Business Case: Target

By Arpita Hoolageri

*Disclaimer:

This analysis is based on the data provided and reflects the state of the dataset as of the time of the analysis. The insights and recommendations are derived solely from my point of view and the dataset in question and do not necessarily represent the broader operations or circumstances of the company. The analysis assumes the accuracy of the data as received and has not been independently verified. Future analyses may yield different insights as new data becomes available or as business conditions change.

*Note on Results:

Due to the large volume of results generated by this analysis, only a subset has been presented here to illustrate the key trends and patterns. Specifically, the top 10 to 15 results have been included as screenshots to provide a snapshot of the most relevant findings. For a complete view of the data and to explore additional insights, please refer to the full dataset.

What does 'good' look like?

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

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

Query:

```
1  select column_name,data_type
2  from feb24-417916.target.INFORMATION_SCHEMA.COLUMNS
3  WHERE TABLE_NAME="customers";
```

Result:

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EX

Row	column_name	data_type
1	customer_id	STRING
2	customer_unique_id	STRING
3	customer_zip_code_prefix	INT64
4	customer_city	STRING
5	customer_state	STRING

Insight: Understanding the data types of each column in the tables is essential for data manipulation and analysis. It allows us to know the nature of the data stored in each column, facilitating appropriate querying and processing.

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

Query:

```
select
min(order_purchase_timestamp),
max(order_purchase_timestamp)
from
target.orders ;
```

Result:

Row	f0_	f1_
1	2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC

Insight: Knowing the time range during which orders were placed provides us with valuable information about the duration covered by the dataset. This insight is crucial for understanding the historical scale of our analysis and can help identify any trends or patterns over time

The orders were placed between September 4th 2016 and 17th October 2018

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

Query:

```sql
select
count(distinct customer_city) as no_of_cities,
count(distinct customer_state) as no_of_states
from
target.customers;
```

Result:

Row	no_of_cities	no_of_states
1	4119	27

Insight: By counting the number of unique cities and states from which orders were placed, we gain insight into the geographic distribution of customers. This information can be useful for identifying regions with high customer engagement and potential areas for targeted marketing or expansion efforts.

Customers have ordered from 27 states and 4119 cities.

2.In-depth Exploration:

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

Query:

```sql
select
count(order_id) as num_of_orders,
extract(year from order_purchase_timestamp) as year
from
target.orders
group by year
order by year;
```

Result:

Row	num_of_orders	year
1	329	2016
2	45101	2017
3	54011	2018

Insight: Analysing the trend in the number of orders placed over the past years is essential for understanding the business's growth trajectory. By examining the order volume month by month or year by year, we can identify whether there is a consistent upward trend indicating growth or not. This insight is crucial for strategic planning, resource allocation, and forecasting future business performance.

The available data for years 2016 and 2018 is limited(we don't have the all the 12months data). With the available data we can say number of orders have increased from 2017 to 2018

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

Query 1:

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

Result 1:

Row	num_of_orders	Month	Year
1	800	1	2017
2	7269	1	2018
3	1780	2	2017
4	6728	2	2018
5	2682	3	2017
6	7211	3	2018

Insight 1: For the given limited data, In the months between May and Sept, there is a spike in orders.

```
select
count(order_id) as num_of_orders,
extract(month from order_purchase_timestamp) as Month
from
target.orders
group by Month
order by Month;
```

Row	num_of_orders	Month
1	8069	1
2	8508	2
3	9893	3
4	9343	4
5	10573	5
6	9412	6
7	10318	7
8	10843	8
9	4305	9
10	4959	10

2.3 During what time of the day, do the Brazilian customers mostly place their orders? (Dawn, Morning, Afternoon or Night)

- 0-6 hrs : Dawn
- 7-12 hrs : Mornings
- 13-18 hrs : Afternoon
- 19-23 hrs : Night

Query:

```
select
count(order_id) as `number_of_orders`,
(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 `hour`
FROM
target.orders
GROUP BY
(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)
order by `number_of_orders`
```

Result:

Row	number_of_orders	hour
1	5242	Dawn
2	27733	Mornings
3	28331	Night
4	38135	Afternoon

Insight: Brazilian customers place their orders maximum during Afternoon.

Recommendations:

Targeted Promotions: Schedule flash sales or special promotions during the afternoon and night hours to capitalize on the higher traffic. This could include limited-time offers to encourage impulse purchases.

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

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

Query 1:

```
select
state,
month,
year,
sum(num_of_orders) as total_num_of_orders
from  (select
customer_state as state,
extract(month from order_delivered_customer_date) as Month,
extract(year from order_delivered_customer_date) as year,
count(order_id) over(partition by extract(month from order_delivered_customer_date) order by c.customer_id ) as num_of_orders
from
target.orders as o join target.customers c
on o.customer_id=c.customer_id
where extract(year from order_delivered_customer_date)=2016 or
extract(year from order_delivered_customer_date)=2017 or
extract(year from order_delivered_customer_date)=2018) as T
group by
month,year,T.state
order by
month;
```

Query2:

```
select
state,
month,
sum(num_of_orders) as total_num_of_orders
from  (select
customer_state as state,
extract(month from order_delivered_customer_date) as Month,
extract(year from order_delivered_customer_date) as year,
count(order_id) over(partition by extract(month from order_delivered_customer_date) order by c.customer_id ) as num_of_orders
from
target.orders as o join target.customers c
on o.customer_id=c.customer_id
where extract(year from order_delivered_customer_date)=2016 or
extract(year from order_delivered_customer_date)=2017 or
extract(year from order_delivered_customer_date)=2018) as T
group by
month,state
order by
month;
```

Result1:

Row	state	month	year	total_num_of_orders
1	MG	1	2018	2649096
2	SP	1	2018	8992710
3	SC	1	2018	808644
4	RS	1	2018	1210218
5	RJ	1	2018	3035295
6	ES	1	2018	470720
7	PE	1	2018	402038
8	MA	1	2018	229182
9	MG	1	2017	135013
10	BA	1	2018	822493

Result 2:

Row	state	month	total_num_of_orders
3	SC	1	840858
4	RS	1	1250664
5	RJ	1	3149840
6	ES	1	487994
7	PE	1	407128
8	MA	1	229182
9	BA	1	839232
10	DF	1	522878
11	CE	1	354502
12	GO	1	631260

Insight: Again, month-on-month analysing the trend in the number of orders placed over the past years is essential for understanding the business's growth trajectory. By examining the order volume month by month, we can identify whether there is a consistent upward trend indicating growth or not. Also, it will help us to check for seasonality in the orders placed In either increasing or decreasing order. This insight is crucial for strategic planning, resource allocation, and forecasting future business performance.

3.2 How are the customers distributed across all the states?

Query:

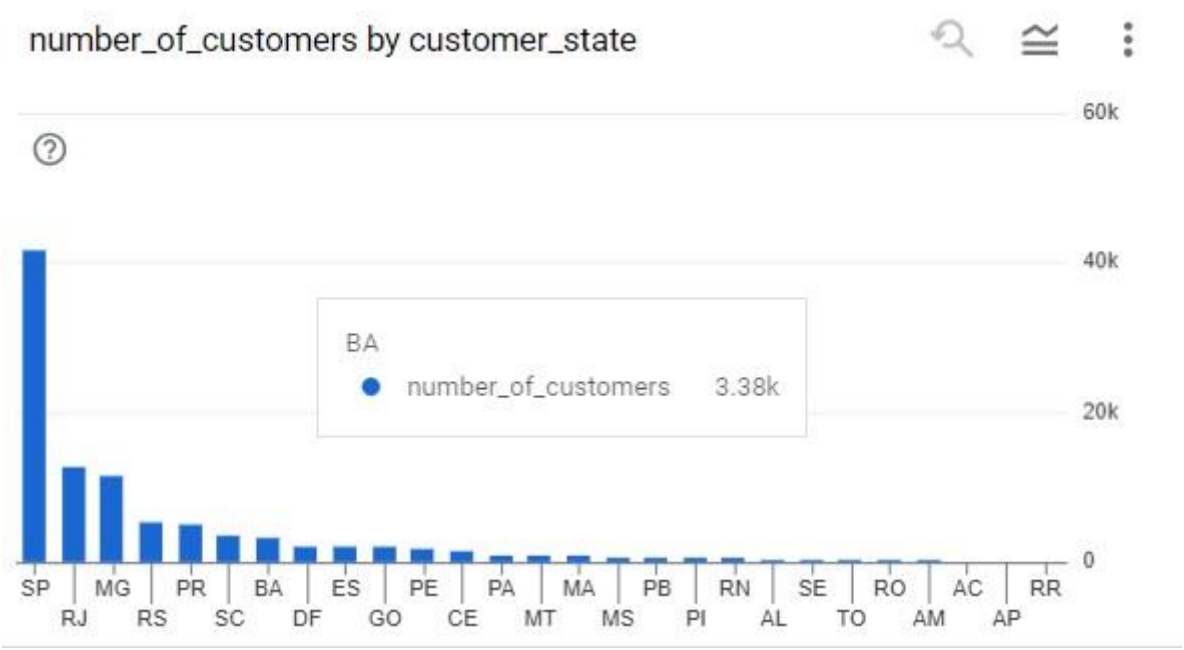
```sql
select
customer_state,
count(customer_id) num_of_customers
from
target.customers
group by
customer_state
order by
num_of_customers desc,
customer_state;
```

Result:

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	customer_state	num_of_customers
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



Insight: Understanding the distribution of customers across all states provides valuable insights into the geographic reach and market penetration of the business.

State SP has maximum number of customers.

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

4.1 Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only).
You can use the "payment_value" column in the payments table to get the cost of orders.

Query:

```sql
WITH total_payments_2017 AS (
SELECT
SUM(payment_value) AS total_2017
FROM target.payments p
JOIN target.orders o ON p.order_id = o.order_id
WHERE EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017
AND EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8
),
total_payments_2018 AS (
SELECT
SUM(payment_value) AS total_2018
FROM target.payments p
JOIN target.orders o ON p.order_id = o.order_id
WHERE EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2018
AND EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8
)
SELECT
total_2017,
total_2018,
((total_2018 - total_2017) / total_2017) * 100 AS percentage_increase
FROM total_payments_2017, total_payments_2018
```

Result:

Row	total_2017	total_2018	percentage_increase
1	3669022.120000...	8694733.839999...	136.9768716466...

Insight: Calculating the percentage increase in the cost of orders from year 2017 to 2018 provides valuable insights into the pricing dynamics and potential changes in customer spending behaviour over time. By focusing on the months between January to August, we capture a significant portion of the annual business activity and can assess trends in order costs during this period.

There is almost 137% increase the cost between 2017 and 2018.

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

Query:

```sql
SELECT
c.customer_state AS state,
SUM(p.payment_value) AS total_order_value,
AVG(p.payment_value) AS average_order_value
FROM target.customers AS c
JOIN target.orders AS o ON c.customer_id = o.customer_id
JOIN target.payments AS p ON o.order_id = p.order_id
GROUP BY state
ORDER BY total_order_value DESC
```

Result:

Row	state	total_order_value	average_order_value
1	SP	5998226.959999...	137.5046297739...
2	RJ	2144379.689999...	158.5258882235...
3	MG	1872257.260000...	154.7064336473...
4	RS	890898.5399999...	157.1804057868...

Insight: Calculating the total and average value of order prices for each state provides valuable insights into the revenue generated from different regions and the spending behaviour of customers in each state. By analysing the total order price, businesses can identify states with the highest and lowest revenue contributions, helps understand the purchasing power and preferences of customers in each state.

Recommendations

Targeted Marketing: Customize marketing strategies for states with high average order values to enhance customer value, focusing on premium products and upselling opportunities.

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

Query:

```
SELECT
c.customer_state AS state,
SUM(oi.freight_value) AS total_freight_value,
AVG(oi.freight_value) AS average_freight_value
FROM target.customers AS c
JOIN target.orders AS o ON c.customer_id = o.customer_id
JOIN target.order_items AS oi ON o.order_id = oi.order_id
GROUP BY state
ORDER BY state
```

Result:

Row	state	total_freight_value	average_freight_valu
1	AC	3686.749999999...	40.07336956521...
2	AL	15914.58999999...	35.84367117117...
3	AM	5478.889999999...	33.20539393939...
4	AP	2788.500000000...	34.00609756097...
5	BA	100156.6799999...	26.36395893656...
6	CE	48351.58999999...	32.71420162381...
7	DF	50625.49999999...	21.04135494596...
8	ES	49764.59999999...	22.05877659574...
9	GO	53114.97999999...	22.76681525932...
10	MA	31523.77000000...	38.25700242718...

Insight: Understanding the total and average order freight costs per state is crucial for logistics and operational planning. By analysing these metrics, businesses can identify states with higher shipping costs and potential inefficiencies in their logistics operations.

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.
Do this in a single query.

You can calculate the delivery time and the difference between the estimated & actual delivery date using the given formula:

- **time_to_deliver** = order_delivered_customer_date - order_purchase_timestamp
- **diff_estimated_delivery** = order_delivered_customer_date - order_estimated_delivery_date

Query:

```sql
SELECT
order_id,
DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY)
AS time_to_deliver,
DATE_DIFF(order_delivered_customer_date, order_estimated_delivery_date,
DAY) AS diff_estimated_delivery
FROM target.orders
WHERE order_status = 'delivered';
```

Result:

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DET

Row	order_id	time_to_deliver	diff_estimated_delive
1	635c894d068ac37e6e03dc54e...	30	-1
2	3b97562c3aee8bdedcb5c2e45...	32	0
3	68f47f50f04c4cb6774570cfde...	29	-1
4	276e9ec344d3bf029ff83a161c...	43	4
5	54e1a3c2b97fb0809da548a59...	40	4
6	fd04fa4105ee8045f6a0139ca5...	37	1
7	302bb8109d097a9fc6e9cefc5...	33	5
8	66057d37308e787052a32828...	38	6
9	19135c945c554eebfd7576c73...	36	2
10	4493e45e7ca1084efcd38ddeb...	34	0

Insights:

Delivery Performance: The time_to_deliver values vary from 29 to 43 days. This range suggests that delivery times can be quite long, possibly due to logistics complexity or operational delays.

Estimated vs. Actual Delivery: The diff_estimated_delivery column contains both negative and positive values. Negative values indicate orders were delivered before the estimated delivery date, whereas positive values indicate that delivery was later than estimated.

Early Deliveries: Orders with a negative diff_estimated_delivery reflect well on the logistics and supply chain efficiency. Customers receiving orders earlier than expected can lead to higher customer satisfaction and trust in the service.

Late Deliveries: Positive values in the diff_estimated_delivery column suggest that these orders were delivered later than Target estimated to customers. Late deliveries can lead to customer

dissatisfaction and potentially impact the company's reputation.

Recommendations:

Improve Logistics: For orders with long delivery times and late deliveries, a review of the logistics process is advisable. Improving partnerships with courier services or optimizing routes can reduce delivery times.

.

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

Top 5 states with highest average freight value

Query:

```
WITH StateFreight AS (
SELECT
c.customer_state AS state,
AVG(oi.freight_value) AS average_freight_value
FROM target.customers AS c
JOIN target.orders AS o ON c.customer_id = o.customer_id
JOIN target.order_items AS oi ON o.order_id = oi.order_id
GROUP BY state
)
SELECT * FROM (
SELECT
state,
average_freight_value
FROM StateFreight
ORDER BY average_freight_value DESC
LIMIT 5
)
```

Result:

JOB INFORMATION | RESULTS | CHART | JSON

Row	state	average_freight_valu
1	RR	42.98442307692...
2	PB	42.72380398671...
3	RO	41.06971223021...
4	AC	40.07336956521...
5	PI	39.14797047970...

Top 5 states with Lowest average freight value

Query:

```sql
WITH StateFreight AS (
SELECT
c.customer_state AS state,
AVG(oi.freight_value) AS average_freight_value
FROM target.customers AS c
JOIN target.orders AS o ON c.customer_id = o.customer_id
JOIN target.order_items AS oi ON o.order_id = oi.order_id
GROUP BY state
)
SELECT * FROM (
SELECT
state,
average_freight_value
FROM StateFreight
ORDER BY average_freight_value
LIMIT 5
)
```

Result:

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	state	average_freight_valu
1	SP	15.14727539041...
2	PR	20.53165156794...
3	MG	20.63016680630...
4	RJ	20.96092393168...
5	DF	21.04135494596...

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

Top 5 states with lowest average delivery time

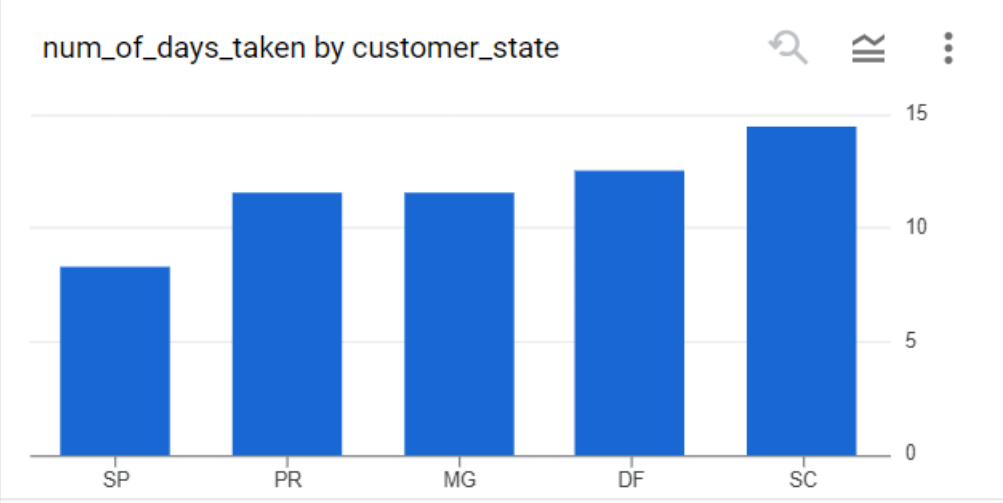
Query:

```sql
select
customer_state,
ROUND(AVG(DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,day)),2) as num_of_days_taken
from
target.orders as o join target.customers as c
on o.customer_id=c.customer_id
group by
customer_state
order by
num_of_days_taken,customer_state
limit 5;
```

Result:

Row	customer_state	num_of_days_taken
1	SP	8.3
2	PR	11.53
3	MG	11.54
4	DF	12.51
5	SC	14.48

Chart:



Top 5 states with highest average delivery time

Query:

Result:

Row	customer_state	no_of_days_taken
1	RR	28.98
2	AP	26.73
3	AM	25.99
4	AL	24.04
5	PA	23.32

Chart:

Insight: Identifying the top 5 states with the highest and lowest average delivery times provides insights into the efficiency of logistics operations and the overall customer experience in different regions.

5.4 Find out the top 5 states where the order delivery is really fast as compared to the estimated date of delivery.
You can use the difference between the averages of actual & estimated delivery date to figure out how fast the delivery was for each state.

Query:

```
WITH DeliverySpeed AS (
SELECT
c.customer_state,
AVG(TIMESTAMP_DIFF(o.order_delivered_customer_date,
o.order_estimated_delivery_date, DAY)) AS avg_days_early
FROM
target.orders o
JOIN
target.customers c ON o.customer_id = c.customer_id
WHERE
o.order_status = 'delivered' AND
o.order_delivered_customer_date IS NOT NULL AND
o.order_estimated_delivery_date IS NOT NULL
GROUP BY
c.customer_state
)
SELECT
customer_state AS state,
avg_days_early
FROM
```

```
DeliverySpeed
WHERE
avg_days_early < 0
ORDER BY
avg_days_early
LIMIT 5;
```

Result:

JOB INFORMATION | RESULTS | CHART | JSON

Row	state	avg_days_early
1	AC	-19.7625000000...
2	RO	-19.1316872427...
3	AP	-18.7313432835...
4	AM	-18.6068965517...
5	RR	-16.4146341463...

Insight: Identifying the top 5 states where order delivery is faster than the estimated date of delivery provides valuable insights into the efficiency of logistics operations and the ability to meet or exceed customer expectations. Additionally, understanding the factors contributing to faster delivery times in specific states enables businesses to implement best practices and improve overall delivery performance across the board.

6.Analysis based on the payments:

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

Query:

```sql
select
distinct
extract(month from order_delivered_customer_date) as month,
extract(year from order_delivered_customer_date) as payment_year,
p.payment_type,
count(p.order_id) over(partition by payment_type order by extract(month from order_delivered_customer_date),extract(year from
order_delivered_customer_date)) num_of_orders
from
target.payments as p join target.orders as o
on p.order_id=o.order_id
order by
month;
```

Result:

Row	month	payment_year	payment_type	num_of_orders
1	null	null	UPI	593
2	null	null	credit_card	2210
3	null	null	debit_card	44
4	null	null	not_defined	3
5	null	null	voucher	282
6	1	2017	UPI	666
7	1	2018	UPI	2047
8	1	2017	credit_card	2411
9	1	2018	credit_card	7421
10	1	2017	debit_card	48

Insight: Analyzing the month-on-month number of orders placed using different payment types provides insights into customer payment preferences and trends over time. It allows businesses to understand which payment methods are most popular among customers and how their usage may vary seasonally or in response to external factors such as promotions or economic conditions. This information can help businesses tailor their payment processing systems, optimize payment options, and develop targeted marketing strategies to better serve customer needs and preferences.

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

Query:

```
select distinct
count(payment_sequential) as `no_of_orders`
from
target.payments
where payment_sequential >=1
```

Result:

Row	no_of_orders
1	103886

Insights: Analysing the number of orders placed based on the number of payment installments that have been paid provides insights into customer payment behaviours and preferences. It allows businesses to understand how customers choose to pay for

their orders, whether they prefer to pay in full upfront or spread payments over multiple installments.

The payment_sequential column gives the sequences of “installments paid.” So count on it will enable to get number of order placed on the basis of installments that have been paid.

Overall recommendations

Optimize Logistical Operations: Implement best practices observed in the states with faster delivery time and efficient operations across the entire logistics network to improve overall delivery performance and enhance customer satisfaction.

Tailor Payment Options: Payment preferences and trends should be used to optimize payment processing systems and offer flexible payment options that align with customer preferences. Consider promoting payment methods that are popular among customers and streamline the checkout process to enhance the overall shopping experience.

Targeted Marketing Campaigns: Using geographic and historical time line insights one should do tailor marketing campaigns and promotions to specific regions and seasons. Customer distribution data should be used to identify high-value regions and implement targeted marketing strategies to drive customer engagement and sales.

Enhance Customer Service: Delivery times and customer feedback to identify areas for improvement in customer service and order fulfillment processes. Need to focus on reducing delivery times, addressing delivery delays, and enhancing communication with customers to provide a seamless shopping experience and improve overall customer satisfaction.

Data-Driven Decision Making: Continuously monitor key metrics such as order volume, delivery times, payment preferences, and customer feedback to make informed business decisions. Use data analytics to identify trends, opportunities, and areas for improvement, and adjust strategies accordingly to stay competitive and drive business growth.

By implementing these recommendations, businesses can optimize operations, improve customer satisfaction, and drive growth in the highly competitive retail landscape.