and freight value by customer state

select c.customer\_state, round(avg(price),2) as Mean\_price,round(Sum(price),2) as Total\_price,

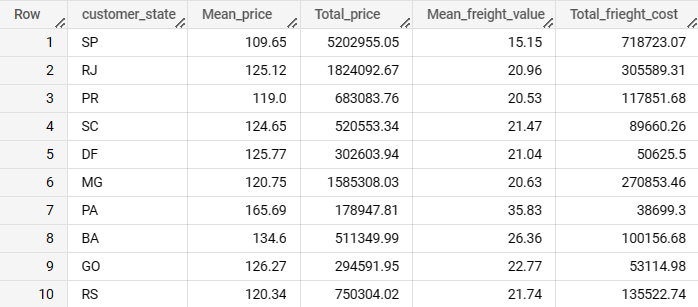
round(avg(freight\_value),2) as Mean\_freight\_value, round(sum(freight\_value),2) as Total\_frieght\_cost

from `Target.order\_items` os left join `Target.orders` o

on os.order\_id = o.order\_id left join `Target.customers` c

on o.customer\_id = c.customer\_id

group by c.customer\_state



* State PB has the highest mean price at 191.48 and SP has the lowest mean price at 109.65.
* State SP has highest total freight cost and State RR has the lowest freight cost.
* State RR has the highest mean freight value and State SP has the lowest freight value.
* State SP has the highest total sales and State RR has the lowest sales.

We can see from the data that SP with lowest mean freight value has recorded the highest freight cost and also highest sales while RR with highest mean freight value has recorded the lowest sales

5. Analysis on sales, freight and delivery time

* Calculate days between purchasing, delivering and estimated delivery

select order\_id,order\_purchase\_timestamp,order\_delivered\_customer\_date,

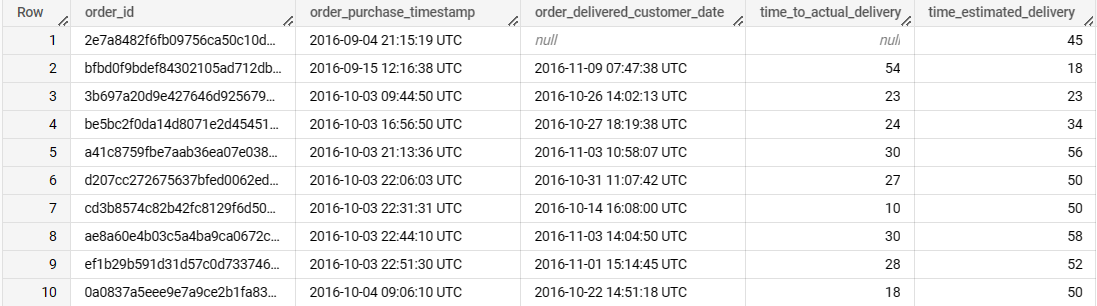
date\_diff(order\_delivered\_customer\_date, order\_purchase\_timestamp,day) as time\_to\_actual\_delivery,

date\_diff(order\_estimated\_delivery\_date, order\_purchase\_timestamp,day)as time\_estimated\_delivery

from `Target.orders`

where order\_status <> 'canceled'and order\_status <> 'unavailable'

order by order\_purchase\_timestamp



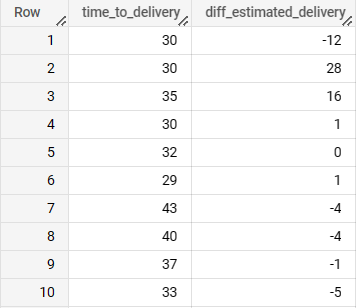
* The Time\_to\_actual\_delivery is the number of days for the order to be delivered to the customer from the date of order placed
* Time\_estimated\_delivery is the number of tentative days to deliver the product to customer from the date of order placed.
* Find time\_to\_delivery & diff\_estimated\_delivery.

select

date\_diff(order\_purchase\_timestamp,order\_delivered\_customer\_date,day) as time\_to\_delivery,

date\_diff(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,day) as diff\_estimated\_delivery

from `Target.orders`



* Here +ve values in diff estimated delivery column signifies that the product was delivered to customer well within the estimated date while -ve values signifies the delivery to customer was delayed and overtook the estimated

Ex. ‘-12’ indicates the order was delayed to customer by 12 days and 28 indicates the customer received the order 28 days before the estimated date.

* ‘30’ in time\_to\_delivery indicates the order reached the customer in 30 days from purchase date but the order was delayed by 12 days than the estimated date.
* Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

select c.customer\_state,

round(avg(freight\_value),2) as Avg\_freight\_value,

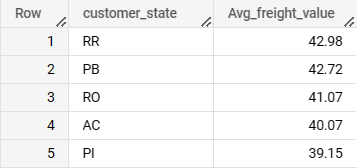
from `Target.orders` o left join `Target.order\_items` ot

on o.order\_id = ot.order\_id left join `Target.customers` c

on o.customer\_id = c.customer\_id

group by c.customer\_state

order by Avg\_freight\_value desc limit 5



* Top 5 states with highest/lowest average time to delivery

select c.customer\_state,

Round(avg(date\_diff(order\_delivered\_customer\_date,order\_purchase\_timestamp,day)),2) as Avg\_time\_to\_delivery,

from `Target.orders` o left join `Target.customers` c

on o.customer\_id = c.customer\_id

group by c.customer\_state

order by Avg\_time\_to\_delivery desc limit 5



* While the average time to delivery across all states stands at 12.09 days, it is recommended for all the states who have more than 12 days average rate of delivery to reduce which is directly linked to the sales.
* States who have less average time to delivery have reported more sales

SP,MG,PR have more sales due to good average time to delivery(i.e less than12 days).

* Customer prefer their orders to be delivered at the earliest else they will choose a different competitor for the same product even if its little pricier.
* Top 5 states where delivery is really fast/ not so fast compared to estimated date

with cte\_diff\_estimate as (select c.customer\_state,

date\_diff(order\_delivered\_customer\_date,order\_estimated\_delivery\_date,day) as diff\_estimated\_delivery,

order\_delivered\_customer\_date,order\_estimated\_delivery\_date

from `Target.orders` o left join `Target.customers` c

on o.customer\_id = c.customer\_id )

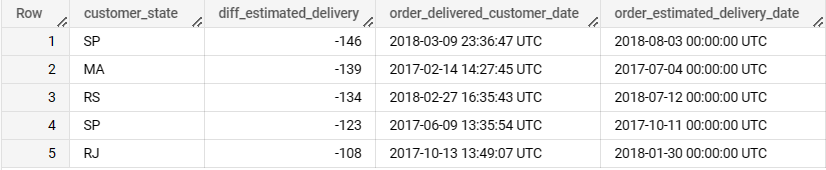
select customer\_state,diff\_estimated\_delivery,order\_delivered\_customer\_date,order\_estimated\_delivery\_date

from cte\_diff\_estimate

where diff\_estimated\_delivery is not null

group by customer\_state,diff\_estimated\_delivery,order\_delivered\_customer\_date,order\_estimated\_delivery\_date

order by diff\_estimated\_delivery limit 5



* The customer in state SP has received an order which was 146 days prior to the estimated date which is highest across all states in Brazil.

6. Payment type analysis:

* Month over Month count of orders for different payment types

select p.payment\_type,

extract(Year from order\_purchase\_timestamp)as purchase\_year,

extract(Month from order\_purchase\_timestamp)as purchase\_month,

FORMAT\_DATE("%h", DATE (order\_purchase\_timestamp))as month\_name,

count(o.order\_id)as Total\_orders

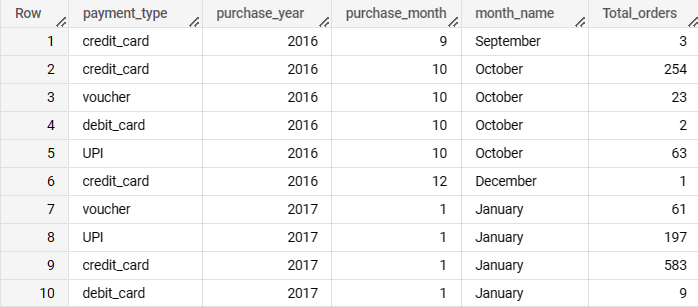
from `Target.orders` o left join `Target.payments`p

on o.order\_id = p.order\_id

where p.payment\_type is not null and p.payment\_type <> 'not\_defined'

group by p.payment\_type,purchase\_year,purchase\_month,month\_name

order by purchase\_year,purchase\_month



* Credit card is the preferred mode of payment for the majority of the orders

over the given time period from 2016-2018.

* UPI is the second preferred mode of payment over debit card.
* Count of orders based on the no. of payment installments

select payment\_installments,

count(o.order\_id)as Total\_orders

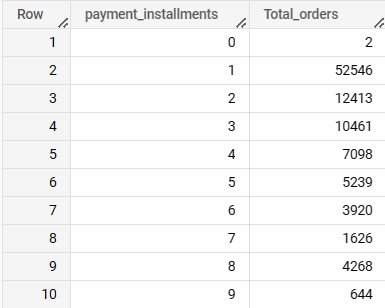
from `Target.orders` o left join `Target.payments`p

on o.order\_id = p.order\_id

where payment\_installments is not null

group by payment\_installments

order by payment\_installments



* More than 50% of the customers have opted for single installment payment for their orders.
* Around 85% of the customers have opted to complete the payment within 5 installments out of 24 that’s available.
* As the no. of installments keeps increasing the number of orders are low that’s because the customers have preferred to complete the payment for their orders as early as possible.

SUMMARY:

* There are total of 99,441 customer ids and 96096 Unique Customers ids.
* Brazil has 27 states, 4119 cities, 14994 zip codes
* There are 3095 sellers from 23 states ,611 cities ,2246 zip codes.
* There are 32951 products and 73 different categories.