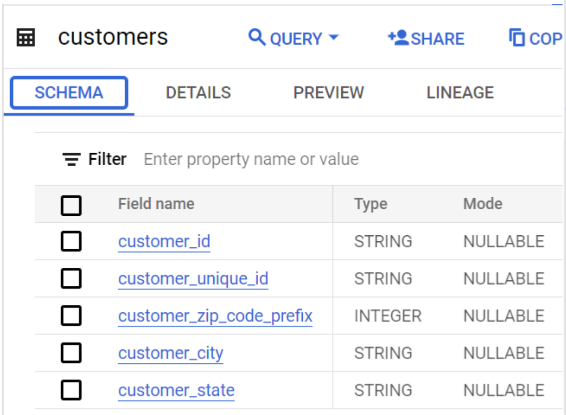
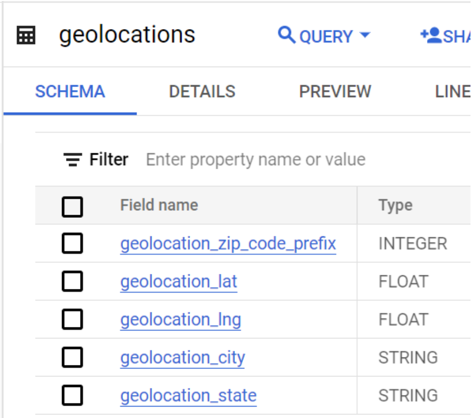
1. Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset

1.1 Data type of columns in a table



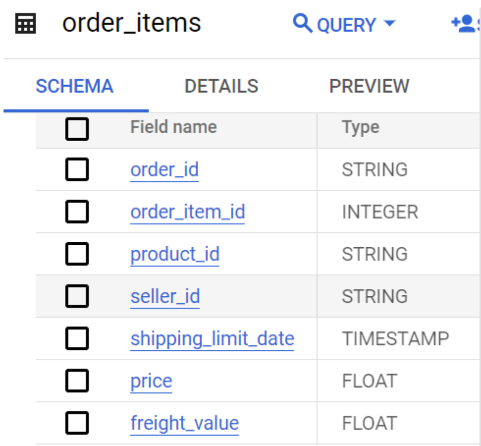
customers

□	Field name	Type	Mode
□	customer_id	STRING	NULLABLE
□	customer_unique_id	STRING	NULLABLE
□	customer_zip_code_prefix	INTEGER	NULLABLE
□	customer_city	STRING	NULLABLE
□	customer_state	STRING	NULLABLE



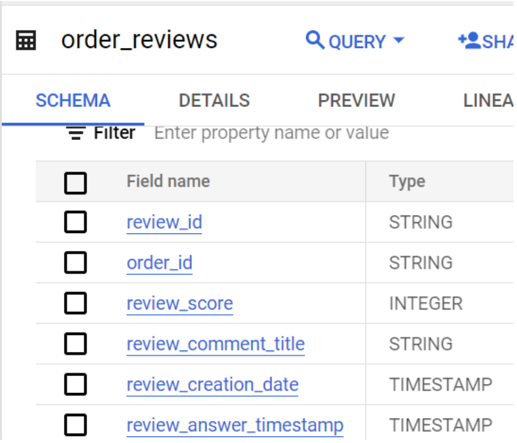
geolocations

□	Field name	Type
□	geolocation_zip_code_prefix	INTEGER
□	geolocation_lat	FLOAT
□	geolocation_lng	FLOAT
□	geolocation_city	STRING
□	geolocation_state	STRING



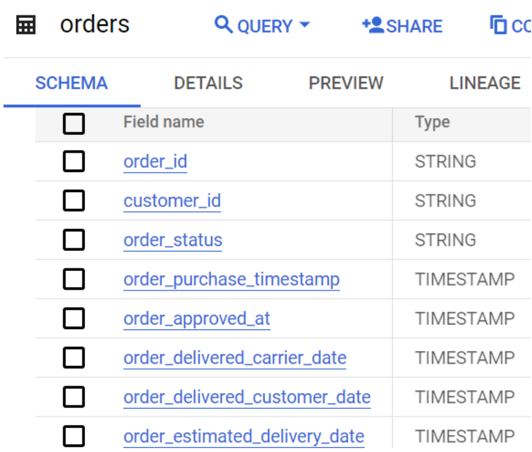
order_items

□	Field name	Type
□	order_id	STRING
□	order_item_id	INTEGER
□	product_id	STRING
□	seller_id	STRING
□	shipping_limit_date	TIMESTAMP
□	price	FLOAT
□	freight_value	FLOAT



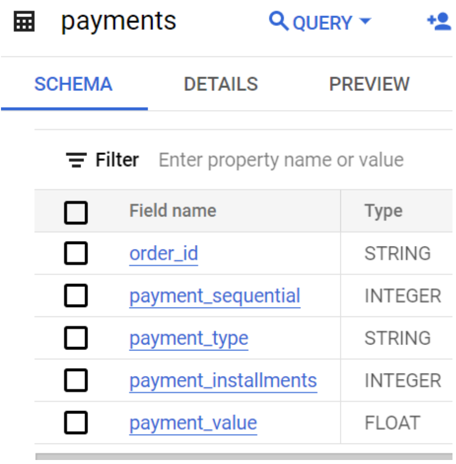
order_reviews

□	Field name	Type
□	review_id	STRING
□	order_id	STRING
□	review_score	INTEGER
□	review_comment_title	STRING
□	review_creation_date	TIMESTAMP
□	review_answer_timestamp	TIMESTAMP



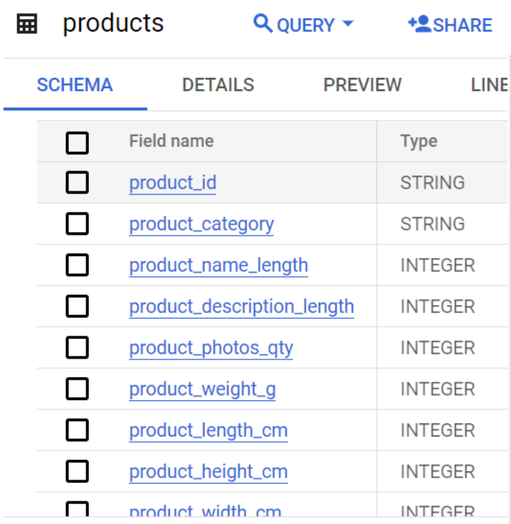
orders

□	Field name	Type
□	order_id	STRING
□	customer_id	STRING
□	order_status	STRING
□	order_purchase_timestamp	TIMESTAMP
□	order_approved_at	TIMESTAMP
□	order_delivered_carrier_date	TIMESTAMP
□	order_delivered_customer_date	TIMESTAMP
□	order_estimated_delivery_date	TIMESTAMP

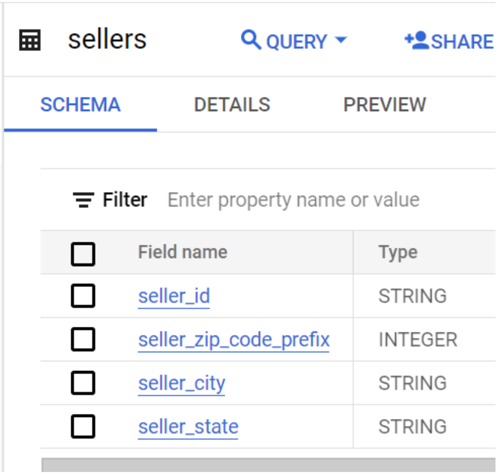


payments

□	Field name	Type
□	order_id	STRING
□	payment_sequential	INTEGER
□	payment_type	STRING
□	payment_installments	INTEGER
□	payment_value	FLOAT



□	Field name	Type
□	product_id	STRING
□	product_category	STRING
□	product_name_length	INTEGER
□	product_description_length	INTEGER
□	product_photos_qty	INTEGER
□	product_weight_g	INTEGER
□	product_length_cm	INTEGER
□	product_height_cm	INTEGER
□	product_width_cm	INTEGER

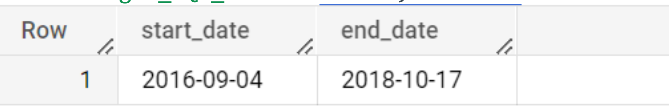


□	Field name	Type
□	seller_id	STRING
□	seller_zip_code_prefix	INTEGER
□	seller_city	STRING
□	seller_state	STRING

In the Target SQL project, the data set given has total of 8 tables with different information of customers, orders, product, payments, sellers, location etc. The data types present are of Integer, String, Float and Timestamp.

1.2 Time period for which the data is given

```
 select
min(date(order_purchase_timestamp)) as start_date,
max(date(order_purchase_timestamp)) as end_date
from `Target_SQL_Intro.orders`;
```

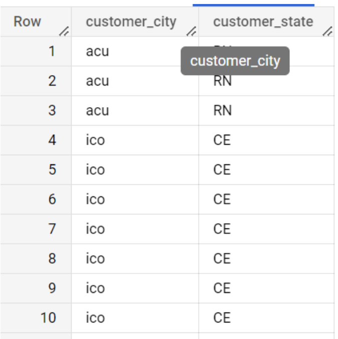


Row	start_date	end_date
1	2016-09-04	2018-10-17

Insight- The business case has information of almost 26 months starting from September 2016 to October 2018.

1.3 Cities and States of customers ordered during the given period

```
select a.customer_city, a.customer_state
from `Target_SQL_Intro.customers` a
inner join
`Target_SQL_Intro.orders` b on
a.customer_id=b.customer_id
```



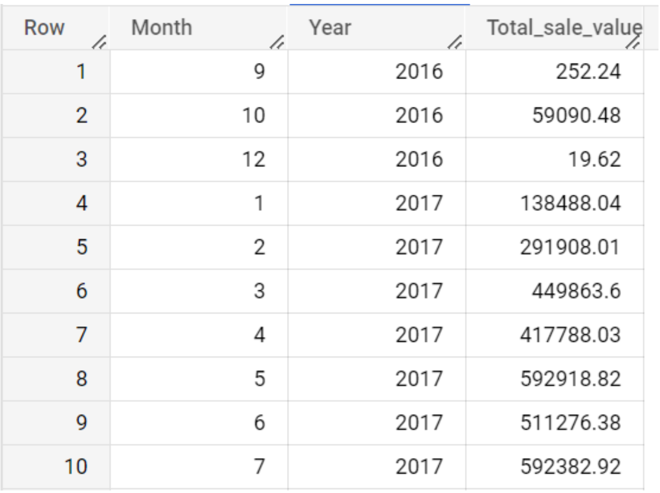
Row	customer_city	customer_state
1	acu	RN
2	acu	RN
3	acu	RN
4	ico	CE
5	ico	CE
6	ico	CE
7	ico	CE
8	ico	CE
9	ico	CE
10	ico	CE

Insight – Not all the registered customers have ordered from Target. The order were from placed from 4310 different cities from 27 different states.
Recommendation- Provide special one time deals to passive/inactive customers.

2. In-depth Exploration

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

```
select extract(month from order_purchase_timestamp) as Month,
extract(year from order_purchase_timestamp) as Year,
round(sum(payment_value),2) as Total_sale_value from
(select * from `Target_SQL_Intro.payments` a left join
`Target_SQL_Intro.orders` b on
a.order_id=b.order_id)
group by 1,2
order by 2,1;
```

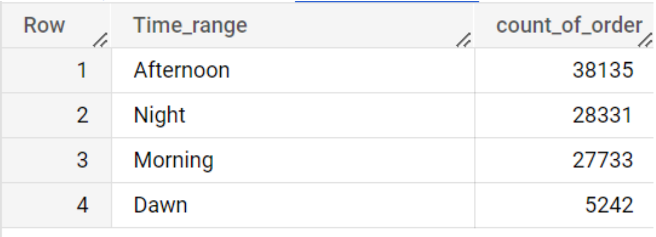


Row	Month	Year	Total_sale_value
1	9	2016	252.24
2	10	2016	59090.48
3	12	2016	19.62
4	1	2017	138488.04
5	2	2017	291908.01
6	3	2017	449863.6
7	4	2017	417788.03
8	5	2017	592918.82
9	6	2017	511276.38
10	7	2017	592382.92

Insight- Overall there is a growing trend in sales in Brazil. In the first three months of launch in 2016, the sales were minimal. But during Winter, i.e between March and October there was a good increase in sales in 2017 and 2018 compared to the summer(November to March) maybe because people prefer to order online than going out during winter.
Recommendation- Provide exciting deals during Summer to increase the no of orders.
Set product recommendation which are specific to the season.
Prepare inventory, manpower and logistic for the peak period.

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

```
select Time_range, count(Time_range) as count_of_order from
(select
    case
      when hour >=19 then 'Night'
      when hour >=13 then 'Afternoon'
      when hour >=7 then 'Morning'
      else 'Dawn'
    end as Time_range from
(select extract(hour from order_purchase_timestamp) as hour from
`Target_SQL_Intro.orders`))
group by 1
order by 2 desc;
```



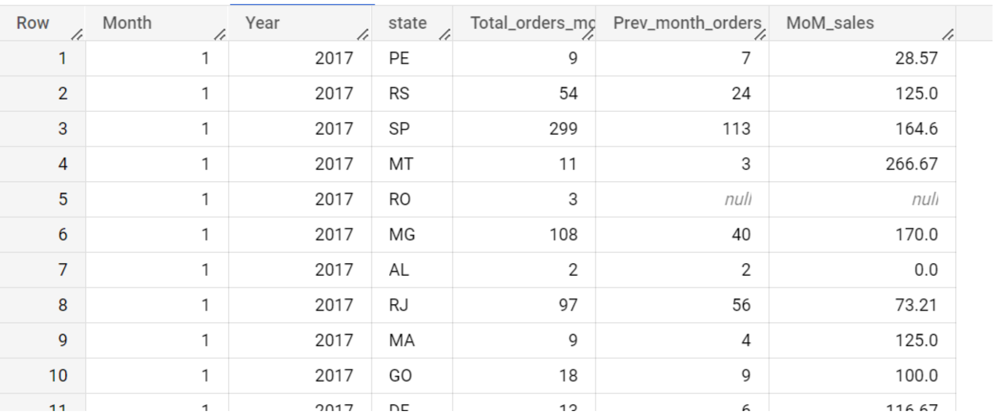
Row	Time_range	count_of_order
1	Afternoon	38135
2	Night	28331
3	Morning	27733
4	Dawn	5242

Insight- In Brazil, Target received majority of their orders in the afternoon(1pm to 6pm) followed by almost evenly during night(7pm to 12AM) and Moring(7AM to 12PM).
Recommendation- Arrange limited time offers during this period to attract more traffic.

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

3.1 Get month on month orders by states

```
select *, round((100*(Total_orders_month-Prev_month_orders)/Prev_month_orders),2) as
MoM_sales from
(select *, lag(Total_orders_month,1) over(partition by state order by Year,Month) as
Prev_month_orders
from (select
extract(month from b.order_purchase_timestamp) as Month,
extract(year from b.order_purchase_timestamp) as Year,
a.customer_state as state,
count(distinct(b.order_id)) as Total_orders_month
from `Target_SQL_Intro.customers` a
inner join `Target_SQL_Intro.orders` b
on a.customer_id=b.customer_id
group by 3,1,2
order by 2,1)
order by Month,Year)
```

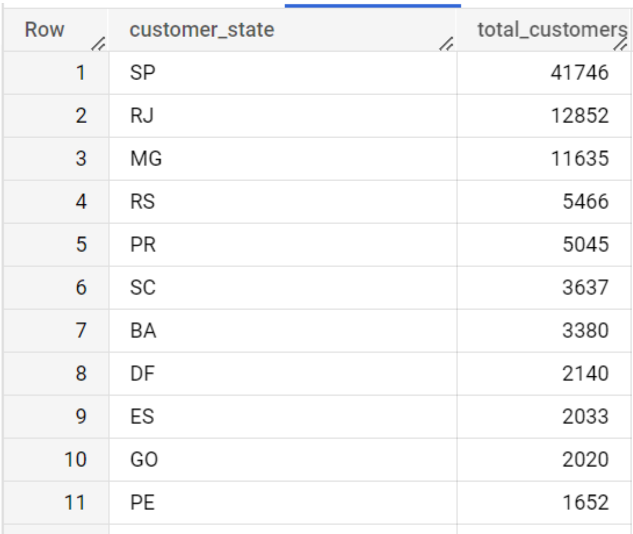


Row	Month	Year	state	Total_orders_mc	Prev_month_orders	MoM_sales
1	1	2017	PE	9	7	28.57
2	1	2017	RS	54	24	125.0
3	1	2017	SP	299	113	164.6
4	1	2017	MT	11	3	266.67
5	1	2017	RO	3	null	null
6	1	2017	MG	108	40	170.0
7	1	2017	AL	2	2	0.0
8	1	2017	RJ	97	56	73.21
9	1	2017	MA	9	4	125.0
10	1	2017	GO	18	9	100.0
11	1	2017	DF	13	6	116.67

Insight- Overall there was a good increase in total order in all the states. On ordering the data by month, it looks like there was drop in count of orders in almost all states during December 2017, August 2018 and June 2017.
Recommendation- In depth analysis and market research to be done to understand why the no of order dipped during the period of December 2017, August 2018 and June 2017.

3.2 Distribution of customers across the states in Brazil

```
select customer_state, count(customer_id) as total_customers
from `Target_SQL_Intro.customers`
group by customer_state
order by total_customers desc;
```



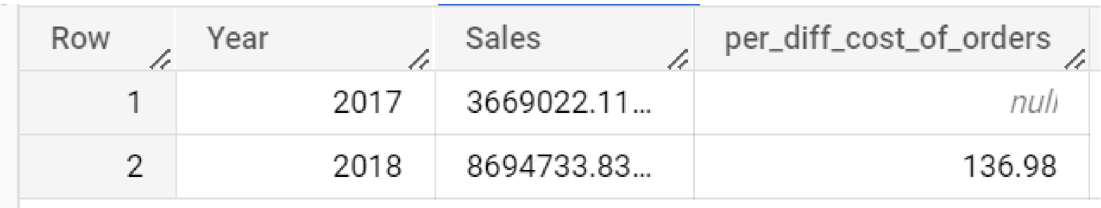
Row	customer_state	total_customers
1	SP	41746
2	RJ	12852
3	MG	11635
4	RS	5466
5	PR	5045
6	SC	3637
7	BA	3380
8	DF	2140
9	ES	2033
10	GO	2020
11	PE	1652

Insight- There are almost 4 times the customers in State SP(could be a metro city) compared to its next RJ and MG. Almost 10 of states have customers in between the range of 5000 to 2000. Over 50% of the states have less than 1000 customers.
Recommendation- Formulate strategies to target Tier 2 and Tier 3 cities where total customer are low. Expand product basket relevant to their requirement.

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

```
select Year,Sales, round(100*(sales-sales_prev_year)/sales_prev_year,2) as
per_diff_cost_of_orders from(
select Year,sales, lag((sales)) over(order by Year asc) sales_prev_year from(
select Year, sum(Total_sales) as sales from
(select Month, Year, Total_sales from
(select ID,extract(month from time_stamp) as Month,
extract(year from time_stamp) as Year, round(sum(amount),2) as Total_sales from
(select a.order_id as ID, a.order_purchase_timestamp as time_stamp,
sum(b.payment_value) as amount from `Target_SQL_Intro.orders` a left join
`Target_SQL_Intro.payments` b on
a.order_id=b.order_id
group by 1,2)
group by 1,2,3)
where (Year between 2017 and 2018) and (Month between 1 and 8)
order by Year, Month)
group by Year)
order by Year)
```

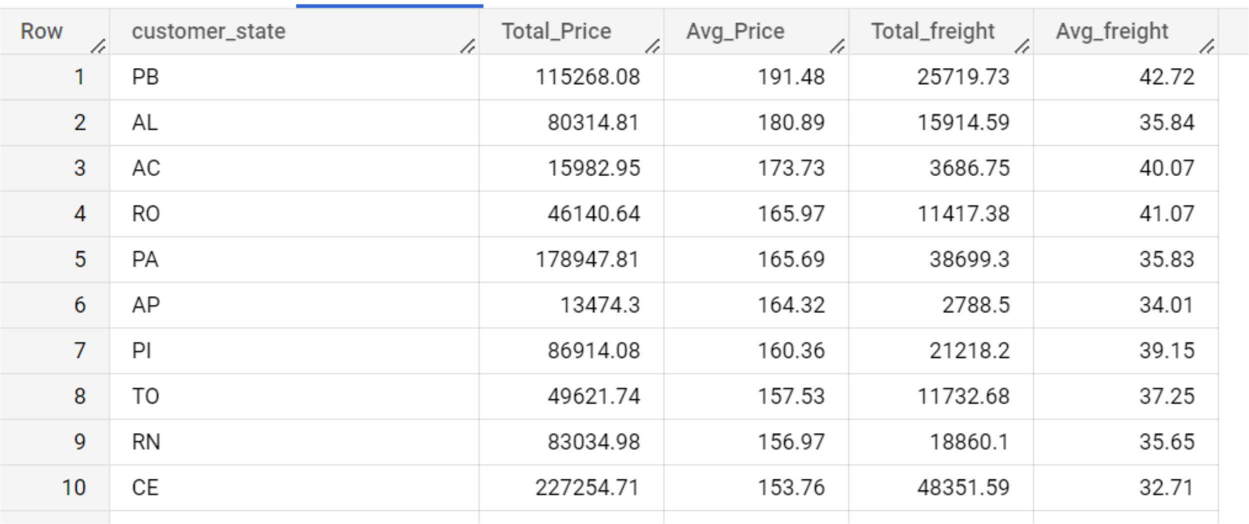


Row	Year	Sales	per_diff_cost_of_orders
1	2017	3669022.11...	*null*
2	2018	8694733.83...	136.98

Insight- There is huge growth of 137% in total sales in year 2018 compared to sales in 2017. This is almost 2.4x growth yoy. The sales data is only for 10 months of 2018 against that of 12 months in 2017. This shows the growth of Target in the Brazilian market.

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

```
select customer_state, round(sum(price),2) as Total_Price, round(avg(price),2) as
Avg_Price,
round(sum(freight_value),2) as Total_freight, round(avg(freight_value),2) as
Avg_freight from
(select * from `Target_SQL_Intro.orders` a  left join
`Target_SQL_Intro.order_items` b on a.order_id=b.order_id
left join `Target_SQL_Intro.customers` c on a.customer_id=c.customer_id)
group by customer_state
order by 3 desc;
```



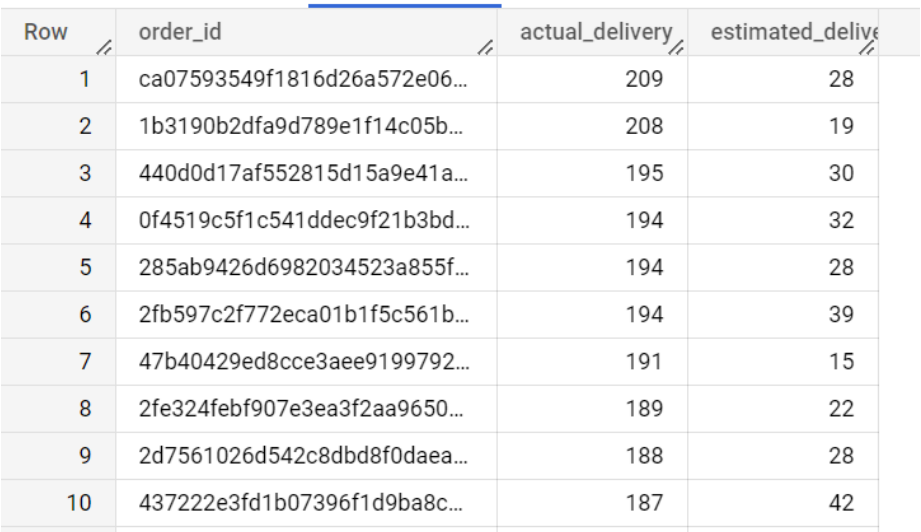
Row	customer_state	Total_Price	Avg_Price	Total_freight	Avg_freight
1	PB	115268.08	191.48	25719.73	42.72
2	AL	80314.81	180.89	15914.59	35.84
3	AC	15982.95	173.73	3686.75	40.07
4	RO	46140.64	165.97	11417.38	41.07
5	PA	178947.81	165.69	38699.3	35.83
6	AP	13474.3	164.32	2788.5	34.01
7	PI	86914.08	160.36	21218.2	39.15
8	TO	49621.74	157.53	11732.68	37.25
9	RN	83034.98	156.97	18860.1	35.65
10	CE	227254.71	153.76	48351.59	32.71

Insight- There is clear trend in increase of average price with the increase in freight. The range for average price is between 192 to 109 and for average freight charge is 42 to 15.
Recommendation- Controlling the variance of Freight charge can bring down the average price of order considerably. Improvement in logistic and proper inventory planning can help bring down the freight cost.

5. Analysis on sales, freight and delivery time

5.1 Calculate days between purchasing, delivering and estimated delivery

```
select order_id,
timestamp_diff(order_delivered_customer_date,order_purchase_timestamp,day)
as actual_delivery,
timestamp_diff(order_estimated_delivery_date,order_purchase_timestamp,day) as
estimated_delivery from `Target_SQL_Intro.orders`
order by 2 desc;
```

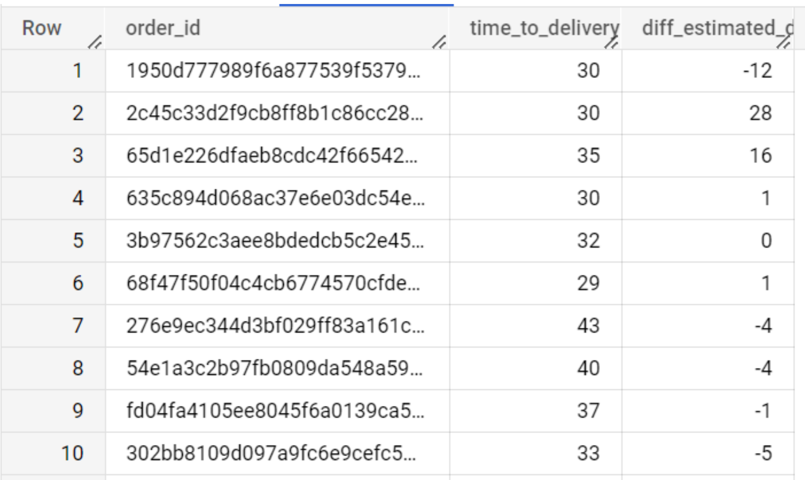


Row	order_id	actual_delivery	estimated_delive
1	ca07593549f1816d26a572e06...	209	28
2	1b3190b2dfa9d789e1f14c05b...	208	19
3	440d0d17af552815d15a9e41a...	195	30
4	0f4519c5f1c541ddec9f21b3bd...	194	32
5	285ab9426d6982034523a855f...	194	28
6	2fb597c2f772eca01b1f5c561b...	194	39
7	47b40429ed8cce3aee9199792...	191	15
8	2fe324febf907e3ea3f2aa9650...	189	22
9	2d7561026d542c8dbd8f0daea...	188	28
10	437222e3fd1b07396f1d9ba8c...	187	42

Insight- There are deliveries which were completed well before the estimated delivery time. But there are many deliveries which took 4 to 5 times than the estimated delivery.
Recommendation- Logistics, man power and inventory has to be planned well in advance and be equipped for the peak sales period. Customer should be provided with accurate estimate delivery details to avoid any sort for negative experience.

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

```
Select order_id,
timestamp_diff(order_delivered_customer_date,order_purchase_timestamp,day) as
time_to_delivery,
timestamp_diff(order_estimated_delivery_date,order_delivered_customer_date,day) as
diff_estimated_delivery from `Target_SQL_Intro.orders`
```

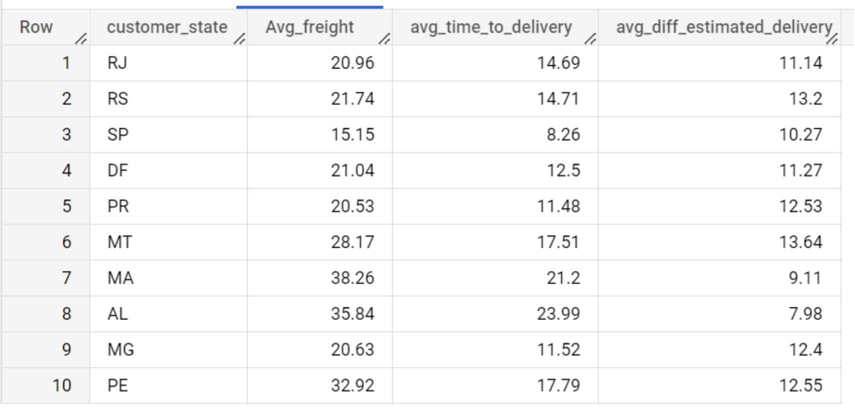


Row	order_id	time_to_delivery	diff_estimated_d
1	1950d777989f6a877539f5379...	30	-12
2	2c45c33d2f9cb8ff8b1c86cc28...	30	28
3	65d1e226dfaeb8cdc42f66542...	35	16
4	635c894d068ac37e6e03dc54e...	30	1
5	3b97562c3aee8bdedcb5c2e45...	32	0
6	68f47f50f04c4cb6774570cfde...	29	1
7	276e9ec344d3bf029ff83a161c...	43	-4
8	54e1a3c2b97fb0809da548a59...	40	-4
9	fd04fa4105ee8045f6a0139ca5...	37	-1
10	302bb8109d097a9fc6e9cefc5...	33	-5

Insight- There are orders which were supposed to be delivered after 50 days, but got delivered within 3 days. Also there are order which were delivered too long after the estimated time.
Recommendation- Customer should be provided with proper tracking details to track their order and also their should be a proper escalation matrix and customer handling team to address issues related delayed delivery

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

```
select customer_state, round(avg(freight_value),2) as Avg_freight,
round(avg(timestamp_diff(order_delivered_customer_date,order_purchase_timestamp,day)),
2) as avg_time_to_delivery,
round(avg(timestamp_diff(order_estimated_delivery_date,order_delivered_customer_date,d
ay)),2) as avg_diff_estimated_delivery from (select * from `Target_SQL_Intro.orders`
a  left join
`Target_SQL_Intro.order_items` b on a.order_id=b.order_id
left join `Target_SQL_Intro.customers` c on a.customer_id=c.customer_id)
group by customer_state;
```



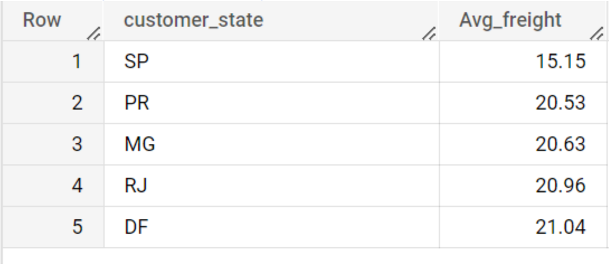
Row	customer_state	Avg_freight	avg_time_to_delivery	avg_diff_estimated_delivery
1	RJ	20.96	14.69	11.14
2	RS	21.74	14.71	13.2
3	SP	15.15	8.26	10.27
4	DF	21.04	12.5	11.27
5	PR	20.53	11.48	12.53
6	MT	28.17	17.51	13.64
7	MA	38.26	21.2	9.11
8	AL	35.84	23.99	7.98
9	MG	20.63	11.52	12.4
10	PE	32.92	17.79	12.55

Insight- States with higher average freight charges took more time to deliver the order whereas the average difference between estimated time and actual delivery ranged between 8 to 17 days.
Recommendation- Delivery time should not depend on Freight charges. There should be a standard delivery time for each state irrespective of the freight chargers

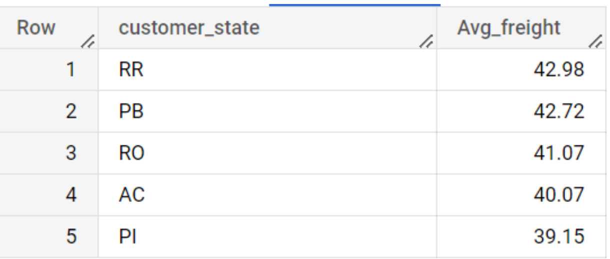
5.4 Sort the data to get the following:

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

```
select customer_state, round(avg(freight_value),2) as Avg_freight from
(select * from `Target_SQL_Intro.orders` a  left join
`Target_SQL_Intro.order_items` b on a.order_id=b.order_id
left join `Target_SQL_Intro.customers` c on a.customer_id=c.customer_id)
group by 1 order by 2 limit 5
```



Row	customer_state	Avg_freight
1	SP	15.15
2	PR	20.53
3	MG	20.63
4	RJ	20.96
5	DF	21.04



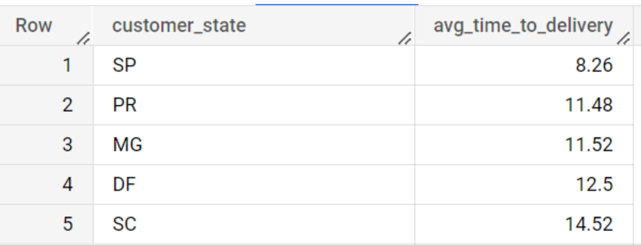
Row	customer_state	Avg_freight
1	RR	42.98
2	PB	42.72
3	RO	41.07
4	AC	40.07
5	PI	39.15

Insight- Minimum avg freight is for State SP-15.15 and Maximum is for RR-42.98.

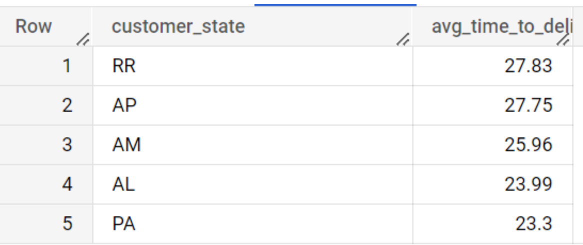
5.6 Top 5 states with highest/lowest average time to delivery

```
select customer_state,

round(avg(timestamp_diff(order_delivered_customer_date,order_purchase_timestamp,day)),
2) as avg_time_to_delivery,
from (select * from `Target_SQL_Intro.orders` a  left join
`Target_SQL_Intro.order_items` b on a.order_id=b.order_id
left join `Target_SQL_Intro.customers` c on a.customer_id=c.customer_id)
group by 1 order by 2 limit 5;
```



Row	customer_state	avg_time_to_delivery
1	SP	8.26
2	PR	11.48
3	MG	11.52
4	DF	12.5
5	SC	14.52

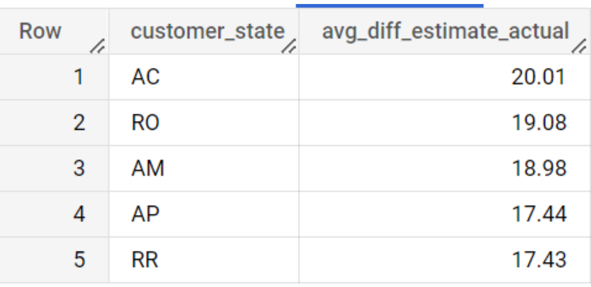


Row	customer_state	avg_time_to_deli
1	RR	27.83
2	AP	27.75
3	AM	25.96
4	AL	23.99
5	PA	23.3

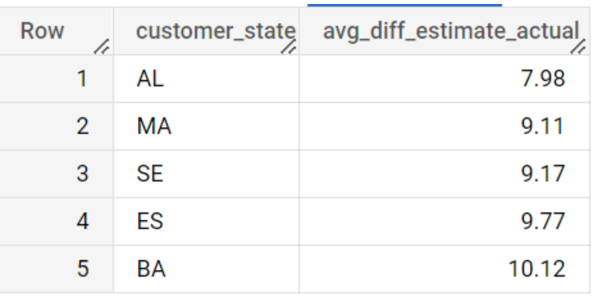
Insight- State SP with most customer takes least average delivery time(8.26) and state RR has the highest(27.83)
Recommendation- Logistics to be improved in states which takes higher delivery time to provide good consumer experience which would help in word of mouth marketing.

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

```
select customer_state,
round(avg(timestamp_diff(order_estimated_delivery_date,order_delivered_customer_date,d
ay)),2) as avg_diff_estimate_actual
from (select * from `Target_SQL_Intro.orders` a  left join
`Target_SQL_Intro.order_items` b on a.order_id=b.order_id
left join `Target_SQL_Intro.customers` c on a.customer_id=c.customer_id)
group by 1 order by 2 desc limit 5;
```



Row	customer_state	avg_diff_estimate_actual
1	AC	20.01
2	RO	19.08
3	AM	18.98
4	AP	17.44
5	RR	17.43



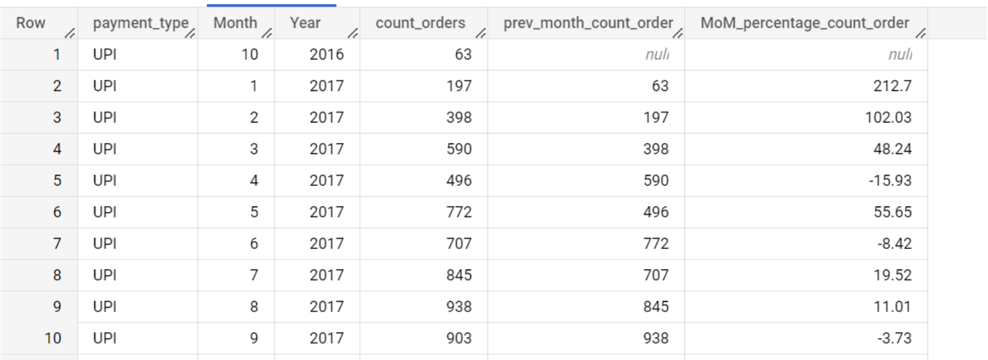
Row	customer_state	avg_diff_estimate_actual
1	AL	7.98
2	MA	9.11
3	SE	9.17
4	ES	9.77
5	BA	10.12

Insight- In state AC, the estimated delivery time is well over the actual delivery time whereas in State AL the estimate delivery date is much closer to the actual delivery time.
Recommendation- Estimated delivery details should be accurate for for states such as AC, RO, AM etc.

6. Payment type analysis:

6.1 Month over Month count of orders for different payment types

```
select *, round(100*(count_orders-prev_month_count_order)/prev_month_count_order,2) as
MoM_percentage_count_order from
(select payment_type,Month,Year,count(distinct(ID)) as count_orders,
lag(count(distinct(ID)),1) over(partition by payment_type order by Year,Month) as
prev_month_count_order from
(select b.order_id as ID, b.payment_type as payment_type, extract(year from
order_purchase_timestamp) as Year, extract(month from order_purchase_timestamp) as
Month  from `Target_SQL_Intro.orders` a left join `Target_SQL_Intro.payments` b on
a.order_id=b.order_id)
group by payment_type,Year, Month
having payment_type is not null order by payment_type)
```

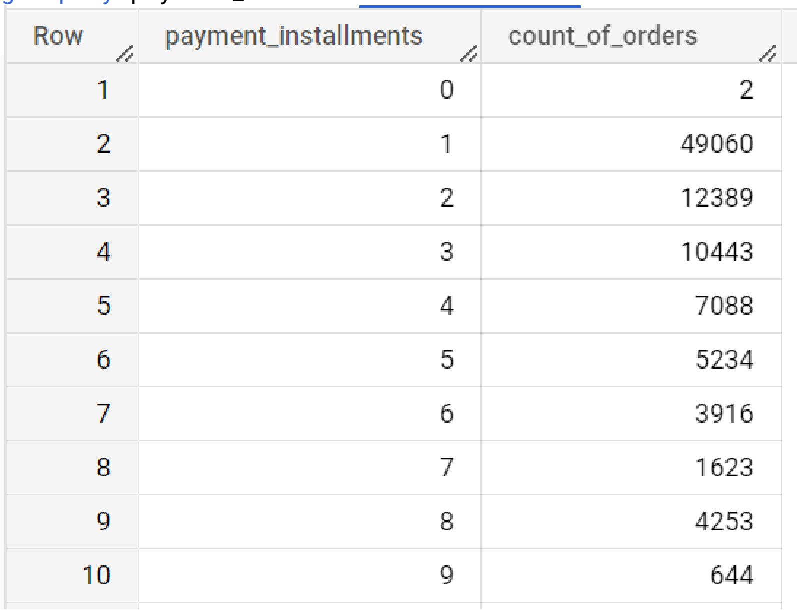


Row	payment_type	Month	Year	count_orders	prev_month_count_order	MoM_percentage_count_order
1	UPI	10	2016	63	null	null
2	UPI	1	2017	197	63	212.7
3	UPI	2	2017	398	197	102.03
4	UPI	3	2017	590	398	48.24
5	UPI	4	2017	496	590	-15.93
6	UPI	5	2017	772	496	55.65
7	UPI	6	2017	707	772	-8.42
8	UPI	7	2017	845	707	19.52
9	UPI	8	2017	938	845	11.01
10	UPI	9	2017	903	938	-3.73

Insight- The MoM count of order for different payment method increased initially in 2016 and 2017 but reduced considerably towards the end of 2018 except for Debit cards.
Recommendation- Tie up banks etc to provide special offers to increase Credit card payments as they are most preferred mode of payment and started dipping considerably towards the end of the year 2018.

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

```
select payment_installments, count(distinct(order_id)) as number_of_orders from
`Target_SQL_Intro.payments`
group by payment_installments
```



Row	payment_installments	count_of_orders
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

Insight- Almost 50% of order came through single installment. Most of multiple installment orders where done with credit card and few with vouchers. Also there are 18 orders which had an installment period of 24 months.

FINAL INSIGHTS

- There is huge growth of 137% in total sales in year 2018 compared to sales in 2017. This is almost 2.4x growth year over year.
- The peak sales appeared during winter(between March and October) and reduced during summer(March to November)
- State wise there was increase in total order month on month but it reduced drastically during December 2017, August 2018 and June 2017.
- There is clear trend in increase of average price with the increase in freight. The range for average price is between 192 to 109 and for average freight charge is 42 to 15.
- States with higher average freight charges took more time to deliver the order.
- Almost 50% of total customers are from a single state SP. More than 50% of states have customer less than 1000.
- There were deliveries which were delayed well over the estimated time. Also there was huge difference between actual and estimated delivery dates which doesn't provide any accurate delivery details.
- Credit card is the most preferred payment mode and then UPI.

RECOMMENDATION

- Provide special one time deals to passive/inactive customers to increase their engagement level.
- Least orders were placed during Dawn. To attract more traffic during this time, limited time offers can be provided to the customers in this time slot.
- Target Tier 2 and Tier 3 cities where total customers are low. Understand their requirement and buying patterns and expand product basket by including the products relevant to them.
- Controlling the variance of Freight charge can bring down the average price of order
- Proper inventory planning, Manpower planning, and logistics should be done well before the peak sales period. Logistics has to be streamlined to provide accurate estimated delivery date.
- Tie up banks etc to provide special offers to increase Credit card payments as they are most preferred mode of payment and started dipping considerably towards the end of the year 2018.