Business Case Study

Target SQL

Target is one of the world's most recognized brands and one of America's leading retailers. Target makes itself a preferred shopping destination by offering outstanding value, inspiration, innovation and an exceptional guest experience that no other retailer can deliver.

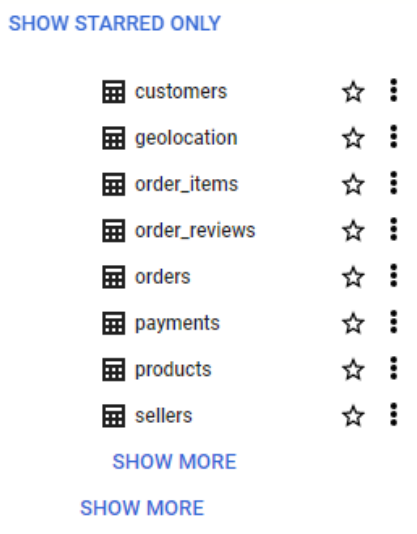
This business case has information of 100k orders from 2016 to 2018 made at Target in Brazil. Its features allows viewing an order from multiple dimensions: from order status, price, payment and freight performance to customer location, product attributes and finally reviews written by customers.

Q1.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

Query:

```
SELECT
        column_name,
        data_type
from dsml_BusinessCase1.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
11	seller_state	STRING

1.2. Time period for which the data is given

Query:

```
SELECT
    date(MIN(order_purchase_timestamp)) AS start_time,
    date(MAX(order_purchase_timestamp)) AS end_time,
    date_diff(MAX(order_purchase_timestamp),MIN(order_purchase_timestamp),DAY)
     as   Time_period_in_days
FROM `dsml_BusinessCase1.orders`;
```

Row	start_time	end_time	Time_period_in_days
1	2016-09-04	2018-10-17	772

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

Query:

```
select
    distinct customer_city as City,
     customer_state as State
from `dsml_BusinessCase1.customers`;
```

Row	City	State
1	acu	RN
2	ico	CE
3	ipe	RS
4	ipu	CE
5	ita	SC
6	itu	SP
7	jau	SP
8	luz	MG
9	poa	SP
10	uba	MG
11	una	BA

Q2. In-depth Exploration:

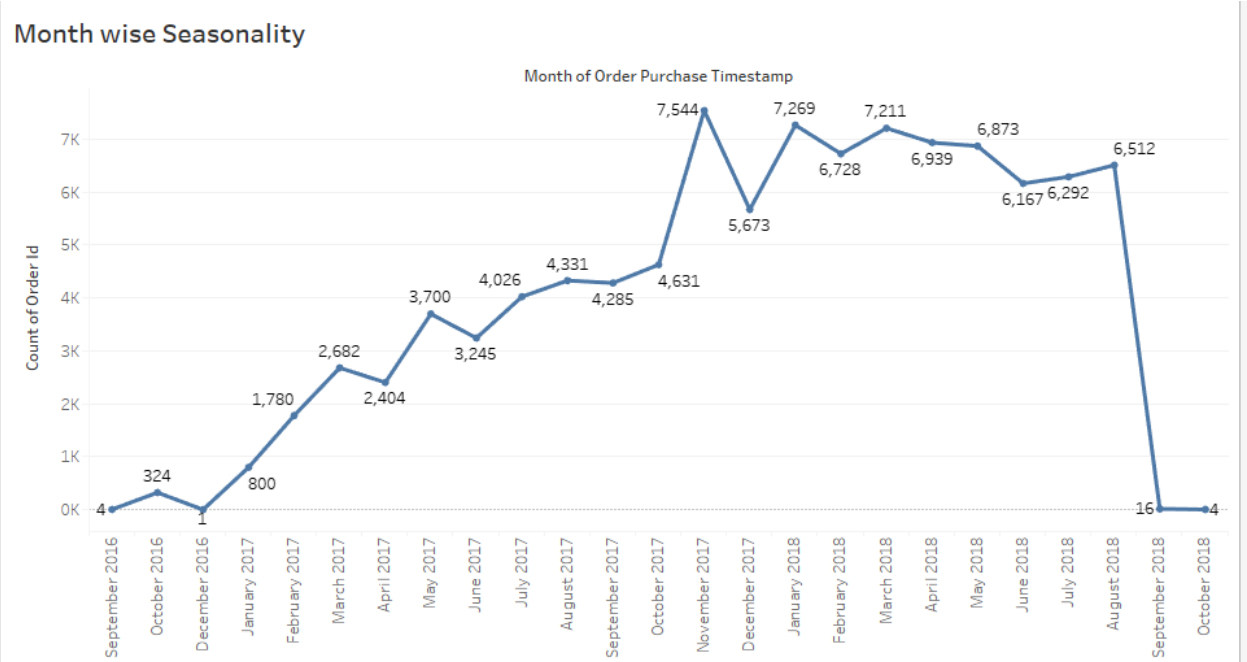
2.1 Is there a growing trend on e-commerce in Brazil? How can we describe a complete scenario? Can we see some seasonality with peaks at specific Months?

Query:

```
with cte as
```

```
(select customer_id, order_id, extract(month from order_purchase_timestamp) Month,
extract(year from order_purchase_timestamp) Year
from `dsml_BusinessCase1.orders`)
 select Month,Year,count(order_id) order_count, from cte
 group by  Year, Month
 order by Year, Month;
```

Row	Month	Year	order_count
1	9	2016	4
2	10	2016	324
3	12	2016	1
4	1	2017	800
5	2	2017	1780
6	3	2017	2682
7	4	2017	2404
8	5	2017	3700
9	6	2017	3245
10	7	2017	4026
11	8	2017	4331



Insight: Yes there is a growing trend on e-commerce which shows seasonality with peak at specific months.

Many specific months coincide with festive seasons, holidays, or major events that drive increased consumer spending. For example, months like November and December typically experience peaks due to the holiday shopping season, including Black Friday, Cyber Monday, and Christmas. Consumers are more inclined to shop online during these periods to take advantage of discounts, promotions, and the convenience of e-commerce.

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

Query:

```
with cte as
(select  c.customer_id,
extract(hour from o.order_purchase_timestamp) as Time1,
case
when extract(hour from o.order_purchase_timestamp) between 0 and 6 then 'Dawn'
when extract(hour from o.order_purchase_timestamp) between 7 and 12 then 'Morning'
when extract(hour from o.order_purchase_timestamp) between 13 and 17 then 'Afternoon'
else 'Night'
 end as Time_period,
from `dsml_BusinessCase1.customers`c
join `dsml_BusinessCase1.orders` o on c.customer_id=o.customer_id
group by c.customer_id, Time1, Time_period
order by Time1)
select  Time_period,count(customer_id) as customer_ordered
from cte
group by Time_period
order by customer_ordered;
```

Row	Time_period	customer_ordered
1	Dawn	5242
2	Morning	27733
3	Afternoon	32366
4	Night	34100

Insight: The number of people ordering during the night is visibly higher than during the dawn.

Q3. Evolution of E-commerce orders in the Brazil region:

3.1. Get month on month orders by states

Query:

```
with cte as
 (select customer_id, order_purchase_timestamp, order_id
from `dsml_BusinessCase1.orders`),
 cte1 as(select customer_id, customer_state
from `dsml_BusinessCase1.customers` )
select customer_state, extract(month from order_purchase_timestamp) Month,
extract(Year from order_purchase_timestamp) Year, count(order_id)Orders
from cte join cte1
on cte.customer_id=cte1.customer_id
group by customer_state,Year, Month
order by Year, Month;
```

Row	customer_state	Month	Year	Orders
1	RR	9	2016	1
2	RS	9	2016	1
3	SP	9	2016	2
4	SP	10	2016	113
5	RS	10	2016	24
6	RJ	10	2016	56
7	MT	10	2016	3
8	GO	10	2016	9
9	MG	10	2016	40
10	CE	10	2016	8
11	SC	10	2016	11

Insight: Examined the month-on-month trends in e-commerce orders to identify growth patterns in the Brazil region. consistently increasing and decreasing in order volumes over time. Understanding how these external factors correlate with order volumes

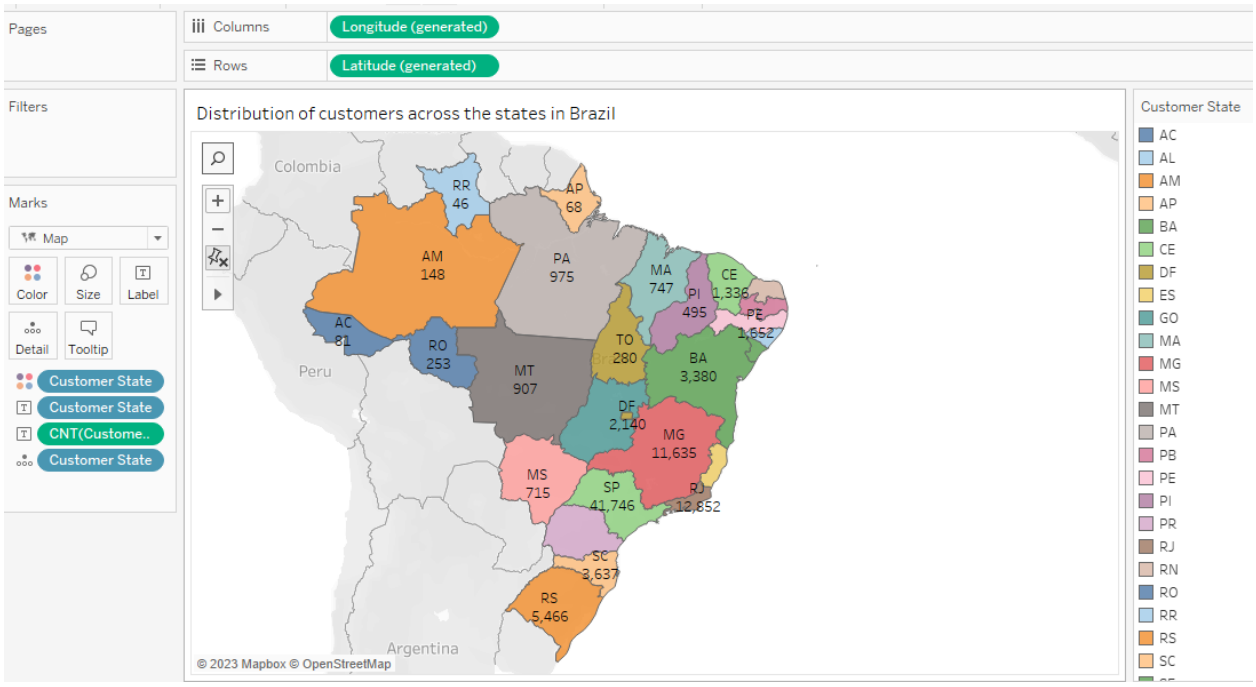
3.2. Distribution of customers across the states in Brazil

Query:

```
select
customer_state, count(customer_id) as customers
from `dsml_BusinessCase1.customers`
group by customer_state;
```

Row	customer_state	customers
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
11	PE	1652

Insight: This insight helps us understand the overall trajectory of e-commerce adoption and popularity in the region.
Understand the concentration of customers in different states to gauge the market potential for your e-commerce business. States with higher population densities may offer larger customer bases and more significant growth opportunities.



Q4. 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

Query:

```
with cte as (select order_id,

extract(Year from order_purchase_timestamp) as Years, extract(month from
order_purchase_timestamp) as Month

from `dsml_BusinessCase1.orders`)

select  round ((SUM(CASE WHEN Years = 2018 THEN payment_value END) - SUM(CASE WHEN
Years = 2017 THEN payment_value END)) / SUM(CASE WHEN Years = 2017 THEN payment_value
END) * 100) AS Percentage_Increase

from cte join `dsml_BusinessCase1.payments` p on cte.order_id=p.order_id

WHERE(Years = 2017 AND Month BETWEEN 1 AND 8) OR (Years = 2018 AND Month BETWEEN 1 AND
8);
```

Row	Percentage_Increase
1	137.0

Insight 1: The payment value from e-commerce transactions has increased by 37% over a year, from 2017 to 2018.

Insight 2: An increase in payment value indicates a potential increase in economic activity and overall economic growth. When more money is being exchanged through payments, It suggests that people and businesses are spending more, which can stimulate the economy.

Insight 3: Increasing payment values could also be a result of inflation.

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

Query:

```
select c.customer_state,
 round(avg(price)) as mean_price,
 round(sum(price)) as sum_price,
 round(avg(freight_value)) as mean_freight,
 round(sum(freight_value)) as sum_freight
from  `dsml_BusinessCase1.order_items` ot left join `dsml_BusinessCase1.orders`
o on ot.order_id=o.order_id
left join `dsml_BusinessCase1.customers` c on o.customer_id=c.customer_id
group by 1
order by 2;
```

Row	customer_state	mean_price	sum_price	mean_freight	sum_freight
1	SP	110.0	5202955.0	15.0	718723.0
2	PR	119.0	683084.0	21.0	117852.0
3	RS	120.0	750304.0	22.0	135523.0
4	MG	121.0	1585308.0	21.0	270853.0
5	ES	122.0	275037.0	22.0	49765.0
6	RJ	125.0	1824093.0	21.0	305589.0
7	SC	125.0	520553.0	21.0	89660.0
8	DF	126.0	302604.0	21.0	50625.0
9	GO	126.0	294592.0	23.0	53115.0
10	BA	135.0	511350.0	26.0	100157.0
11	AM	135.0	22357.0	33.0	5479.0

Insight 1: Tracking changes in average order value over time can help identify trends and patterns in consumer purchasing behavior.

Insight 2: Analyzing the order prices can reveal customer preferences in terms of product categories or price ranges.

Q5. Analysis on sales, freight and delivery time

5.1.Calculate days between purchasing, delivering and estimated delivery

Query:

```
select * from

(SELECT DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day) AS time_to_delivery,

 DATE_DIFF(order_estimated_delivery_date, order_purchase_timestamp, day) AS estimated_time_delivery,

 FROM `dsml_BusinessCase1.orders`

where time_to_delivery is not null and estimated_time_delivery is not null;
```

Row	time_to_delivery	estimated_time_delivery
1	30	17
2	30	59
3	35	52
4	30	32
5	32	33
6	29	31
7	43	39
8	40	36
9	37	35
10	33	28
11	38	32

Insight : By comparing the actual delivery dates with the estimated delivery dates, evaluating the speed and efficiency of the delivery process. If the actual delivery consistently falls within or before the estimated delivery range, it indicates that the

e-commerce company is meeting customer expectations and delivering orders promptly.

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

Query:

```
select * from

(SELECT DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day) AS time_to_delivery,

  DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date, day) AS diff_estimated_delivery

FROM `dsml_BusinessCase1.orders`)

where diff_estimated_delivery is not null;
```

Row	time_to_delivery	diff_estimated_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
11	38	-6

Insight : The days between purchasing, delivering, and estimated delivery can impact communication and transparency between the e-commerce company and the customer. there are significant delays or discrepancies between estimated and actual delivery, it's important for the company to communicate proactively with customers, providing

updates and managing expectations. Monitoring this aspect can help identify areas for improvement in customer communication and transparency.

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

Query:

```
with cte as

(select o.order_id, c.customer_state,

 DATE_DIFF(o.order_delivered_customer_date, o.order_purchase_timestamp, day) AS time_to_delivery,

 DATE_DIFF(o.order_estimated_delivery_date, o.order_delivered_customer_date, day) AS diff_estimated_delivery

 from `dsml_BusinessCase1.orders` o  join `dsml_BusinessCase1.customers` c  on o.customer_id=c.customer_id),

cte1 as

 (select cte.customer_state, round(avg(ot.freight_value)) as mean_freight

 from `dsml_BusinessCase1.order_items` ot join cte on ot.order_id=cte.order_id

 group by customer_state)

select cte.customer_state, cte1.mean_freight, cte.time_to_delivery, cte.diff_estimated_delivery

from cte join cte1 on cte.customer_state=cte1.customer_state

where time_to_delivery is not null and diff_estimated_delivery is not null;
```

Row	customer_state	mean_freight	time_to_delivery	diff_estimated_delivery
1	SE	37.0	29	1
2	TO	37.0	14	12
3	TO	37.0	24	8
4	RN	36.0	17	14
5	RN	36.0	16	19
6	AL	36.0	38	-6
7	RO	41.0	17	16
8	RN	36.0	27	5
9	AM	33.0	13	25
10	PI	39.0	9	18
11	TO	37.0	30	-8

Insight : the diffi_estimated and actual time_to_delivery dates, it indicating issues with order fulfillment, transportation, or other logistics-related challenges that need to be addressed but majorly orders delivered on time.

5.4. Sort the data to get the following:

1. **Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5**

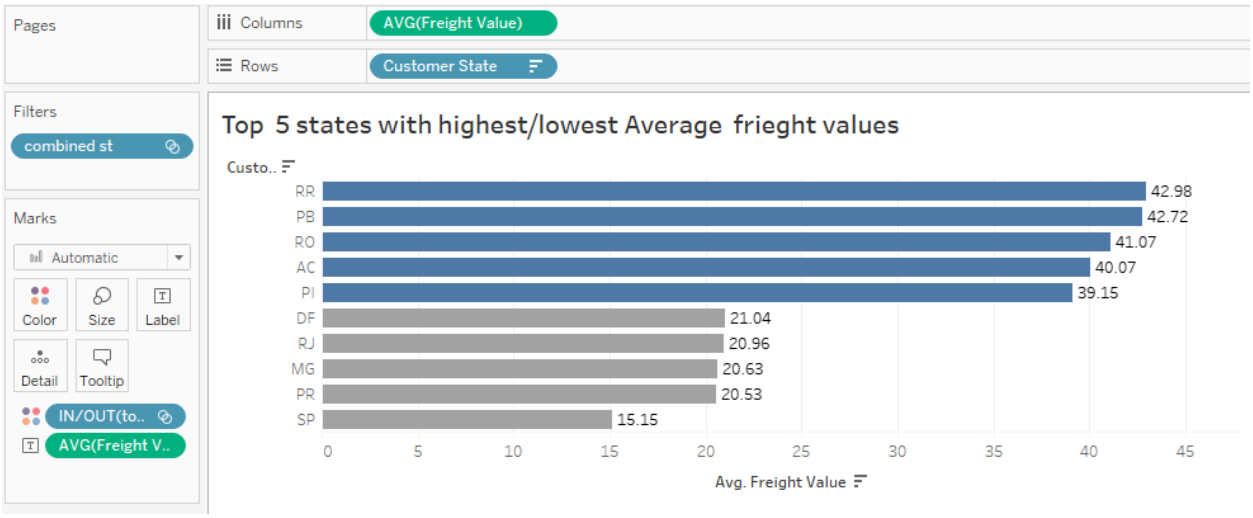
Query:

```
with cte as

 (select o.order_id, c.customer_state,

 from `dsml_BusinessCase1.orders` o join `dsml_BusinessCase1.customers` c on
o.customer_id=c.customer_id),

 cte1 as

(select cte.customer_state, round(avg(ot.freight_value),2) as
highest_and_lowest_avg_freight

 from `dsml_BusinessCase1.order_items` ot join cte on ot.order_id=cte.order_id

 group by customer_state)

(select cte1.customer_state, cte1.highest_and_lowest_avg_freight

from  cte1
```

```
order by cte1.highest_and_lowest_avg_freight desc

limit 5)

union all

(select cte1.customer_state, cte1.highest_and_lowest_avg_freight

from  cte1

order by cte1.highest_and_lowest_avg_freight

limit 5);
```

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

Insight :



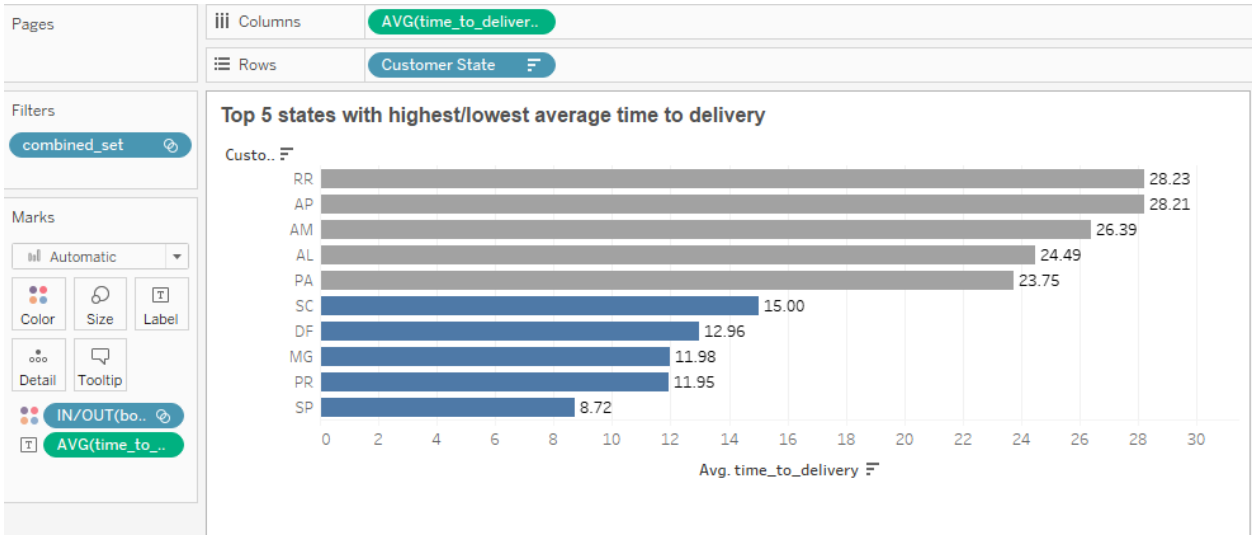
5.6. Top 5 states with highest/lowest average time to delivery

Query:

```
(select c.customer_state,

round(avg(DATE_DIFF(o.order_delivered_customer_date, o.order_purchase_timestamp,
day)),2) AS avg_time_to_delivery

 from `dsml_BusinessCase1.orders` o join `dsml_BusinessCase1.customers` c on
o.customer_id=c.customer_id

 group  by c.customer_state order by avg_time_to_delivery desc limit 5)

 union all

 (select c.customer_state,

round(avg(DATE_DIFF(o.order_delivered_customer_date, o.order_purchase_timestamp,
day)),2) AS avg_time_to_delivery

 from `dsml_BusinessCase1.orders` o join `dsml_BusinessCase1.customers` c on
o.customer_id=c.customer_id

 group  by c.customer_state order by avg_time_to_delivery limit 5 );
```

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

Insight :



5.7. Top 5 states where delivery is really fast/ not so fast compared to estimated date

Query:

```
with cte as

(select customer_id,

case when DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date, day)>=0 then "Fast" else "Not_to_fast"
```

```
 end as delivery_pace
 from `dsml_BusinessCase1.orders`)
 (select c.customer_state, cte.delivery_pace
 from cte join `dsml_BusinessCase1.customers` c on cte.customer_id=c.customer_id
 group by 1,2 order by 2 limit 5)
union all
 (select c.customer_state, cte.delivery_pace
 from cte join `dsml_BusinessCase1.customers` c on cte.customer_id=c.customer_id
 group by 1,2 order by 2 desc limit 5);
```

Row	customer_state	delivery_pace
1	GO	Fast
2	RJ	Fast
3	SC	Fast
4	RS	Fast
5	SP	Fast
6	DF	Not_to_fast
7	RJ	Not_to_fast
8	RS	Not_to_fast
9	PR	Not_to_fast
10	SP	Not_to_fast

Insight : It highlights the areas where our delivery operations are most efficient, effective and with low effective areas, potentially indicating factors like proximity to distribution centers or optimized transportation routes.

Q6.Payment type analysis:

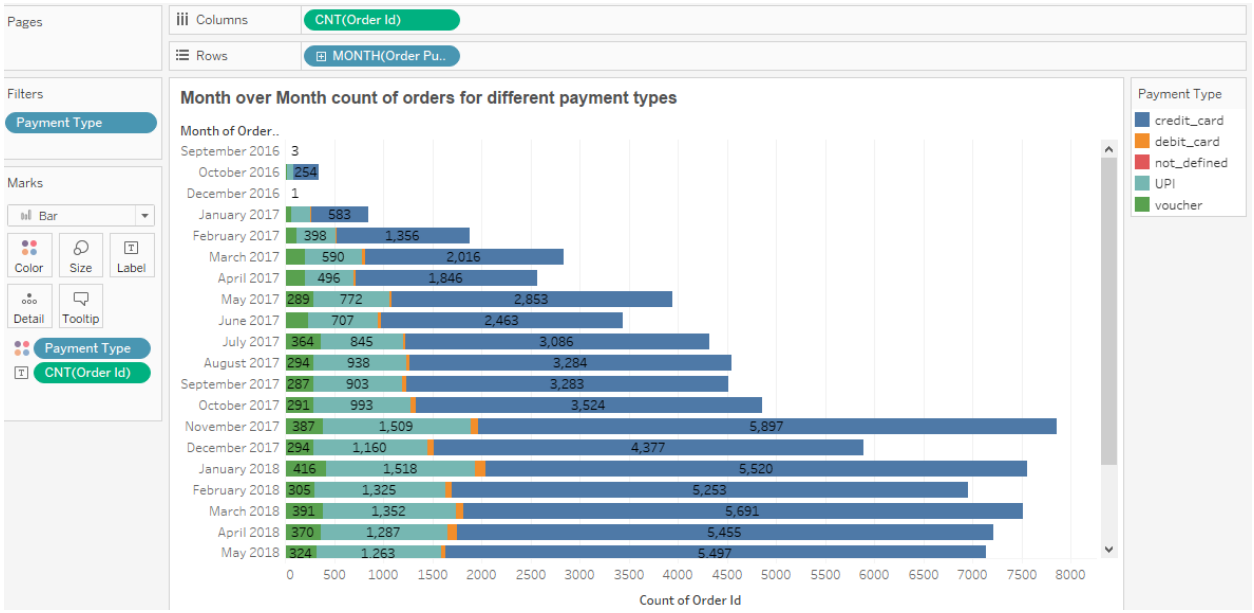
6.1. Month over Month count of orders for different payment types

Query:

```
with cte as
(select order_purchase_timestamp, order_id
from `dsml_BusinessCase1.orders`),
 cte1 as(select order_id, payment_type
from `dsml_BusinessCase1.payments` )
select payment_type, extract(month from order_purchase_timestamp) Month, extract(Year
from order_purchase_timestamp) Year, count(cte.order_id)Orders
from cte join cte1 on cte.order_id=cte1.order_id
group by 1,2,3 order by Year, Month;
```

Row	payment_type	Month	Year	Orders
1	credit_card	9	2016	3
2	credit_card	10	2016	254
3	UPI	10	2016	63
4	voucher	10	2016	23
5	debit_card	10	2016	2
6	credit_card	12	2016	1
7	credit_card	1	2017	583
8	UPI	1	2017	197
9	voucher	1	2017	61
10	debit_card	1	2017	9
11	credit_card	2	2017	1356

Insight : By comparing the count of orders for different payment types over time, we can identify the popularity of each payment method. This helps us to understand which payment options are preferred by your customers. It can guide decisions regarding which payment methods to prioritize and optimize in order to cater to customer preferences.



6.2.Count of orders based on the no. of payment installments

Query: `select`

```
Payment_installments, count(order_id)count_of_orders,

from `dsml_BusinessCase1.payments` group by 1;
```

Row	payment_installments	count_of_orders
1	0	2
2	1	52546
3	2	12413
4	3	10461
5	4	7098
6	5	5239
7	6	3920
8	7	1626
9	8	4268
10	9	644
11	10	5328

Insight : we can understand the affordability preferences of your customers. It helps to identify whether customers prefer to pay for their purchases in full or if they opt for installment-based payment plans. This insight guides pricing strategies and informs product offerings to cater to different affordability preferences.

Overall recommendations

Through observing the history of the company's datasets, clear ideas on the sales for the previous years were realized which will be very helpful to the company on its own. Additionally, seasonality trend and randomness and future forecasts will help to analyze sales drops which the companies can avoid by using more focused and efficient tactics to minimize the sale drop and maximize the profit and remain in competition.

There are other companies who are constantly rising as well and would give a tough competition in the future if it does not stay at the top of their game. In order to do so, they will need to understand their business trends, the customer needs and manage the resources wisely.

Transparent and timely communication builds trust and improves the overall customer experience.

Adapt your strategies, inventory management, and marketing efforts to align with peak seasons and leverage opportunities during high-demand periods.