Question 1

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

1. Data type of columns in a table

Query used:

```
SELECT column_name, data_type
FROM `Customer.INFORMATION_SCHEMA.COLUMNS`
WHERE table_name = "Customer_info";
```

Sample Results:

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

Comments:

We have imported the all the dataset to bigquery and we checked the structure of all the dataset and provide example of one snippet with one query

Question 1

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

2. Time period for which the data is given

Query used:

```
SELECT MIN(extract(YEAR FROM order_purchase_timestamp)) as Start_year_data,MAX(extract(YEAR FROM order_purchase_timestamp)) as End_year_data FROM `first-business-case-study.Customer.orders`;
```

Sample Results:

Row	Start_year_data	End_year_data
1	2016	2018

Question 1

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

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

Query used:

```
select distinct customer.customer_city,customer.customer_state
from `first-business-case-study.Customer.orders` orders
join `first-business-case-study.Customer.Customer_info` customer on orders.customer_id=customer.customer_id
order by customer.customer_city,customer.customer_state;
```

Sample Results:

Row	customer_city	customer_state
1	abadia dos dourados	MG
2	abadiania	GO
3	abaete	MG
4	abaetetuba	PA
5	abaiara	CE
6	abaira	BA
7	abare	BA

Question 2

2. In-depth Exploration:
    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 used:

```
SELECT count(orders.order_id) as cnt_order,
extract(YEAR from orders.order_purchase_timestamp) as order_year,
extract(MONTH from orders.order_purchase_timestamp) as order_month
FROM `first-business-case-study.Customer.orders` orders
LEFT JOIN `first-business-case-study.Customer.Customer_info` customer
ON orders.customer_id = customer.customer_id
group by order_year,order_month;
```

Sample Results:

Row	cnt_order	order_year	order_month
1	4	2016	9
2	324	2016	10
3	1	2016	12
4	800	2017	1
5	1780	2017	2
6	2682	2017	3
7	2404	2017	4

Comments:

Data are exponentially increasing and decreasing there is no seasonality in brazil

Question 2

2. In-depth Exploration: (Approach 1)
    2. What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)?

Query used:

```
select extract(year from date(order_time.order_purchase_timestamp)) as year,
extract(month from date(order_time.order_purchase_timestamp)) as month,
order_time.timeslot, count(*) as order_count
from
(select *,
case
  when time(orders.order_purchase_timestamp) between '00:00:00' and '05:59:59'
  then 'Dawn'
  when time(orders.order_purchase_timestamp) between '06:00:00' and '11:59:59'
  then 'Morning'
  when time(orders.order_purchase_timestamp) between '12:00:00' and '17:59:59'
  then 'Afternoon'
 when time(orders.order_purchase_timestamp) between '18:00:00' and '23:59:59'
  then 'Night'
end as timeslot
from `first-business-case-study.Customer.orders` orders
) order_time
group by year, month, order_time.timeslot
order by year, month, order_time.timeslot
```

Comments:

Mostly customer are ordering product on afternoon time

Row Labels	Afternoon	Dawn	Morning	Night	Grand Total
Grand Total	38.58%	4.77%	22.37%	34.29%	100.00%

Sample Results:

Row	year	month	timeslot	order_count
1	2016	9	Afternoon	2
2	2016	9	Dawn	1
3	2016	9	Night	1
4	2016	10	Afternoon	125
5	2016	10	Dawn	15
6	2016	10	Morning	84
7	2016	10	Night	100

Question 3

3. Evolution of E-commerce orders in the Brazil region:
    1. Get month on month orders by states

Query used:

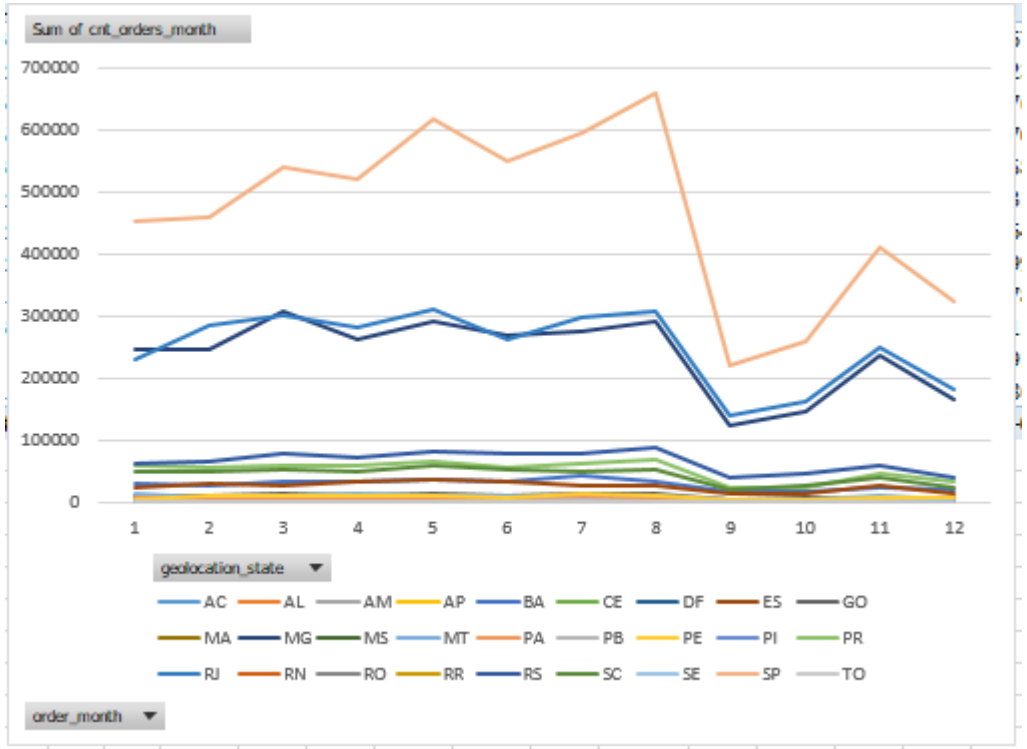
```
SELECT extract(MONTH from orders.order_purchase_timestamp) as order_month, count(*) as cnt_orders_month, geolocation.geolocation_state FROM `first-business-case-study.Customer.orders` orders
INNER JOIN `first-business-case-study.Customer.Customer_info` cust
ON orders.customer_id = cust.customer_id
INNER JOIN `first-business-case-study.Customer.Geolocation` geolocation
ON cust.customer_zip_code_prefix = geolocation.geolocation_zip_code_prefix
group by order_month,geolocation.geolocation_state
order by order_month;
```

Sample Result :

Row	order_month	cnt_orders_mon	geolocation_state
1	1	2634	RN
2	1	453515	SP
3	1	246203	MG
4	1	32144	BA
5	1	230923	RJ
6	1	62867	RS
7	1	4008	MA

Comments:

Measurely three State having max orders in brazil



Question 3

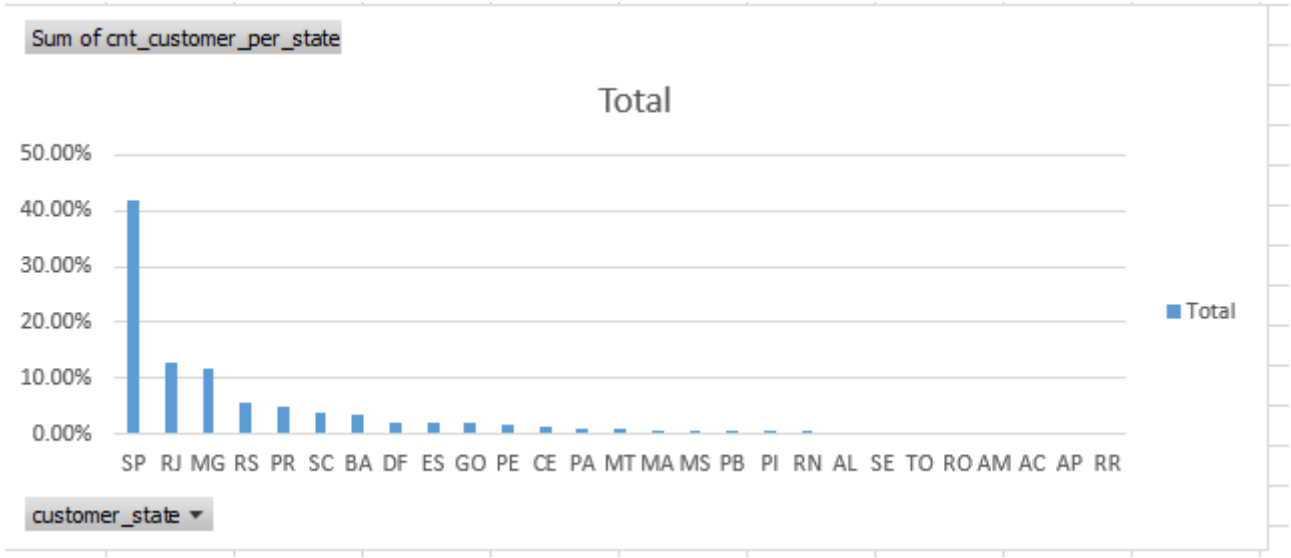
3. Evolution of E-commerce orders in the Brazil region:
    2. Distribution of customers across the states in Brazil

Query used:

SELECT customer_state, ROUND((count(*)/(SELECT count(*)
FROM `first-business-case-study.Customer.Customer_info`))*100,2) as cnt_customer_per_state FROM `first-business-case-study.Customer.Customer_info` group by customer_state;

Sample Results:

Row	customer_state	cnt_customer_p
1	RN	0.49
2	CE	1.34
3	RS	5.5
4	SC	3.66
5	SP	41.98
6	MG	11.7
7	BA	3.4



Question 4

4. Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.
    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 used: (to obtain cost for each year)

```
WITH payment_2017 as
(SELECT extract(YEAR from orders.order_purchase_timestamp) as orders_year, extract(MONTH from orders.order_purchase_timestamp) as orders_month, ROUND(SUM(payment.payment_value),2) as payme
nt_amount FROM `first-business-case-study.Customer.payments` payment
LEFT JOIN `first-business-case-study.Customer.orders` orders
ON payment.order_id = orders.order_id
where extract(MONTH from orders.order_purchase_timestamp) between 1 and 8 AND extract(YEAR from orders.order_purchase_timestamp)= 2017
group by orders_year, orders_month
order by orders_year, orders_month),
payment_2018 as
(SELECT extract(YEAR from orders.order_purchase_timestamp) as orders_year_next, extract(MONTH from orders.order_purchase_timestamp) as orders_month_next, ROUND(SUM(payment.payment_value),2
) as payment_amount_next FROM `first-business-case-study.Customer.payments` payment
LEFT JOIN `first-business-case-study.Customer.orders` orders
ON payment.order_id = orders.order_id
where extract(MONTH from orders.order_purchase_timestamp) between 1 and 8 AND extract(YEAR from orders.order_purchase_timestamp)= 2018
group by orders_year_next, orders_month_next
order by orders_year_next, orders_month_next)
select *,
ROUND(((payment_2018.payment_amount_next - payment_2017.payment_amount)/payment_2017.payment_amount)*100,2) as percentage_increase from payment_2017
INNER JOIN payment_2018
ON payment_2017.orders_month = payment_2018.orders_month_next
order by payment_2017.orders_month;
```

Sample Results:

Row	orders_year	orders_month	payment_amour	orders_year_nex	orders_month_p	payment_amour	percentage_incr
1	2017	1	138488.04	2018	1	1115004.18	705.13
2	2017	2	291908.01	2018	2	992463.34	239.99
3	2017	3	449863.6	2018	3	1159652.12	157.78
4	2017	4	417788.03	2018	4	1160785.48	177.84
5	2017	5	592918.82	2018	5	1153982.15	94.63
6	2017	6	511276.38	2018	6	1023880.5	100.26
7	2017	7	592382.92	2018	7	1066540.75	80.04
8	2017	8	674396.32	2018	8	1022425.32	51.61

Question 4

4. Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.
    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 used: (to obtain cost for each year)

```
select round((cost_2018-cost_2017)*100/cost_2017,2) as pc_inc_in_cost
from
(select extract(year from date(o.order_purchase_timestamp)) as year,
round(sum(p.payment_value),2) as cost_2017
from `first-business-case-study.Customer.orders` o
join `first-business-case-study.Customer.payments` p on p.order_id=o.order_id
where extract(month from date(o.order_purchase_timestamp)) between 1 and 8
group by year
having year=2017)t1,

(select extract(year from date(o.order_purchase_timestamp)) as year,
round(sum(p.payment_value),2) as cost_2018
from `first-business-case-study.Customer..orders` o
join `first-business-case-study.Customer.payments` p on p.order_id=o.order_id
where extract(month from date(o.order_purchase_timestamp)) between 1 and 8
group by year
having year=2018)t2
```

Sample Results:

Row	pc_inc_in_cost
1	136.98

Comments:

These problem can be solve by two ways :
1. Calculating percentage increase from month of year (2017,2018). It will give the picture of month by month increase percentage from 2017 to 2018
2. Calculate the Percentage increase from 2017 to 2018 on the basis of total amount

Question 4

4. Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.
    2. Mean & Sum of price and freight value by customer state

Query used:

```
SELECT customer.customer_state, ROUND(avg(order_item.price),2) as mean_price, ROUND(sum(order_item.price),2) as total_pric
e,
ROUND(avg(order_item.freight_value),2) as mean_freight_value, ROUND(sum(order_item.freight_value),2) as total_freight_value
FROM `first-business-case-study.Customer.order_items` order_item
INNER JOIN `first-business-case-study.Customer.orders` orders
ON order_item.order_id = orders.order_id
INNER JOIN `first-business-case-study.Customer.Customer_info` customer
ON orders.customer_id = customer.customer_id
group by customer.customer_state
order by customer.customer_state;
```

Sample Results:

Row	customer_state	mean_price	total_price	mean_freight_va	total_freight_val
1	AC	173.73	15982.95	40.07	3686.75
2	AL	180.89	80314.81	35.84	15914.59
3	AM	135.5	22356.84	33.21	5478.89
4	AP	164.32	13474.3	34.01	2788.5
5	BA	134.6	511349.99	26.36	100156.68
6	CE	153.76	227254.71	32.71	48351.59
7	DF	125.77	302603.94	21.04	50625.5

Question 5

5. Analysis on sales, freight and delivery time
    1. Calculate days between purchasing, delivering and estimated delivery

Query used:

```
SELECT order_id,
DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day) as purchase_delivery_day_diff,
DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date, day) as estimated_delivery_day_diff,
DATE_DIFF(order_estimated_delivery_date, order_purchase_timestamp, day) as estimated_purchase_day_diff
FROM `first-business-case-study.Customer.orders`
where order_status='delivered';
```

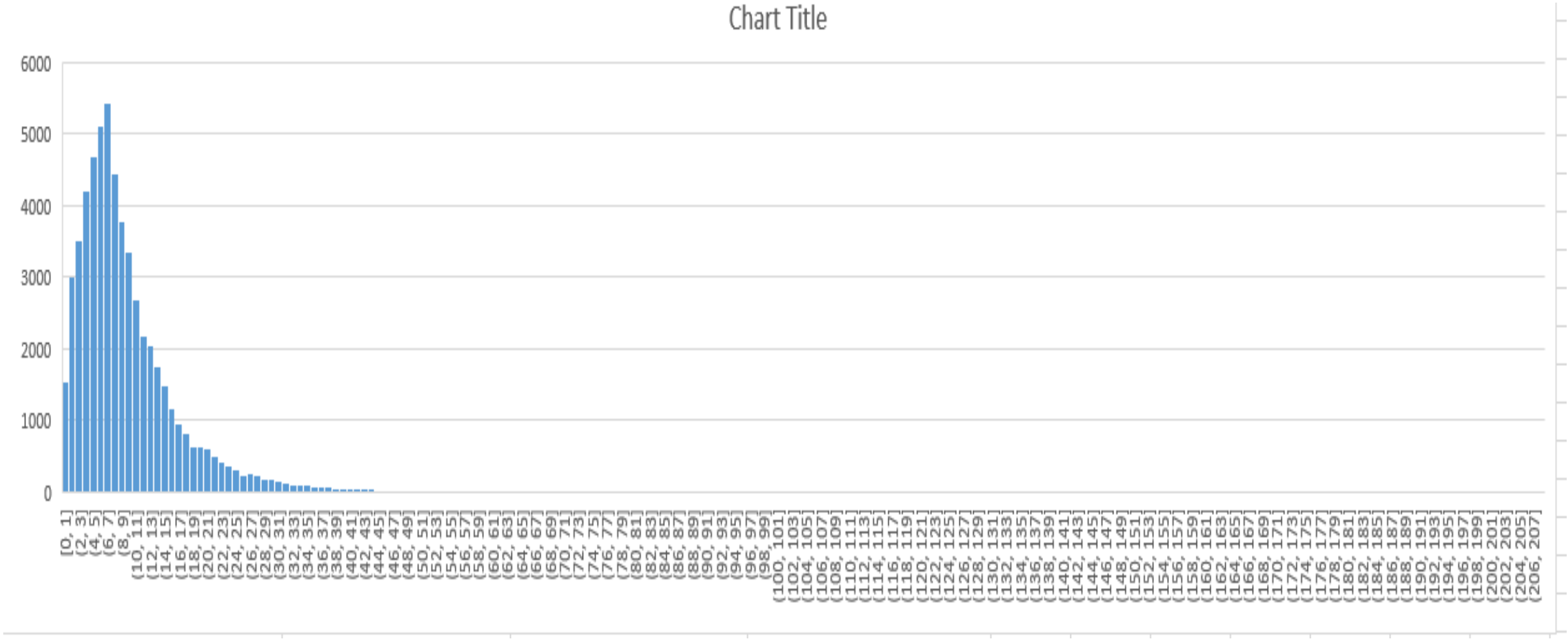
Sample Results:

Row	order_id	purchase_delive	estimated_deliv	estimated_purch
1	635c894d068ac37e6e03dc54e...	30	1	32
2	3b97562c3aee8bdedcb5c2e45...	32	0	33
3	68f47f50f04c4cb6774570cfde...	29	1	31
4	276e9ec344d3bf029ff83a161c...	43	-4	39
5	54e1a3c2b97fb0809da548a59...	40	-4	36
6	fd04fa4105ee8045f6a0139ca5...	37	-1	35
7	302bb8109d097a9fc6e9cefc5...	33	-5	28

Question 5

5. Analysis on sales, freight and delivery time
   1.Calculate days between purchasing, delivering and estimated delivery

Graph:



Question 5

5. Analysis on sales, freight and delivery time
    2. Find time_to_delivery & diff_estimated_delivery. Formula for the same given below:
        1. time_to_delivery = order_purchase_timestamp - order_delivered_customer_date
        2. diff_estimated_delivery = order_estimated_delivery_date - order_delivered_customer_date

```
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 `first-business-case-study.Customer.orders`
where (order_delivered_customer_date - order_purchase_timestamp) is not null or (order_estimated_delivery_date -
 order_delivered_customer_date) is not null;
```

Sample results :

Row	time_to_delivery	diff_estimated_c
1	30	-12
2	30	28
3	35	16
4	30	1
5	32	0
6	29	1
7	43	-4

Comments:

By analyzing the data is shows that time to delivery is higher than estimate delivery date

Question 5

5. Analysis on sales, freight and delivery time
    3. Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery

```
SELECT customer.customer_state,
ROUND(AVG(order_items.freight_value),2) as mean_freight_value,
ROUND(AVG(DATE_DIFF(orders.order_delivered_customer_date, orders.order_purchase_timestamp,day)),
2) as time_to_delivery,
ROUND(AVG(DATE_DIFF(orders.order_estimated_delivery_date, orders.order_delivered_customer_date, d
ay)),2) as diff_estimated_delivery
FROM `first-business-case-study.Customer.orders` orders
LEFT JOIN `first-business-case-study.Customer.order_items` order_items
ON orders.order_id = order_items.order_id
LEFT JOIN `first-business-case-study.Customer.Customer_info` customer
ON orders.customer_id = customer.customer_id
group by customer.customer_state
order by customer.customer_state;
```

Comments:

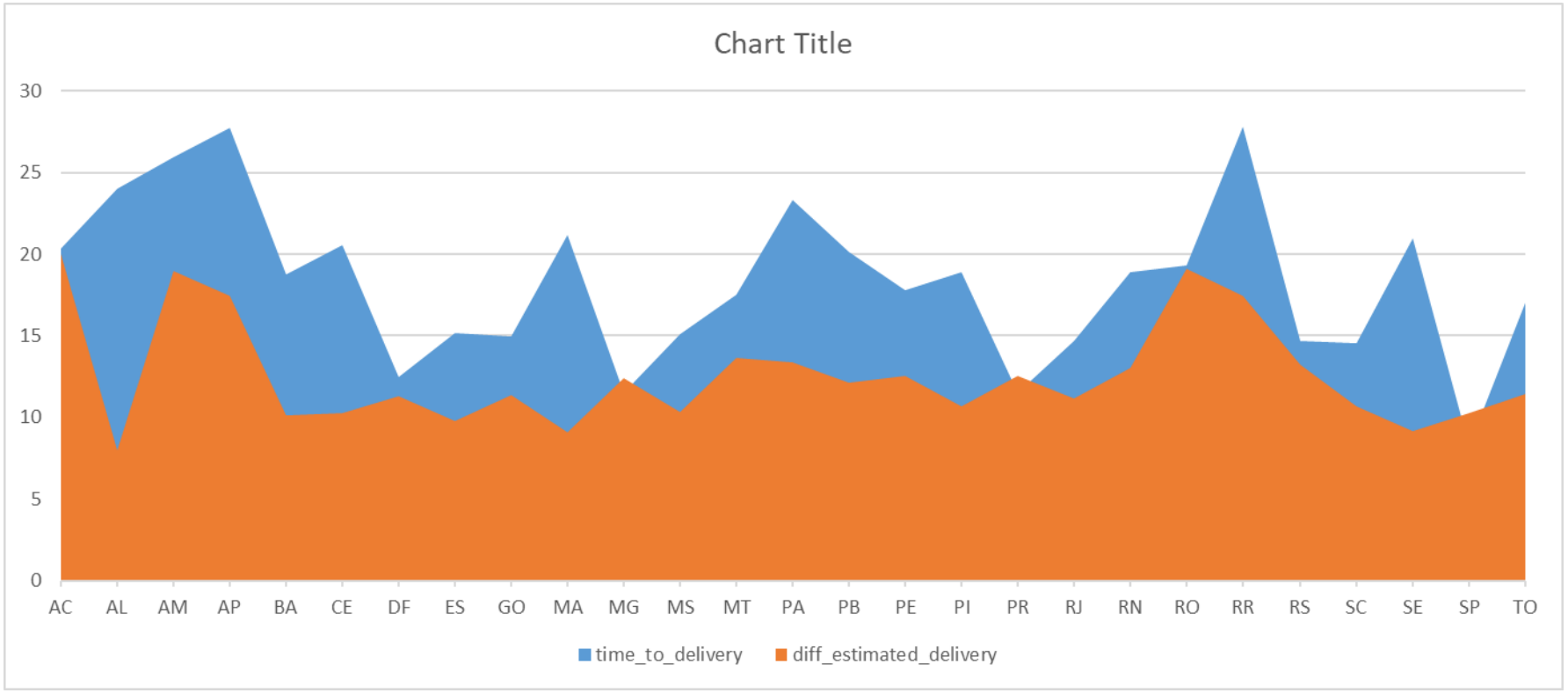
Comparing the data by state is shows that time to delivery is higher than estimate delivery date

Sample results :

Row	customer_state	mean_freight_va	time_to_delivery	diff_estimated_c
1	AC	40.07	20.33	20.01
2	AL	35.84	23.99	7.98
3	AM	33.21	25.96	18.98
4	AP	34.01	27.75	17.44
5	BA	26.36	18.77	10.12
6	CE	32.71	20.54	10.26
7	DF	21.04	12.5	11.27

Question 5

5. Analysis on sales, freight and delivery time
    3. Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery



Question 5

5. Analysis on sales, freight and delivery time
    5. Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

```
WITH State_freight_value as,
(SELECT customer.customer_state,
ROUND(AVG(order_items.freight_value),2) as mean_freight_value,
ROUND(AVG(DATE_DIFF(orders.order_delivered_customer_date, orders.order_purchase_timestamp,day)),2) as time_to_delivery,
ROUND(AVG(DATE_DIFF(orders.order_estimated_delivery_date, orders.order_delivered_customer_date, day)),2) as diff_estimated_delivery,
dense_rank() over(order by ROUND(AVG(order_items.freight_value),2) desc) as max_freight_value
dense_rank() over(order by ROUND(AVG(order_items.freight_value),2) asc) as min_freight_value
FROM `first-business-case-study.Customer.orders` orders
LEFT JOIN `first-business-case-study.Customer.order_items` order_items
ON orders.order_id = order_items.order_id
LEFT JOIN `first-business-case-study.Customer.Customer_info` customer
ON orders.customer_id = customer.customer_id
group by customer.customer_state),
max_freight as
(SELECT customer_state,mean_freight_value,max_freight_value FROM State_freight_value
WHERE max_freight_value<=5),
min_freight as
(SELECT customer_state,mean_freight_value,min_freight_value FROM State_freight_value
WHERE min_freight_value<=5)
SELECT max_freight.customer_state as top_five_state,max_freight.mean_freight_value as top_five_freight_value,
min_freight.customer_state as lower_five_state,min_freight.mean_freight_value as lower_five_freight_value
FROM max_freight
INNER JOIN min_freight
ON max_freight.max_freight_value = min_freight.min_freight_value
order by max_freight.max_freight_value;
```

Question 5

5. Analysis on sales, freight and delivery time
    5. Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

Sample results :

Row	top_five_state	top_five_freight	lower_five_state	lower_five_freigl
1	RR	42.98	SP	15.15
2	PB	42.72	PR	20.53
3	RO	41.07	MG	20.63
4	AC	40.07	RJ	20.96
5	PI	39.15	DF	21.04

Question 5

5. Analysis on sales, freight and delivery time
   7. Top 5 states where delivery is really fast/ not so fast compared to estimated date

```
WITH fast_delivery as
(SELECT distinct customer.customer_state,
TIMESTAMP_DIFF(orders.order_delivered_customer_date, orders.order_purchase_timestamp,HOUR) as time_to_del
ivery,
dense_rank() over(partition by customer.customer_state order by TIMESTAMP_DIFF(orders.order_delivered_custome
r_date, orders.order_purchase_timestamp,HOUR)) as d_rnk
FROM `first-business-case-study.Customer.orders` orders
LEFT JOIN `first-business-case-study.Customer.order_items` order_items
ON orders.order_id = order_items.order_id
LEFT JOIN `first-business-case-study.Customer.Customer_info` customer
ON orders.customer_id = customer.customer_id
where TIMESTAMP_DIFF(orders.order_estimated_delivery_date, orders.order_delivered_customer_date, HOUR)>0)
SELECT customer_state, time_to_delivery FROM fast_delivery where d_rnk=1 order by time_to_delivery limit 5;
```

Sample results :

Row	customer_state	time_to_delivery
1	RJ	12
2	SP	18
3	BA	20
4	RS	25
5	MG	25

Question 6

6 Payment type analysis:

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

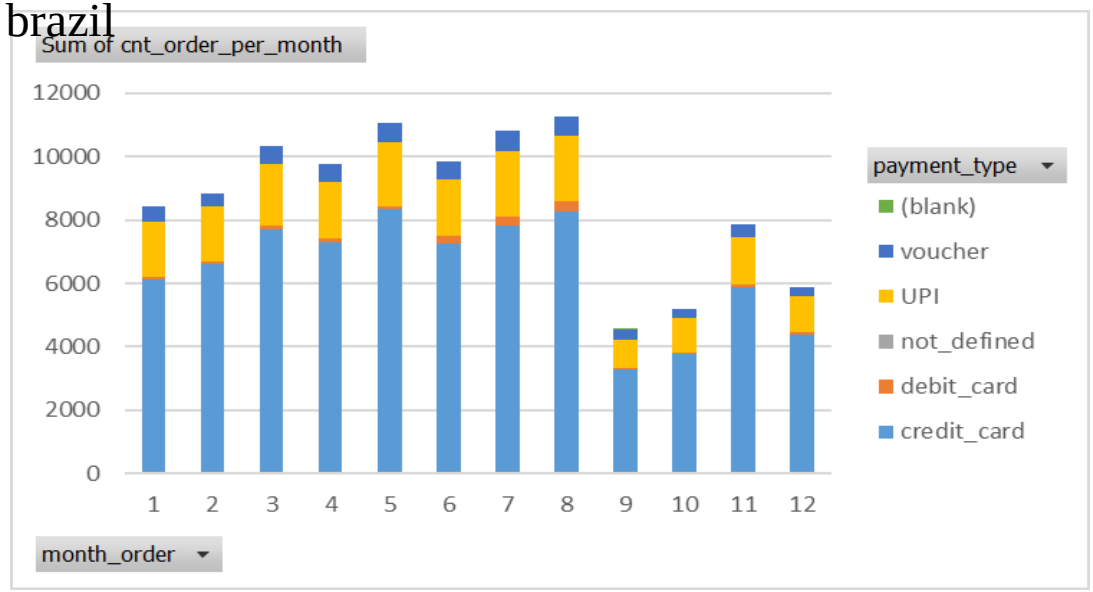
```
SELECT count(orders.order_id) as cnt_order_per_month, EXTRACT(MONTH from orders.order_purchase_timestamp) as mo
nth_order, payments.payment_type
FROM `first-business-case-study.Customer.orders` orders
LEFT JOIN `first-business-case-study.Customer.payments` payments
ON orders.order_id = payments.order_id
group by EXTRACT(MONTH from orders.order_purchase_timestamp),payments.payment_type;
```

Sample results :

Row	cnt_order_per_m	month_order	payment_type
1	1509	11	UPI
2	4378	12	credit_card
3	1723	2	UPI
4	5897	11	credit_card
5	572	4	voucher
6	7841	7	credit_card
7	2074	7	UPI

Comments:

It's looks like Credit card method is most frequent used and three of the payment mode is more usable in brazil



Question 6

6 Payment type analysis:

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

```
SELECT payment_installments, count(order_id) as cnt_order FROM `first-business-case-study.Customer.payments`
group by payment_installments
order by payment_installments;
```

Sample results :

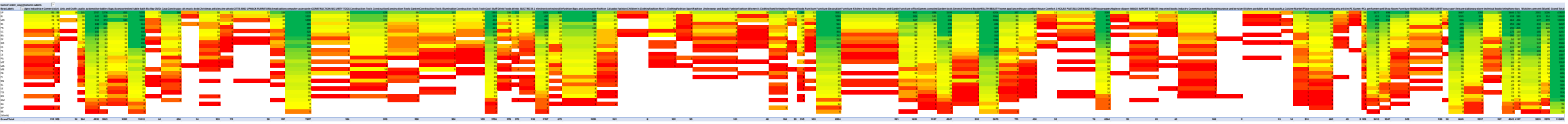
Row	payment_installr	cnt_order
1	0	2
2	1	52546
3	2	12413
4	3	10461
5	4	7098
6	5	5239
7	6	3920

Question 7 - Actionable Insights

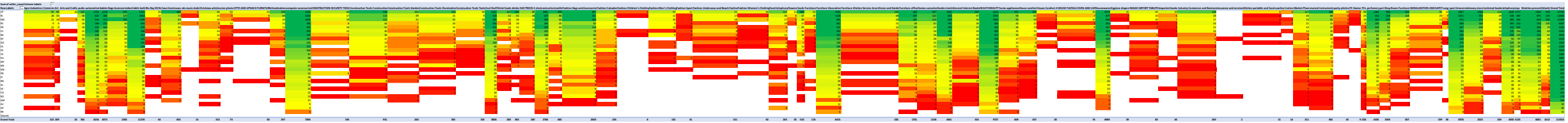
2. Need more seller in other states

By analyzing data it’s seems the there is correlation between count of orders and sellers with respect to product category and states in other word below charts suggested that the order count in certain states are low because the sellers them self are low

Count of orders



Count of sellers



Product categories

States

Question 7 - Actionable Insights

2. Need more seller in other states

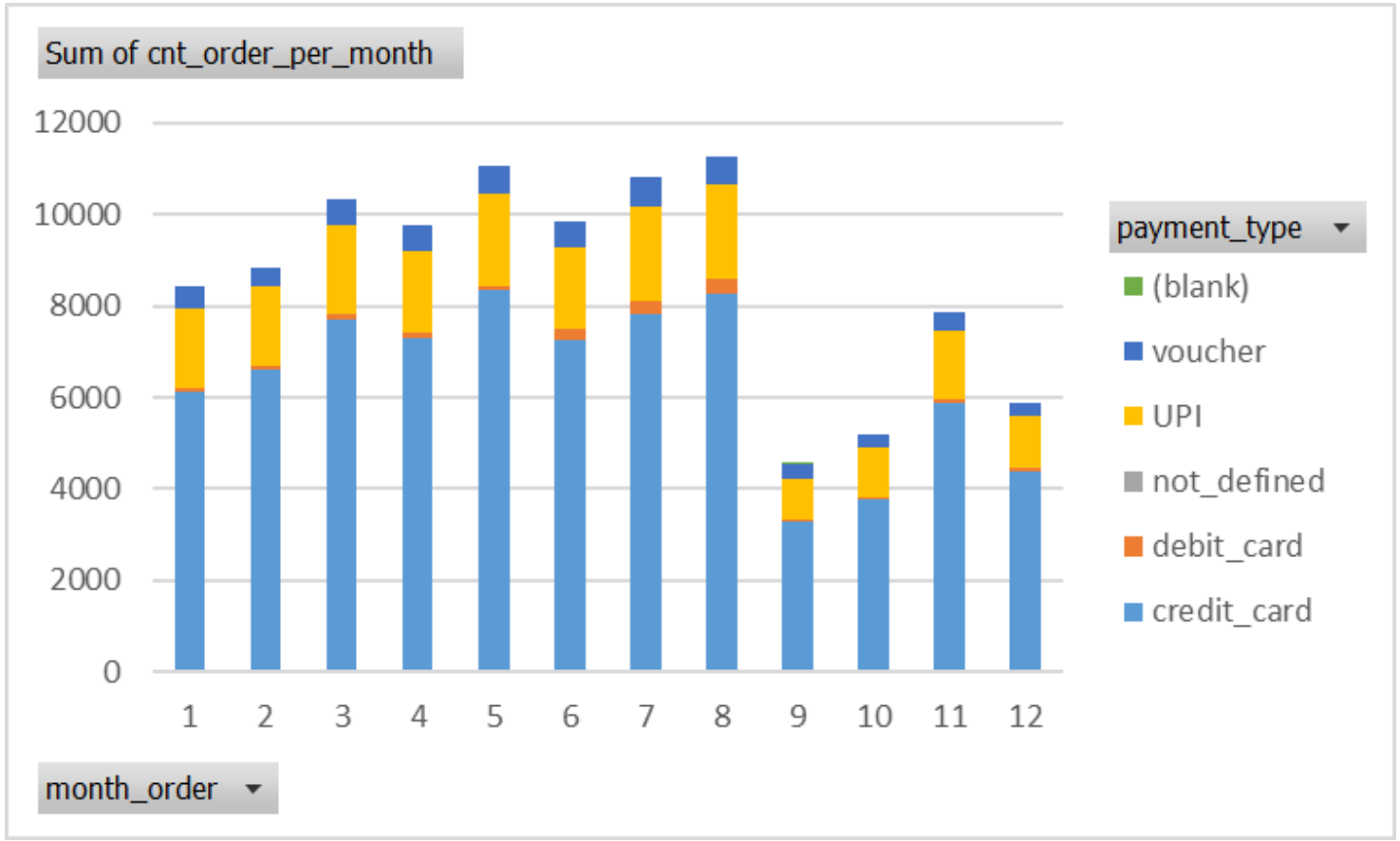
```
select c.customer_state, p.product_category, count(s.seller_id) as seller_count
from `first-business-case-study.Customer.orders` o
left join `first-business-case-study.Customer.order_items` oi on oi.order_id=o.order_id
left join `first-business-case-study.Customer.products` p on p.product_id=oi.product_id
left join `first-business-case-study.Customer.Customer_info` c on c.customer_id=o.customer_id
left join `first-business-case-study.Customer.order_reviews` o_r on o_r.order_id=o.order_id
left join `first-business-case-study.Customer.sellers` s on s.seller_id=oi.seller_id
group by c.customer_state, p.product_category
order by c.customer_state, p.product_category
```

```
select c.customer_state, p.product_category, count(o.order_id) as order_count
from `first-business-case-study.Customer.orders` o
left join `first-business-case-study.Customer.order_items` oi on oi.order_id=o.order_id
left join `first-business-case-study.Customer.products` p on p.product_id=oi.product_id
left join `first-business-case-study.Customer.Customer_info` c on c.customer_id=o.customer_id
group by c.customer_state, p.product_category
order by c.customer_state, p.product_category
```

Question 7 - Actionable Insights

3. Need to use more UPI

The chart below clearly shows that UPI option is being underused a lot specially when compared to the credit card.



Question 8 - Recommendation

Drawing from the insights pointed out in the Question 7 following are the recommendations:

1. Target should work on capturing potential customer base in other state as well
2. Target should work on increasing the seller footprint among the other states where the seller presence is very low
3. Encourage among the sellers the usage of UPI as payment method