Project - Target SQL dataset Analysis by Gajanan.M

Q.1] Import The Dataset And Do Usual Exploratory Analysis Steps Like Checking The Structure & Characteristics Of The Dataset
1.1] Data Type Of Columns In A Table

```
SELECT column_name, data_type
FROM project1.INFORMATION_SCHEMA.COLUMNS
```

 # Output

Row	column_name	data_type
1	order_id	STRING
2	customer_id	STRING
3	order_status	STRING
4	order_purchase_timestamp	TIMESTAMP
5	order_approved_at	TIMESTAMP
6	order_delivered_carrier_date	TIMESTAMP
7	order_delivered_customer_date	TIMESTAMP
8	order_estimated_delivery_date	TIMESTAMP

1.2] Time period for which data is given

```
select min(order_purchase_timestamp) as min_time_period,
max(order_purchase_timestamp) as max_time_period
from `project1.orders`;
```

#Output

Row	min_time_period	max_time_period
1	2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC

1.3] Cities and States of customers ordered during the given period

```
SELECT C.customer_id, C.customer_city, C.customer_state, O.order_purchase_timestamp,
FROM `gajanan147.project1.customers` AS C
JOIN `project1.orders` AS O ON C.customer_id = O.customer_id

ORDER BY order_purchase_timestamp
limit 10
```

#Output

Row	customer_id	customer_city	customer_state	order_purchase_timestamp
1	08c5351a6aca1c1589a38f244...	boa vista	RR	2016-09-04 21:15:19 UTC
2	683c54fc24d40ee9f8a6fc179f...	passo fundo	RS	2016-09-05 00:15:34 UTC
3	622e13439d6b5a0b486c4356...	sao jose dos campos	SP	2016-09-13 15:24:19 UTC
4	86dc2ffce2dfff336de2f386a78...	sao joaquim da barra	SP	2016-09-15 12:16:38 UTC
5	b106b360fe2ef8849fbbd056f7...	sao paulo	SP	2016-10-02 22:07:52 UTC
6	355077684019f7f60a031656b...	sao paulo	SP	2016-10-03 09:44:50 UTC
7	7ec40b22510fdbea1b08921dd...	panambi	RS	2016-10-03 16:56:50 UTC
8	70fc57eeae292675927697fe0...	rio de janeiro	RJ	2016-10-03 21:01:41 UTC
9	6f989332712d3222b6571b1cf...	porto alegre	RS	2016-10-03 21:13:36 UTC
10	b8cf418e97ae795672d326288...	hortolandia	SP	2016-10-03 22:06:03 UTC

Q.2] In-depth Exploration:

2.1] Is there a growing trend on e-commerce in Brazil? How can we describe a complete scenario? Can we see some seasonality with peaks at specific months?

```
SELECT X.YEAR, SUM(X.value) as value_per_year
FROM (SELECT EXTRACT(year FROM O.order_purchase_timestamp ) AS YEAR,
p.payment_value as value

FROM `project1..orders` AS O
LEFT JOIN `project1..payments` AS p ON O.order_id = p.order_id ) AS X

GROUP BY X.YEAR
ORDER BY YEAR desc;
```

#Output

Row	YEAR	value_per_year
1	2018	8699763.0499998648
2	2017	7249746.7299996857
3	2016	59362.340000000026

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

```
SELECT Y.time_of_day,
COUNT(Y.time_of_day) as no_of_purchases

FROM ( SELECT CASE
WHEN X.HOUR < 12 THEN 'MORNING'
WHEN X.HOUR < 16 THEN 'AFTERNOON'
WHEN X.HOUR < 19 THEN 'EVENING'
ELSE 'NIGHT' END AS time_of_day

FROM (SELECT EXTRACT( HOUR FROM O.order_purchase_timestamp ) AS HOUR,

FROM `project1.orders` AS O) AS X) AS Y

GROUP BY Y.time_of_day;
```

#Output

Looker Studio Reporting - 2/12/23, 3:02 PM

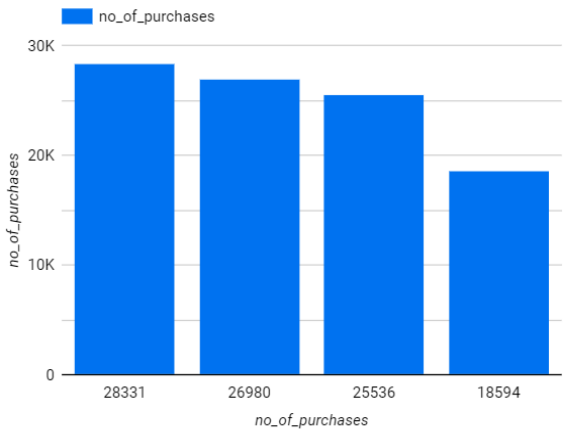
File Edit View Insert Page Arrange Resource Help

Add page | Add data | Add a chart | Add a control | Theme and layout

BigQuery Custom SQL

	time_of_day	no_of_purchases
1.	NIGHT	28,331
2.	MORNING	26,980
3.	AFTERNOON	25,536
4.	EVENING	18,594

1 - 4 / 4



Row	time_of_day	no_of_purchases
1	MORNING	26980
2	EVENING	18594
3	AFTERNOON	25536
4	NIGHT	28331

Q.3] Evolution of E-commerce orders in the Brazil region:
3.1] Get month on month orders by states

```
SELECT X.MONTH, X.region,
COUNT(X.MONTH) as orders_per_month
FROM (SELECT EXTRACT(MONTH FROM O.order_purchase_timestamp ) AS MON
TH, S.seller_state as region
FROM `project1.orders` AS O
LEFT JOIN `project1.order_items` AS OI ON OI.order_id = O.order_id
LEFT JOIN `project1.sellers` S ON OI.seller_id = S.seller_id ) AS X

GROUP BY X.MONTH, X.region
ORDER BY X.MONTH, X.region
```

Output

Row	MONTH	region	orders_per_month
1	1	*null*	60
2	1	BA	30
3	1	CE	7
4	1	DF	88
5	1	ES	25
6	1	GO	39
7	1	MA	9
8	1	MG	688
9	1	MS	7
10	1	MT	22

3.2] Distribution of customers across the states in Brazil

```
SELECT X.state,
COUNT(X.state) as orders_per_state

FROM ( SELECT S.seller_state as state FROM `project1.orders` AS O
LEFT JOIN `project1.order_items` AS OI ON OI.order_id = O.order_id
LEFT JOIN `project1.sellers` S ON OI.seller_id = S.seller_id ) AS X

GROUP BY X.state
ORDER BY X.state
```

Output

Row	state	orders_per_state
1	*null*	0
2	AC	1
3	AM	3
4	BA	643
5	CE	94
6	DF	899
7	ES	372
8	GO	520
9	MA	405
10	MG	8827

```
SELECT customer_state, COUNT(customer_id)
FROM `project1.customers`
GROUP BY customer_state
```

Output

Row	customer_state	f0_
1	RN	485
2	CE	1336
3	RS	5466
4	SC	3637
5	SP	41746
6	MG	11635
7	BA	3380
8	RJ	12852
9	GO	2020
10	MA	747

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

4.1] 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 orders_payments AS (
  SELECT o.order_id, o.order_purchase_timestamp, p.payment_value
  FROM `project1.orders` o
  JOIN `project1.payments` p ON o.order_id = p.order_id
),

order_payments_2017 AS(
  SELECT SUM(payment_value) as total_cost_2017
  FROM orders_payments

  WHERE EXTRACT(YEAR FROM order_purchase_timestamp) = 2017
  AND EXTRACT(MONTH FROM order_purchase_timestamp) >=1
  AND EXTRACT(MONTH FROM order_purchase_timestamp) <=8
),

order_payments_2018 AS(
  SELECT SUM(payment_value) as total_cost_2018
  FROM orders_payments

  WHERE EXTRACT(YEAR FROM order_purchase_timestamp) = 2018
  AND EXTRACT(MONTH FROM order_purchase_timestamp) >=1
  AND EXTRACT(MONTH FROM order_purchase_timestamp) <=8
)

SELECT (total_cost_2018
total_cost_2017) / total_cost_2017 * 100 AS percentage_increases
FROM order_payments_2017 , order_payments_2018;
```

Output

Row	percentage_increases
1	136.97687164665447

4.2] Mean & Sum of price and freight value by customer state

```
WITH value AS (
SELECT c.customer_state AS customer_state,
SUM(oi.price) as price_sum, AVG(oi.price) as price_avg,
SUM(oi.freight_value) as freight_sum, AVG(oi.freight_value) as freight_avg

FROM `project1.order_items` oi
JOIN `project1.orders` o ON oi.order_id = o.order_id
JOIN `project1.customers` c ON o.customer_id = c.customer_id

GROUP BY customer_state )

SELECT
customer_state, price_sum, price_avg, freight_sum, freight_avg
FROM value;
```

Output

Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	customer_state	price_sum	price_avg	freight_sum	freight_avg
1	SP	5202955.05...	109.653629...	718723.069...	15.1472753...
2	RJ	1824092.66...	125.117818...	305589.310...	20.9609239...
3	PR	683083.760...	119.004139...	117851.680...	20.5316515...
4	SC	520553.340...	124.653577...	89660.2600...	21.4703687...
5	DF	302603.939...	125.770548...	50625.4999...	21.0413549...
6	MG	1585308.02...	120.748574...	270853.460...	20.6301668...
7	PA	178947.809...	165.692416...	38699.3000...	35.8326851...
8	BA	511349.990...	134.601208...	100156.679...	26.3639589...
9	GO	294591.949...	126.271731...	53114.9799...	22.7668152...
10	RS	750304.020...	120.337453...	135522.740...	21.7358043...

Q.5] Analysis on sales, freight and delivery time
5.1] Calculate days between purchasing, delivering and estimated delivery

```
SELECT
TIMESTAMP_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY)
AS Days_between_purchase_delivery,

TIMESTAMP_DIFF(order_delivered_customer_date, order_estimated_delivery_date, DAY)
AS Days_between_estimated_delivery_delivery,

FROM `project1.orders`
```

Output

Row	Days_between_purchase_delivery	Days_between_estimated_delivery_delivery
1	30	12
2	30	-28
3	35	-16
4	30	-1
5	32	0
6	29	-1
7	43	4
8	40	4
9	37	1
10	33	5

5.2] Find time_to_delivery & diff_estimated_delivery. Formula for the same given below:

- time_to_delivery = order_purchase_timestamp-order_delivered_customer_date
- diff_estimated_delivery = order_estimated_delivery_date-order_delivered_customer_date

```
SELECT
TIMESTAMP_DIFF(order_delivered_customer_date, order_purchase_timestamp, HOUR)
 AS time_to_delivery,

TIMESTAMP_DIFF(order_delivered_customer_date, order_estimated_delivery_date, HOUR
) AS diff_estimated_delivery

FROM `project1.orders`
```

Output

Row	time_to_delivery	diff_estimated_delivery
1	168	-1088
2	722	310
3	743	-681
4	181	-1065
5	262	-989
6	853	-397
7	565	-228
8	311	133
9	309	-298
10	173	-24

5.3] Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery

```
SELECT AVG(X.time_to_delivery) AS mean_delivery_time,
AVG(X.diff_estimated_delivery) AS mean_diff_estimated_delivery,
AVG(X.freight_value) AS mean_freight_value,
X.customer_state

FROM ( SELECT TIMESTAMP_DIFF(O.order_delivered_customer_date,
O. order_purchase_timestamp, HOUR) AS time_to_delivery,

TIMESTAMP_DIFF(O.order_delivered_customer_date,
O.order_estimated_delivery_date, HOUR) AS diff_estimated_delivery,
OI.freight_value, C.customer_state

FROM `project1.order_items` AS OI

FULL OUTER JOIN `project1.orders` AS O ON OI.order_id = O.order_id
FULL OUTER JOIN `project1.customers` C ON C.customer_id=O.order_id ) AS X

GROUP BY X.customer_state;
```

Output

Row	mean_delivery_time	mean_diff_estimated_delivery	mean_freight_value	customer_state
1	298.84265309085...	-271.58727177029994	19.9903199289...	null
2	null	null	null	SP
3	null	null	null	RS
4	null	null	null	RJ
5	null	null	null	PB
6	null	null	null	MG
7	null	null	null	PA
8	null	null	null	BA
9	null	null	null	SC
10	null	null	null	ES

5.4] Sort the data to get the following:
Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

```
SELECT X.city,
COUNT(X.city) as orders_per_city

FROM ( SELECT S.seller_city as city FROM `project1.orders` AS O
LEFT JOIN `project1.order_items` AS OI ON OI.order_id = O.order_id
LEFT JOIN `project1.sellers` S ON OI.seller_id = S.seller_id ) AS X

GROUP BY X.city
ORDER BY COUNT(X.city)
DESC LIMIT 5
```

Output

Row	city	orders_per_city
1	sao paulo	27983
2	ibitinga	7750
3	curitiba	3016
4	santo andre	2964
5	belo horizonte	2593

Q.6] Payment type analysis:
6.1] Month over Month count of orders for different payment types

```
SELECT COUNT(O.order_id) AS no_of_orders,
EXTRACT(MONTH FROM order_purchase_timestamp) AS purchase_month,
P.  payment_type,

FROM `project1.orders` AS O
LEFT JOIN `project1.payments` AS P ON O.order_id = P.order_id

GROUP BY purchase_month, P.payment_type
ORDER BY purchase_month,P.payment_type
```

Output

Row	no_of_orders	purchase_month	payment_type
1	1715	1	UPI
2	6103	1	credit_card
3	118	1	debit_card
4	477	1	voucher
5	1723	2	UPI
6	6609	2	credit_card
7	82	2	debit_card
8	424	2	voucher
9	1942	3	UPI
10	7707	3	credit_card

Count of orders for different payment types

```
SELECT COUNT(O.order_id) no_of_orders, p.payment_type
FROM `project1.orders` AS O
LEFT JOIN `project1.payments` AS P ON O.order_id = P.order_id
GROUP BY P.payment_type
```

Output

Row	no_of_orders	payment_type
1	19784	UPI
2	76795	credit_card
3	5775	voucher
4	1529	debit_card
5	3	not_defined
6	1	null

6.2] Count of orders based on the no. of payment instalments

```
SELECT COUNT(O.order_id) no_of_orders, P.payment_installments
FROM `project1.orders` AS O
LEFT JOIN `project1.payments` AS P ON O.order_id = P.order_id
GROUP BY P.payment_installment
```

Output

Row	no_of_orders	payment_installments
1	52546	1
2	1626	7
3	5328	10
4	3920	6
5	12413	2
6	7098	4
7	10461	3
8	4268	8
9	644	9
10	5239	5

A] Number of cities in our dataset

```
SELECT COUNT ( DISTINCT customer_city) as No_Of_Cities
FROM `project1.customers` ;
```

Output

Row	No_Of_Cities
1	4119

B] Number of states in our dataset

```
SELECT
COUNT ( DISTINCT customer_state) as no_of_states
FROM `project1.customers` ;
```

Output

Row	No_Of_States
1	27

BUSINESS INSIGHTS:

1. There are 4119 Cities
2. There are 29 States
3. There is clear growth in sales in 2017 and 2018 when compared with 2016
4. There are more purchases made in NIGHT, MORNING and AFTERNOON when compared with EVENING.
5. State SP has highest orders
6. City Sao Pulao has highest sales
7. There are more customers in state SP
8. % increase in sales from 2017 to 2018 is 136.97% increase
9. Customers Tend to buy from credit card.

BUSINESS RECOMMENDATIONS:

1. Customers are spread across Brazil in various cities, so business has great reach.
2. There is steady growth in sales from 2016 to 2018 so business can think of expanding their reach and market.
3. There are less no of purchases made in evening so we should not launch any deal of the product during evening as there will be less customers.
4. There are more customers in SP state so we can plan in opening more stores in that state.
5. There are more customers located in Sao Pulao city so we can have branches in various places to increase the sales.
6. There is increase of sales of 136.97% from 2017 to 2018 so business is running in profit so we can expand the market to reach more people.
7. More customers tend to buy with credit card so we can target those customers for special offer.