Project - Target SQL dataset Analysis by Gajanan.M

Context:

Target is a globally renowned brand and a prominent retailer in the United States. Target makes itself a preferred shopping destination by offering outstanding value, inspiration, innovation and an exceptional guest experience that no other retailer can deliver.

This particular business case focuses on the operations of Target in Brazil and provides insightful information about 100,000 orders placed between 2016 and 2018. The dataset offers a comprehensive view of various dimensions including the order status, price, payment and freight performance, customer location, product attributes, and customer reviews.

By analyzing this extensive dataset, it becomes possible to gain valuable insights into Target's operations in Brazil. The information can shed light on various aspects of the business, such as order processing, pricing strategies, payment and shipping efficiency, customer demographics, product characteristics, and customer satisfaction levels.

Problem Statement :

Extract valuable insights and provide actionable recommendations from dataset..

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

Row	column_name	data_type
1	order_id	STRING
2	order_item_id	INT64
3	product_id	STRING
4	seller_id	STRING
5	shipping_limit_date	TIMESTAMP
6	price	FLOAT64
7	freight_value	FLOAT64
8	seller_id	STRING
9	seller_zip_code_prefix	INT64
10	seller_city	STRING

1. In this dataset there are 49 columns. The data helps us to check the structure & characteristics of the dataset like the Data type of all columns.
2. Also big query(schema) will provide the details of the table names

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`;

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

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;
```

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;
```
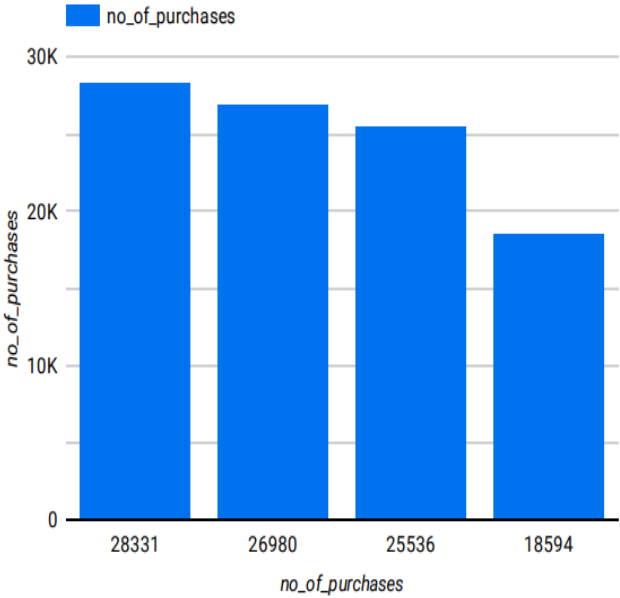
BigQuery Custom SQL

	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 ‹ ›

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 MONTH, 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

Row	MONTH	region	orders_per_month
1	5	SP	8799
2	8	SP	8713
3	7	SP	8235
4	3	SP	8027
5	4	SP	7698
6	6	SP	7547
7	2	SP	6614
8	1	SP	6451
9	11	SP	6238
10	12	SP	4546
11	10	SP	4020
12	9	SP	3454
13	2	PR	1133
14	5	MG	967

BigQuery Custom SQL

	region	orders_per_month ▾
1.	SP	80,342
2.	MG	8,827
3.	PR	8,671
4.	RJ	4,818
5.	SC	4,075
6.	RS	2,199
7.	DF	899
8.	null	775
9.	BA	643
10.	GO	520
11.	PE	448

1 - 24 / 24 ‹ ›
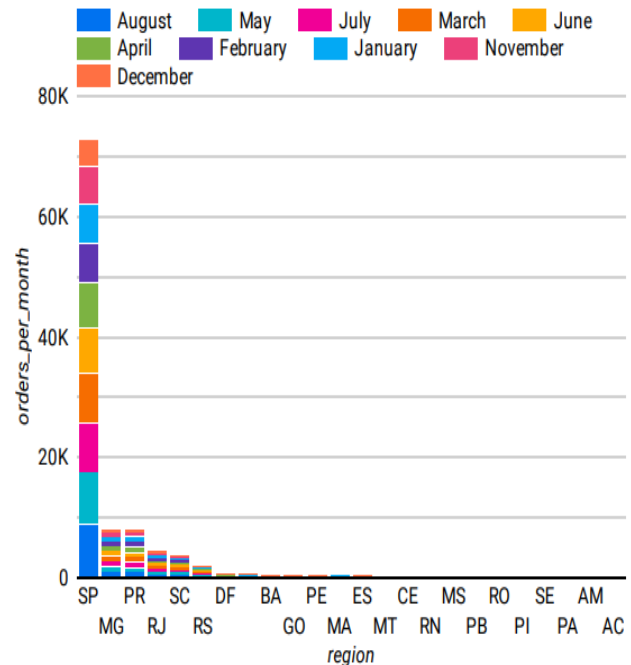
3.2] Distribution of customers across the states in Brazil

```sql
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
```
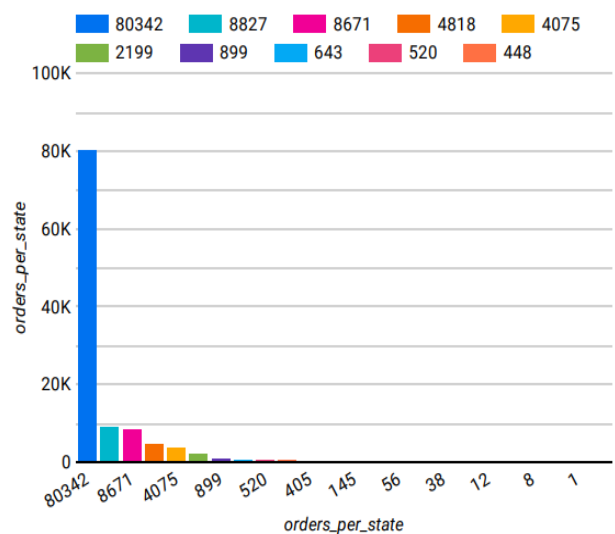
Row	state	orders_per_state
1	SP	80342
2	MG	8827
3	PR	8671
4	RJ	4818
5	SC	4075
6	RS	2199
7	DF	899
8	BA	643
9	GO	520
10	PE	448
11	MA	405
12	ES	372
13	MT	145
14	CE	94

BigQuery Custom SQL

	state	orders_per_state
1.	SP	80,342
2.	MG	8,827
3.	PR	8,671
4.	RJ	4,818
5.	SC	4,075
6.	RS	2,199
7.	DF	899
8.	BA	643
9.	GO	520
10.	PE	448
11.	MA	405

1 - 24 / 24 ‹ ›

Count total Customer State
SELECT customer_state,
COUNT(customer_id) Total

FROM `project1.customers`
GROUP BY customer_state

Row	customer_state	Total
1	SP	41746
2	RJ	12852
3	MG	11635
4	RS	5466
5	PR	5045
6	SC	3637
7	BA	3380
8	DF	2140
9	ES	2033
10	GO	2020
11	PE	1652
12	CE	1336
13	PA	975
14	MT	907

BigQuery Custom SQL

customer_state	Total ▼
1. SP	41,746
2. RJ	12,852
3. MG	11,635
4. RS	5,466
5. PR	5,045
6. SC	3,637
7. BA	3,380
8. DF	2,140
9. ES	2,033
10. GO	2,020
11. PE	1,652

1 - 27 / 27 ‹ ›
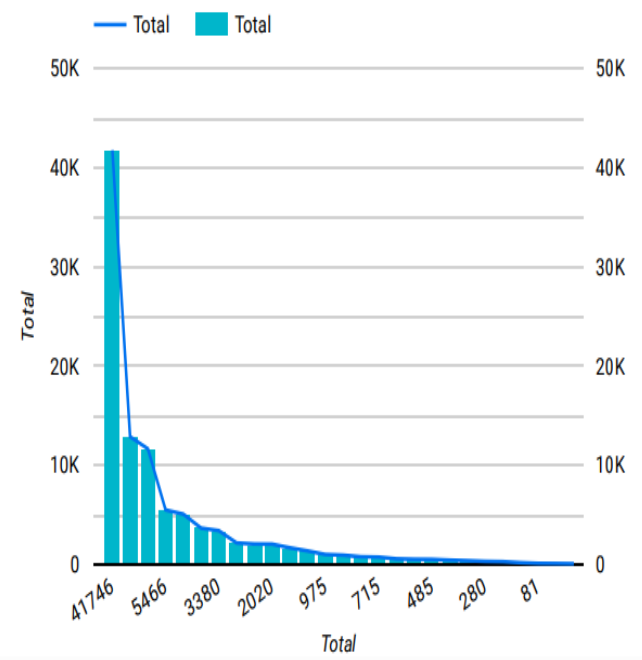
Q.4] Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.

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

```sql
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;
```

Row	percentage_increases
1	136.97687164665447

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

```sql
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;
```

Row	customer_state	price_sum	price_avg	freight_sum	freight_avg
1	RR	7829.42999…	150.565…	2235.19	42.9844230…
2	PB	115268.079…	191.475…	25719.730…	42.7238039…
3	RO	46140.6400…	165.973…	11417.379…	41.0697122…
4	AC	15982.9499…	173.727…	3686.7499…	40.0733695…
5	PI	86914.0800…	160.358…	21218.200…	39.1479704…
6	MA	119648.219…	145.204…	31523.770…	38.2570024…
7	TO	49621.7400…	157.529…	11732.680…	37.2466031…
8	SE	58920.8500…	153.041…	14111.469…	36.6531688…
9	AL	80314.81	180.889…	15914.589…	35.8436711…
10	PA	178947.809…	165.692…	38699.300…	35.8326851…

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`

Row	Days_between_purchase_delivery ▼	Days_between_estimated_delivery_delivery ▼
1	208	188
2	209	181
3	191	175
4	189	167
5	194	166
6	195	165
7	187	162
8	194	161
9	175	161
10	188	159

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

- o time_to_delivery = order_purchase_timestamp- order_delivered_customer_date
- o 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`

Row	time_to_delivery	diff_estimated_delivery
1	5000	4535
2	5031	4358
3	4595	4220
4	4556	4025
5	4671	3998
6	4695	3975
7	4505	3905
8	4205	3882
9	4657	3878
10	4515	3830

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;

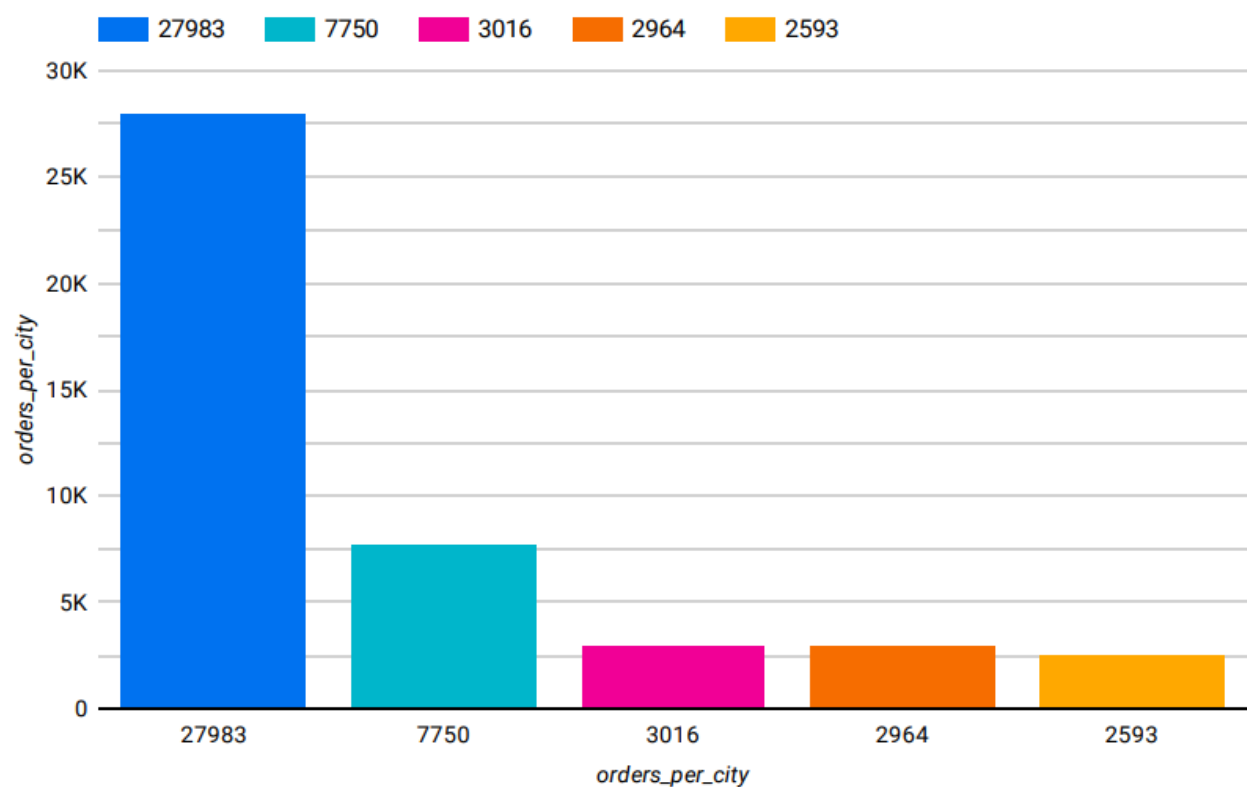| Row	mean_delivery_time	mean_diff_estimated_delivery	mean_freight_value	customer_state
1	298.84265309085623	-271.58727177030028	19.990319928983578	null
2	null	null	null	SP
3	null	null	null	ES
4	null	null	null	RJ
5	null	null	null	CE
6	null	null	null	MG
7	null	null	null	PA
8	null	null	null	BA
9	null	null	null	PE
10	null	null	null	GO

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

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

BigQuery Custom SQL

Legend: ■ 27983 ■ 7750 ■ 3016 ■ 2964 ■ 2593

y-axis: orders_per_city (0, 5K, 10K, 15K, 20K, 25K, 30K)

x-axis: orders_per_city (27983, 7750, 3016, 2964, 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

| 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

```sql
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
```

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*

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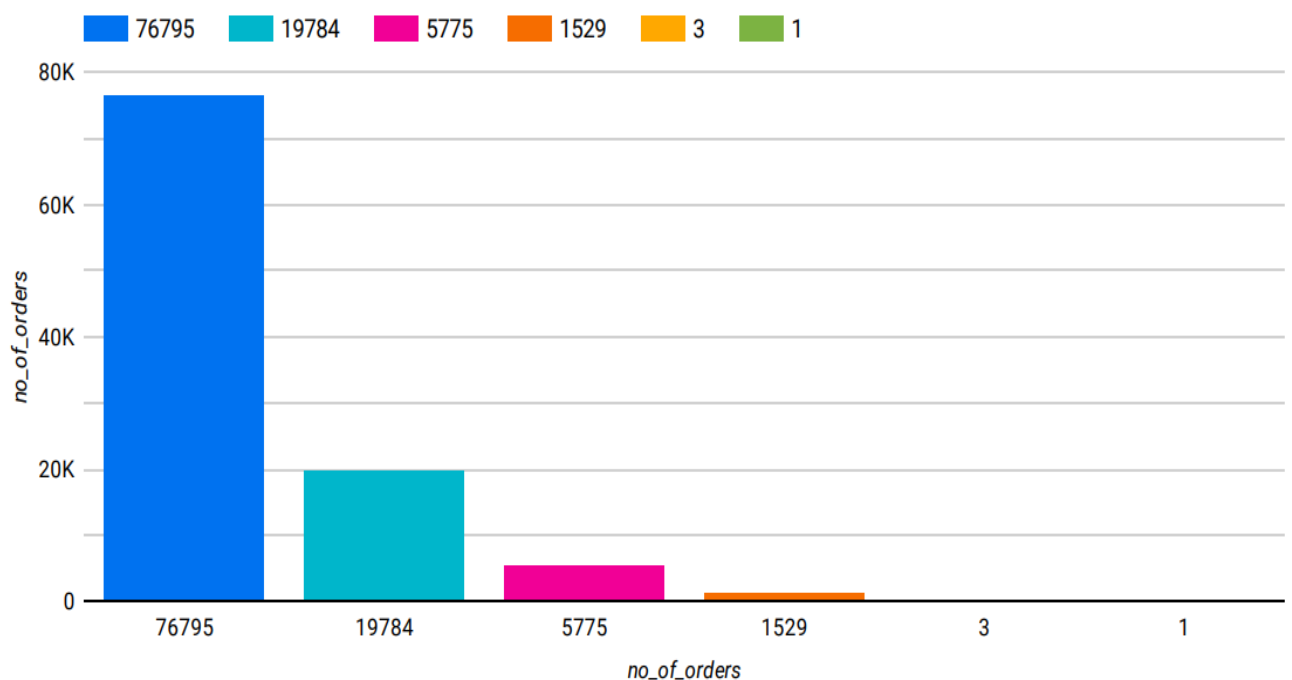
6.2] Count of orders based on the no. of payment instalments

```sql
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
```

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

A] Number of # states and # cities in our dataset

```sql
SELECT
COUNT ( DISTINCT customer_state) as No_of_states
COUNT ( DISTINCT customer_city) as No_Of_Cities

FROM `project1.customers` ;
```

Row	No_of_states	No_Of_Cities
1	27	4119

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.