Target SQL – Business Case Study

- *by Gokul Krishanth*

Table of Contents

Dataset Description

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

Assuming you are a data analyst/ scientist at Target, you have been assigned the task of analyzing the given dataset to extract valuable insights and provide actionable recommendations.

Analysis

1. Initial exploratory analysis of the structure & characteristics of the dataset:

1.1) Data type of all columns in the "customers" table.

Answer:

Query:

```
SELECT
table_schema,
table_name,
column_name,
data_type
FROM `Target.INFORMATION_SCHEMA.COLUMNS`
where table_name = 'customers';
```

Result:

Row	table_schema	table_name	column_name	data_type
1	Target	customers	customer_id	STRING
2	Target	customers	customer_unique_id	STRING
3	Target	customers	customer_zip_code_prefix	INT64
4	Target	customers	customer_city	STRING
5	Target	customers	customer_state	STRING

Insights:

- All the columns in the customers table except customer_zip_code_prefix column has **string** data type.
- customer_zip_code_prefix column contains the customer zip code, so this column is of **integer** data type.
- **customer_id** and **customer_unique_id** are unique id columns for each customer where customer_id column can be used to join with orders table to get the order details of the customers.

1.2) Get the time range between which the orders were placed.

Answer:

Query:

```
Select
min(order_purchase_timestamp) as order_purchase_start_timestamp,
max(order_purchase_timestamp) as order_purchase_end_timestamp
from `Target.orders`;
```

Result:

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

Insights:

- The orders were placed within the period from **4th September 2016** to **17th October 2018**.
- The order_purchase_timestamp column contains timestamp in the **UTC time zone**.

1.3) Count the Cities & States of customers who ordered during the given period.

Answer:

Query:

```
select
count(distinct customer_city) as city_count,
count(distinct customer_state) as state_count
from `Target.customers` c
join `Target.orders` o on c.customer_id = o.customer_id;
```

Result:

Row	city_count	state_count
1	4119	27

Insights:

- The total count of cities is 4119 and the total count of states is 27 for customers who placed orders within the specified time range.
- This result only contains the Brazilian states and cities of the customers who placed orders.

2. In-depth Exploration:

2.1) Is there a growing trend in the no. of orders placed over the past years?

Answer:

Query:

Month-wise analysis

```
select
extract(year from order_purchase_timestamp) as Year,
extract(month from order_purchase_timestamp) as Month,
count(distinct order_id) order_count
from `Target.orders`
group by Year, Month
order by Year, Month;
```

Year-wise analysis

```
select
extract(year from order_purchase_timestamp) as Year,
count(order_id) order_count
from `Target.orders`
group by Year
order by Year;
```

Result:

Month-wise analysis

Row	Year	Month	order_count
1	2016	9	4
2	2016	10	324
3	2016	12	1
4	2017	1	800
5	2017	2	1780
6	2017	3	2682
7	2017	4	2404
8	2017	5	3700
9	2017	6	3245
10	2017	7	4026

Year-wise analysis

Row	Year	order_count
1	2016	329
2	2017	45101
3	2018	54011

Insights:

- The results show that the number of orders in the first four months was unstable, and notably no orders were placed in November 2016. However, starting in January 2017 the number of orders began to rise each month.
- I also observed that in certain months, the order count peaked, likely due to factors like festive sales or year-end promotions.
- In the last two months of 2018, there was a significant drop in order count. Overall, the data indicates a general increase in orders over the years since 2016.
- This result also includes the orders which are cancelled and unavailable as we need to calculate the orders which are placed. These orders were initially placed then the status was changed to cancelled or unavailable later.

2.2) Can we see some kind of monthly seasonality in terms of the no. of orders being placed?

Answer:

Query:

```
select
extract(year from order_purchase_timestamp) as Year,
extract(month from order_purchase_timestamp) as Month,
count(distinct order_id) order_count
from `Target.orders`
```

```
group by Year, Month
order by Year, Month;
```

Result:

Row	Year	Month	order_count
11	2017	8	4331
12	2017	9	4285
13	2017	10	4631
14	2017	11	7544
15	2017	12	5673
16	2018	1	7269
17	2018	2	6728
18	2018	3	7211
19	2018	4	6939
20	2018	5	6873

Insights:

- Yes, we can clearly observe seasonality in the number of orders placed. For instance, there were significant spikes in order counts during certain months, such as November 2017 and January and March 2018.
- These increases could be attributed to events like Black Friday, Christmas, and New Year sales, among other factors.
- This indicates that order volume fluctuates with seasonal trends.

2.3) During what time of the day, do the Brazilian customers mostly place their orders?

Answer:

Query:

```
select
placement_category,
count(*) as order_count
from
(select order_id, order_purchase_timestamp,
case
when extract(hour from order_purchase_timestamp) between 0 and 6
then 'Dawn'
when extract(hour from order_purchase_timestamp) between 7 and 12
then 'Mornings'
when extract(hour from order_purchase_timestamp) between 13 and 18
then 'Afternoon'
else 'Night'
```

```
end as placement_category
from `Target.orders`)
group by placement_category
order by order_count desc;
```

Result:

Row	placement_category	order_count
1	Afternoon	38135
2	Night	28331
3	Mornings	27733
4	Dawn	5242

Insights:

- Brazilian customers primarily placed their orders in the afternoon, with a total of 38,135 orders recorded during that time.
- Orders made in the night and morning ranked second and third, respectively.
- Orders placed at dawn were significantly lower in comparison to other times of day.
- Planning on increasing the workforce, stability of Online server, load balancing etc. during Afternoon will help the company to increase their revenue and customer satisfaction.

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

3.1) Get the month-on-month no. of orders placed in each state.

Answer:

Query:

```
with mtm as(
select
extract(year from o.order_purchase_timestamp) as Year,
extract(month from o.order_purchase_timestamp) as Month,
c.customer_state,
count(distinct o.order_id) order_count
from `Target.orders` o
join `Target.customers` c on o.customer_id = c.customer_id
group by Year, Month,c.customer_state
)
select *, row_number() over(partition by Year, Month) as
state_number_mtm from mtm
order by Year, Month
```

Result:

Row	Year	Month	customer_state	order_count	state_number_mtm
1	2016	9	RR	1	1
2	2016	9	RS	1	2
3	2016	9	SP	2	3
4	2016	10	SP	113	1
5	2016	10	RS	24	2
6	2016	10	RJ	56	3
7	2016	10	MT	3	4
8	2016	10	GO	9	5
9	2016	10	MG	40	6
10	2016	10	CE	8	7
11	2016	10	SC	11	8
12	2016	10	AL	2	9
13	2016	10	BA	4	10
14	2016	10	PE	7	11

Insights:

- The month-on-month results indicate fluctuations in order volume.
- At the start of 2016, the order count was low, but it increased over time, with noticeable peaks occurring in some states during 2017.
- We can also observe that initially the number of states were less, but more states came in to picture over time.
- state_number_mtm column contains the serial number for states in each month.
- With this result, we could observe that in many states the number of orders placed are very less. Proper planning in increasing the productivity and marketing strategies might improve the business.

3.2) How are the customers distributed across all the states?

Answer:

Query:

```
Select
customer_state,
count(distinct customer_unique_id) as customer_count
from `Target.customers`
group by customer_state
order by customer_count desc;
```

Result:

Row	customer_state	customer_count
1	SP	40302
2	RJ	12384
3	MG	11259
4	RS	5277
5	PR	4882
6	SC	3534
7	BA	3277
8	DF	2075
9	ES	1964
10	GO	1952
11	PE	1609
12	CE	1313
13	PA	949
14	MT	876

Insights:

- From the result we can see that state SP contains a greater number of customers (40302) and RR (45) with lowest customer count.
- Increasing marketing and promotions in the states with lower customer count might attract the customers.

4. Impact on Economy: Analyze the money movement by e-commerce.

4.1) Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only).

Answer:

Query:

Year wise

```
select
concat(cost2017,' BRL') as Total_cost_2017,
concat(cost2018,' BRL') Total_cost_2018,
concat(round(cost2018-cost2017,2),' BRL') as cost_difference,
concat(round(((cost2018-cost2017)/cost2017)*100,2),'%') as
difference_percentage
from
(select sum(case when Year = 2017 then cost else 0 end) cost2017,
sum(case when Year = 2018 then cost else 0 end) cost2018
from
(select extract(year from o.order_purchase_timestamp) Year,
extract(month from o.order_purchase_timestamp) Month,
```

```
round(sum(p.payment_value),2) as cost
from `Target.orders` o
join `Target.payments` p on o.order_id = p.order_id
Group by Year, Month)b
where Year <> 2016 and Month < 9
)a;
```

Month wise

```
Select
month_1 as Month,
concat(cost2017,' BRL') as Total_cost_2017,
concat(cost2018,' BRL') Total_cost_2018,
concat(round(cost2018-cost2017,2),' BRL') as cost_difference,
concat(round(((cost2018-cost2017)/cost2017)*100,2),'%') as
difference_percentage
from
(select month_1, Month, sum(case when Year = 2017 then cost else 0
end) cost2017,sum(case when Year = 2018 then cost else 0 end)
cost2018
from
(select extract(year from o.order_purchase_timestamp) Year,
extract(month from o.order_purchase_timestamp) Month,
format_datetime('%b', o.order_purchase_timestamp) month_1,
round(sum(p.payment_value),2) as cost
from `Target.orders` o
join `Target.payments` p on o.order_id = p.order_id
Group by Year, Month, month_1)b
where Year <> 2016 and Month < 9
group by Month, month_1
order by Month)a;
```

Result:

Year wise

Row	Total_cost_2017	Total_cost_2018	cost_difference	difference_percentage
1	3669022.12 BRL	8694733.84 BRL	5025711.72 BRL	136.98%

Month wise

Row	Month	Total_cost_2017	Total_cost_2018	cost_difference	difference_percentage
1	Jan	138488.04 BRL	1115004.18 BRL	976516.14 BRL	705.13%
2	Feb	291908.01 BRL	992463.34 BRL	700555.33 BRL	239.99%
3	Mar	449863.6 BRL	1159652.12 BRL	709788.52 BRL	157.78%
4	Apr	417788.03 BRL	1160785.48 BRL	742997.45 BRL	177.84%
5	May	592918.82 BRL	1153982.15 BRL	561063.33 BRL	94.63%
6	Jun	511276.38 BRL	1023880.5 BRL	512604.12 BRL	100.26%
7	Jul	592382.92 BRL	1066540.75 BRL	474157.83 BRL	80.04%
8	Aug	674396.32 BRL	1022425.32 BRL	348029 BRL	51.61%

Insights:

- The yearly total of order prices reveals a 136.98% increase from January to August 2018 compared to January to August 2017.
- The month-by-month analysis highlights the percentage difference in total order costs between 2017 and 2018. Notably, January shows a cost difference of 705.13%, marking the highest variation among all months.
- Month wise analysis shows more granular level of cost of orders for each month and helps us understand where the company must take actions to improve their business.

4.2) Calculate the Total & Average value of order price for each state.

Answer:

Query:

```
select
c.customer_state,
concat(round(sum(p.price),2), ' BRL') as total_cost,
concat(round(avg(p.price),2), ' BRL') as avg_cost
from `Target.customers` c
join `Target.orders` o on c.customer_id = o.customer_id
join `Target.order_items` p on o.order_id = p.order_id
group by c.customer_state
order by c.customer_state;
```

Result:

Row	customer_state	total_cost	avg_cost
1	AC	15982.95 BRL	173.73 BRL
2	AL	80314.81 BRL	180.89 BRL
3	AM	22356.84 BRL	135.5 BRL
4	AP	13474.3 BRL	164.32 BRL
5	BA	511349.99 BRL	134.6 BRL
6	CE	227254.71 BRL	153.76 BRL
7	DF	302603.94 BRL	125.77 BRL
8	ES	275037.31 BRL	121.91 BRL
9	GO	294591.95 BRL	126.27 BRL
10	MA	119648.22 BRL	145.2 BRL
11	MG	1585308.03 BRL	120.75 BRL
12	MS	116812.64 BRL	142.63 BRL

Insights:

- From the above result we can observe the total order price and average order price for each state.
- The BRL in total cost and the average cost denotes Brazilian Real.

4.3) Calculate the Total & Average value of order freight for each state.

Answer:

Query:

```
select
c.customer_state,
concat(round(sum(distinct p.freight_value),2), ' BRL') as total_cost,
concat(round(avg(p.freight_value),2), ' BRL') as avg_cost
from `Target.customers` c
join `Target.orders` o on c.customer_id = o.customer_id
join `Target.order_items` p on o.order_id = p.order_id
group by c.customer_state
order by c.customer_state;
```

Result:

Row	customer_state	total_cost	avg_cost
1	AC	3078.18 BRL	40.07 BRL
2	AL	12031.87 BRL	35.84 BRL
3	AM	4065.5 BRL	33.21 BRL
4	AP	2282.73 BRL	34.01 BRL
5	BA	47819.59 BRL	26.36 BRL
6	CE	30658.4 BRL	32.71 BRL
7	DF	25772.03 BRL	21.04 BRL
8	ES	25575.68 BRL	22.06 BRL
9	GO	27620.61 BRL	22.77 BRL
10	MA	20819.56 BRL	38.26 BRL
11	MG	77356.18 BRL	20.63 BRL
12	MS	12485.4 BRL	23.37 BRL

Insights:

- From the above result we can observe the total order freight value and average order freight value for each state.
- The BRL in total cost and the average cost denotes Brazilian Real.
- By analyzing this result, we will be able to see the logistic cost for each state. RR has the highest average freight cost whereas SP has the lowest.
- We can notice high total freight cost but lower average cost in some states.

5. Analysis based on sales, freight, and delivery time.

5.1) Find the no. of days taken to deliver each order from the order's purchase date as delivery time. Also, calculate the difference (in days) between the estimated & actual delivery date of an order.

Answer:

Query:

```
select
order_id,
order_delivered_customer_date,
order_purchase_timestamp,
datetime_diff(order_delivered_customer_date,order_purchase_timestamp,DAY) as days_taken_to_deliver,
datetime_diff(order_estimated_delivery_date,order_delivered_customer_date,DAY) as delivery_days_estimated_vs_actual
from `Target.orders`;
```

Result:

Row	order_id	order_delivered_customer_date	order_purchase_timestamp	days_taken_to_delive	delivery_days_estim
1	1950d777989f6a877539f5379...	2018-03-21 22:03:51 UTC	2018-02-19 19:48:52 UTC	30	-12
2	2c45c33d2f9cb8ff8b1c86cc28...	2016-11-09 14:53:50 UTC	2016-10-09 15:39:56 UTC	30	28
3	65d1e226dfaeb8cdc42f66542...	2016-11-08 10:58:34 UTC	2016-10-03 21:01:41 UTC	35	16
4	635c894d068ac37e6e03dc54e...	2017-05-16 14:49:55 UTC	2017-04-15 15:37:38 UTC	30	1
5	3b97562c3aee8bdedcb5c2e45...	2017-05-17 10:52:15 UTC	2017-04-14 22:21:54 UTC	32	0
6	68f47f50f04c4cb6774570cfde...	2017-05-16 09:07:47 UTC	2017-04-16 14:56:13 UTC	29	1
7	276e9ec344d3bf029ff83a161c...	2017-05-22 14:11:31 UTC	2017-04-08 21:20:24 UTC	43	-4
8	54e1a3c2b97fb0809da548a59...	2017-05-22 16:18:42 UTC	2017-04-11 19:49:45 UTC	40	-4
9	fd04fa4105ee8045f6a0139ca5...	2017-05-19 13:44:52 UTC	2017-04-12 12:17:08 UTC	37	-1
10	302bb8109d097a9fc6e9cefc5...	2017-05-23 14:19:48 UTC	2017-04-19 22:52:59 UTC	33	-5
11	66057d37308e787052a32828...	2017-05-24 08:11:57 UTC	2017-04-15 19:22:06 UTC	38	-6
12	19135c945c554eebfd7576c73...	2017-08-16 20:19:32 UTC	2017-07-11 14:09:37 UTC	36	-2

Insights:

- From the above result we can analyze the number of days taken to deliver the order from the date of order placement and the difference in number of days taken to deliver the order from the estimated delivery date.
- Analyzing this helps us to notice the orders which are delivered faster and we could analyze why the orders were delivered faster so that the same strategy can be implemented in all the other orders.
- With respect to orders that are delivered late, we can look more in to the factors that are causing this and improvise it to deliver product on time to gain customer satisfaction.

5.2) Find out the top 5 states with the highest & lowest average freight value.

Answer:

Query:

```
select customer_state, category, freight_avg, rn as
category_order_number
from
(select customer_state, 'High Average' as category, freight_avg,
dense_rank() over(order by freight_avg desc) as rn
from
(select c.customer_state, round(avg(i.freight_value),2) as freight_avg
from `Target.customers` c
join `Target.orders` o on c.customer_id = o.customer_id
join `Target.order_items` i on o.order_id = i.order_id
group by c.customer_state)a
union all
select customer_state,'Low Average' as category, freight_avg,
dense_rank() over(order by freight_avg) as rn
from
```

```
(select c.customer_state, round(avg(i.freight_value),2) as freight_avg
from `Target.customers` c
join `Target.orders` o on c.customer_id = o.customer_id
join `Target.order_items` i on o.order_id = i.order_id
group by c.customer_state)a)b
where rn <= 5
order by category, rn;
```

Result:

Row	customer_state	category	freight_avg	category_order_number
1	RR	High Average	42.98	1
2	PB	High Average	42.72	2
3	RO	High Average	41.07	3
4	AC	High Average	40.07	4
5	PI	High Average	39.15	5
6	SP	Low Average	15.15	1
7	PR	Low Average	20.53	2
8	MG	Low Average	20.63	3
9	RJ	Low Average	20.96	4
10	DF	Low Average	21.04	5

Insights:

- From the above result we can see the top 5 states with Highest and lowest average freight value.
- The Category column shows us whether the state has high or low average.
- Some states have high freight average which needs to be optimized in a way to reduce the cost to improve the business.

5.3) Find out the top 5 states with the highest & lowest average delivery time.

Answer:

Query:

```
select customer_state, category, delivery_avg_days, category_order
from
(select customer_state, 'HIGH Average' category, delivery_avg_days, rn
as category_order
from
(select customer_state, round(avg(delivery_time),2) delivery_avg_days,
dense_rank() over(order by avg(delivery_time) desc) as rn
from
(select c.customer_state,
datetime_diff(o.order_delivered_customer_date,o.order_purchase_times
tamp,DAY) as delivery_time
from `Target.customers` c
join `Target.orders` o on c.customer_id = o.customer_id
```

```
)a
group by customer_state)b
union all
select customer_state, 'LOW Average' category, delivery_avg_days, rn as
category_order
from
(select customer_state, round(avg(delivery_time),2) delivery_avg_days,
dense_rank() over(order by avg(delivery_time)) as rn
from
(select c.customer_state,
datetime_diff(o.order_delivered_customer_date,o.order_purchase_times
tamp,DAY) as delivery_time
from `Target.customers` c
join `Target.orders` o on c.customer_id = o.customer_id
)a
group by customer_state)b)c
where category_order <= 5
order by category, category_order;
```

Result:

Row	customer_state	category	delivery_avg_days	category_order
1	RR	HIGH Average	28.98	1
2	AP	HIGH Average	26.73	2
3	AM	HIGH Average	25.99	3
4	AL	HIGH Average	24.04	4
5	PA	HIGH Average	23.32	5
6	SP	LOW Average	8.3	1
7	PR	LOW Average	11.53	2
8	MG	LOW Average	11.54	3
9	DF	LOW Average	12.51	4
10	SC	LOW Average	14.48	5

Insights:

- From the above result we can see the top 5 states with Highest and lowest average delivery time.
- The Category column shows us whether the state has high or low average.
- Average delivery days are high in states like RR, AP. This might affect the business and customer satisfaction.
- To improve this, analysis must be done on the states where the delivery average days are less. This will help us to implement the same strategy in the higher average states.

5.4) Find out the top 5 states where the order delivery is fast as compared to the estimated date of delivery.

Answer:

Query:

```
select * from
(select customer_state, round(delivery_average,2)delivery_average,
dense_rank() over(order by delivery_average desc) as
fast_delivery_number
from
(select
c.customer_state,
avg(datetime_diff(o.order_estimated_delivery_date,o.order_delivered_cu
stomer_date,day))delivery_average
from `Target.customers` as c
join `Target.orders` as o on c.customer_id = o.customer_id
group by c.customer_state)a)b
where fast_delivery_number <= 5
order by fast_delivery_number;
```

Result:

Row	customer_state	delivery_average	fast_delivery_numbe
1	AC	19.76	1
2	RO	19.13	2
3	AP	18.73	3
4	AM	18.61	4
5	RR	16.41	5

Insights:

- From the above result we can see the top 5 states with the fastest order delivery.
- This shows how fast the orders are delivered when compared to the estimated delivery time.
- This will improve the business to achieve higher order counts every month and it will tract other platform customer to purchase from Target as the delivery is very fast.

6. Analysis based on the payments:

6.1) Find the month-on-month no. of orders placed using different payment types.

Answer:

Query:

```
select a.order_time, payment_type, sum(order_count) order_count
from
(select format_datetime('%Y-%m' ,o.order_purchase_timestamp)
order_time,p.payment_type, count(o.order_id) as order_count
from `Target.orders` o
join `Target.payments` p on o.order_id = p.order_id
where o.order_status not in ('canceled','unavailable')
group by o.order_purchase_timestamp,p.payment_type)a
group by order_time, payment_type
order by order_time;
```

Result:

Row	order_time	payment_type	order_count
1	2016-09	credit_card	1
2	2016-10	credit_card	227
3	2016-10	UPI	60
4	2016-10	voucher	22
5	2016-10	debit_card	2
6	2016-12	credit_card	1
7	2017-01	credit_card	574
8	2017-01	UPI	193
9	2017-01	voucher	60
10	2017-01	debit_card	9
11	2017-02	credit_card	1309
12	2017-02	UPI	383

Insights:

- Above result shows the number of orders placed each month using different payment methods.
- Most of the customers in different states used credit card payment.

6.2) Find the no. of orders placed based on the payment installments that have been paid.

Answer:

Query:

Orders including zero installments

```
select payment_installments, count(distinct order_id) as order_count
from `Target.payments`
group by payment_installments
order by payment_installments;
```

Result:

Row	payment_installments	order_count
1	0	2
2	1	49060
3	2	12389
4	3	10443
5	4	7088
6	5	5234
7	6	3916
8	7	1623
9	8	4253
10	9	644
11	10	5315
12	11	23
13	12	133

Orders with installments

```
select payment_installments, count(distinct order_id) as order_count
from `Target.payments`
where payment_installments != 0
group by payment_installments
order by payment_installments
```

Result:

Row	payment_installment	order_count
1	1	49060
2	2	12389
3	3	10443
4	4	7088
5	5	5234
6	6	3916
7	7	1623
8	8	4253
9	9	644
10	10	5315
11	11	23
12	12	133
13	13	16

Insights:

- Above result shows the number of orders that are placed under installments.
- Installment method is very helpful for the business as it gives the customer to purchase whatever product they want now and pay later or every month as per their convenience.
- Business should enable as many banks as possible to provide installment or EMI options.
- This will increase the business growth significantly.

7. Actionable Insights & Recommendations:

Insights:

From the overall analysis of the Target dataset, I can observe:

- Customer orders started off quite low but saw a significant increase in 2017, only to drop sharply in 2018.
- The volume of orders varies by state, likely due to factors such as insufficient marketing and promotions, competition from established retail stores, pricing issues, limited product variety, store location, negative customer feedback, and inconsistent delivery services.

```
select
o.order_id, r.review_score, r.review_comment_title
from `Target.orders` o
join `Target.order_reviews` r on o.order_id = r.order_id
where review_comment_title is not null;
```

Row	order_id	review_score	review_comment_title
1	e34d7ca05296d868e693409be...	1	The product came wrong
2	65e004516d71c3c2e5e08910a...	1	Wrong product
3	61c724e40941762318a52917...	1	I hated it sa's at loss
4	574f906eedf477adaf35072228...	1	Lack of respect
5	430bd310105f069ac72ef7758f...	1	The cÃ
6	50e46ab38a7ca83546cab50d4...	1	NON-RECEIPT
7	474f2b1387a07ea0159ca7a99...	1	I didn't receive my order
8	72fb560d115ecf3b15de9b853...	1	I received the product
9	d1aa2409cda0cbe698a489a0f...	1	Delay
10	37cd892c7a6275aecbe6dfdce...	1	Wrong product
11	d34b43bca6cc1cb6d49c06d60...	1	I don't want to evaluate
12	0a79d1506c2e0249c9aff8723...	1	Delivery delay
13	058e0a8627dc25b49eae4ce6c...	1	delivery delay

- I can observe that there are more than 2000 orders with review score less than 3. This was caused due to various reasons including sending wrong product, delayed delivery, product not received etc.
- There are many orders that are in unavailable and cancelled status. In-depth analysis on the reason for unavailability and cancellation will provide good strategies to mitigate these occurrences.
- Other orders under different categories are as below:

```
select
order_status, count(distinct order_id) order_count
from `Target.orders`
group by 1
order by order_count desc
```

Row	order_status	order_count
1	delivered	96478
2	shipped	1107
3	canceled	625
4	unavailable	609
5	invoiced	314
6	processing	301
7	created	5
8	approved	2

- From the above result, we can see that 96478 orders are already delivered, 1107 orders are shipped currently, 314 orders are invoiced, 301 orders are in processing state, 5 in created and 2 in approved state.
- Many orders were delivered later than the estimated delivery date, which could negatively affect both the business and customer satisfaction.

```
select
order_id,
order_estimated_delivery_date,
datetime_diff(order_delivered_customer_date,order_purchase_ti
mestamp,DAY) as days_taken_to_deliver,
datetime_diff(order_delivered_customer_date,
order_estimated_delivery_date, DAY) as
delivery_days_estimated_vs_actual
from `Target.orders`
```

```
where order_status not in('cancelled', 'unavailable') and
order_delivered_customer_date is not null
order by delivery_days_estimated_vs_actual desc;
```

Row	order_id	order_estimated_delivery_date	days_taken_to_delive	delivery_days_estima
1	1b3190b2dfa9d789e1f14c05b...	2018-03-15 00:00:00 UTC	208	188
2	ca07593549f1816d26a572e06...	2017-03-22 00:00:00 UTC	209	181
3	47b40429ed8cce3aee9199792...	2018-01-19 00:00:00 UTC	191	175
4	2fe324febf907e3ea3f2aa9650...	2017-04-05 00:00:00 UTC	189	167
5	285ab9426d6982034523a855f...	2017-04-06 00:00:00 UTC	194	166
6	440d0d17af552815d15a9e41a...	2017-04-07 00:00:00 UTC	195	165
7	c27815f7e3dd0b926b5855262...	2017-04-10 00:00:00 UTC	187	162
8	0f4519c5f1c541ddec9f21b3bd...	2017-04-11 00:00:00 UTC	194	161
9	d24e8541128cea179a11a6517...	2017-06-26 00:00:00 UTC	175	161
10	2d7561026d542c8dbd8f0daea...	2017-04-13 00:00:00 UTC	188	159
11	6e82dcfb5eada6283dba34f16...	2017-06-14 00:00:00 UTC	182	155
12	2fb597c2f772eca01b1f5c561b...	2017-04-17 00:00:00 UTC	194	155

- The results indicate instances of significantly delayed deliveries, with some orders arriving over 150 days late. This greatly affects customer trust and satisfaction.
- Most of the customers used credit card to purchase the products in most of the states.
- Important thing that the business offered is the installments, which enables customers to buy more products now and pay the amount in installments.

```
select
case
when `product category` is null then 'Not Defined'
else `product category`
end as product_category,
count(distinct product_id) as product_count
from `Target.products`
group by `product category`
order by `product category`;
```

Row	product_category	product_count
1	Not Defined	610
2	Agro Industria e Comercio	74
3	Art	55
4	Arts and Crafts	19
5	Bags Accessories	349
6	Blu Ray DVDs	48
7	CITTE AND UPHACK FURNITURE	10
8	CONSTRUCTION SECURITY TO...	91
9	Casa Construcao	225
10	Christmas articles	65
11	Construction Tools Construction	400
12	Construction Tools Garden	88

- From the above result we can see the number of products under each category. We have 3029 products under 'bed table bath' category which is the highest of all other categories. 'cds music dvds' category contains the lowest number of products.

```
select
case
when `product category` is null then 'Not Defined'
else `product category`
end as product_category,
count(distinct o.order_id) as order_count
from `Target.products` p
join `Target.order_items` i on i.product_id = p.product_id
join `Target.orders` o on o.order_id = i.order_id
group by `product category`
order by order_count desc;
```

Row	product_category	order_count
1	bed table bath	9417
2	HEALTH BEAUTY	8836
3	sport leisure	7720
4	computer accessories	6689
5	Furniture Decoration	6449
6	housewares	5884
7	Watches present	5624
8	telephony	4199
9	automotive	3897
10	toys	3886
11	Cool Stuff	3632
12	Garden tools	3518

- We can see the distinct count of orders placed under each category, where 'bed table bath' bath category has the highest

order count, this is because this category has a greater number of products. Similarly other categories also exhibit similar characteristics.

Recommendations:

- In today's world data is everything. This business can be improved well by analyzing the sales data, understanding the demand, customer preferences and offering wide variety of products.
- Proper staffing and staff training is important. From the data we could see there are many orders delivered late to the customers. By improving faster delivery, we can get good feedback from the users which in turn attracts other customers.
- Regularly getting customer feedback will help the business to understand the places where more improvement is needed.
- Offer more attractive deals than the competitors on products occasionally and importantly during festive seasons like New Year, Christmas, Thanks Giving, Black Friday etc. to attract customers.
- We can also see in many places the order count is very less. In this case business should analyze why there is a low order count.
- Collaboration with different banks will drastically increase the customer purchases using credit and debit cards, which will also offer the customers installment options.
- Partnership with the vendors/sellers will also play a key role in the development. Finding the best products with proper price from the vendor is important to increase customer satisfaction and business growth.
- By analyzing the peak hours business can utilize proper and right amount of workforce to help the business achieve higher target every month or week.
- Business can slowly introduce new stores in the cities having a greater number of orders to reduce shipping charges, logistic charges, and other delivery charges.