



**NOTE:****Projectname:** sql-business-case-418004**Datasetname:** geolocation**Tables used:** customers, geolocation, order\_items, order\_reviews, orders, payments, products, sellers**Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset:****Q1) Data type of all columns in the "customers" table.****ANS:****QUERY:**

```
select column_name, data_type
from `sql-business-case-418004.geolocation.INFORMATION_SCHEMA.COLUMNS`
where table_name = 'customers'
```

Query results		 SAVE RESL	
 JOB INFORMATION		RESULTS	CHART
JSON			
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	

**INSIGHTS:** N/A**RECOMMENDATIONS:** N/A**Q2) Get the time range between which the orders were placed.****ANS:****QUERY**

```
SELECT min(order_purchase_timestamp) as min_, max(order_purchase_timestamp) as max_
from `geolocation.orders`
```

<	JOB INFORMATION	RESULTS	CHART	JSON	EXECU
Row	min_ ▼	max_ ▼			
1	2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC			

**INSIGHTS:** Data is available only from '2016-09-04' to '2018-10-17'. It's not complete 3 years data.

**RECOMMENDATION:** N/A

**Q3) Count the Cities & States of customers who ordered during the given period.**  
ANS:

**QUERY**

```
SELECT count(distinct geolocation_city) as city_count, count(distinct geolocation_state) as state_count from
`geolocation.geolocation`
```

Query results				SAV
<	JOB INFORMATION	RESULTS	CHART	
Row	city_count ▼	state_count ▼		
1	8011	27		

**INSIGHTS:** N/A


**RECOMMENDATION:** N/A

**In-depth Exploration:**

**Q1) Is there a growing trend in the no. of orders placed over the past years?**  
ANS:

**QUERY:**

```
select YofSale, count(*) as no_of_orders from
(
SELECT order_id, EXTRACT(YEAR FROM order_purchase_timestamp) as YofSale from `geolocation.orders`) as o
group by YofSale
order by YofSale
```

Query results				 SAVE RESULTS ▾
<	JOB INFORMATION	RESULTS	CHART	JSON
Row	YofSale ▾	no_of_orders ▾		
1	2016	329		
2	2017	45101		
3	2018	54011		

#### ORDERS THAT HAVE BEEN CANCELLED AMONG THEM:

	JOB INFORMATION	RESULTS	CHART
Row	YofSale ▾	no_of_orders ▾	
1	2016	26	
2	2017	265	
3	2018	334	

#### INSIGHTS:

According to the data provided, As per my observation, there is a visible growth in the number of sales from 2016 to 2018.

But the data in year 2016 is only available from '2016-09-04', so we are able to see very low orders in that year. If entire year data is available, then there might be variation in number of orders.

In year 2018, even though data is available till '2018-10-17', orders are higher than year 2017.

**RECOMMENDATION:** Even though data for 2016 is less, as November and December we have festival season which is a good chance to increase our sales. If we should have made more advertisements and gave good discounts on products. Then there will be an increase in the number of sales.

**Q2) Can we see some kind of monthly seasonality in terms of the no. of orders being placed?**

ANS:

QUERY:

```
SELECT EXTRACT(MONTH FROM order_purchase_timestamp) as MofSale, count(order_id) as
no_of_orders
from `geolocation.orders`
where order_status <> 'canceled'
group by MofSale
order by MofSale
```

Row	MofSale	no_of_orders
1	1	8032
2	2	8418
3	3	9834
4	4	9310
5	5	10520
6	6	9378
7	7	10249
8	8	10732
9	9	4268
10	10	4905
11	11	7507
12	12	5663

### INSIGHTS:

From the data we can see the number of sales are peaked during month of May, July and August. We can see from march to august the number of sales are more compared to other months due to summer and schools reopening. In the month of November we can see increase in sales, Might be because of thanksgiving festival. Jan and Feb also have high sales as new year and carnival festival happens.

### RECOMMENDATIONS:

We are facing more challenges during the period of September and October, we can increase our sales by giving discounts and promoting. And we can make a combo offer which includes high demand product with low sales product.

**Q3) 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**

ANS:

QUERY:

```
select order_time, count(*) as no_of_orders
from
(
select *, case
  when hours < 6 then 'Dawn'
  when hours > 6 and hours < 12 then 'Mornings'
  when hours > 12 and hours < 18 then 'Afternoon'
  else 'night'
end as order_time
from
(
```

```

select order_purchase_timestamp,extract(hour from order_purchase_timestamp) as
hours
from `geolocation.orders`
) o
)
group by order_time

```

Press Alt+F1 for

Query results [SAVE RESULTS](#) [EXPLORE](#)

<	JOB INFORMATION	RESULTS	CHART	JSON	EXECUTION DETAIL
Row	order_time	no_of_orders			
1	Mornings	21738			
2	Dawn	4740			
3	Afternoon	32366			
4	night	40597			

**INSIGHTS:** As most of the people sleep after 12PM we can clearly see the number of orders at Dawn are low. We can see that most orders are placed during the night time when everyone comes to home and ordering items as per their requirements.

**RECOMMENDATIONS:** As most of the orders are not being placed at dawn time. We can scale down our servers to decrease our expenses on servers.

## Evolution of E-commerce orders in the Brazil region:

### Q1) Get the month on month no. of orders placed in each state.

ANS:

QUERY:

```

select MofSale,YofSale, geolocation_state, count(*) as no_of_sale
from
(
select g.geolocation_state,extract(month from o.order_purchase_timestamp) as
MofSale,extract(year from o.order_purchase_timestamp) as YofSale
from `geolocation.orders` as o
left join `geolocation.customers` as c
on o.customer_id = c.customer_id
left join `geolocation.geolocation` as g
on c.customer_zip_code_prefix = g.geolocation_zip_code_prefix
)
group by YofSale,MofSale, geolocation_state
order by YofSale,MofSale, geolocation_state

```

## Query results

[SAVE RESULTS](#)
[EXPLORE D](#)

<	JOB INFORMATION	RESULTS	CHART	JSON	EXECUTION DETAILS
Row	MofSale	YofSale	geolocation_state	no_of_sale	
1	9	2016	RR	65	
2	9	2016	RS	103	
3	9	2016	SP	492	
4	10	2016	AL	52	
5	10	2016	BA	292	
6	10	2016	CE	477	
7	10	2016	DF	305	
8	10	2016	ES	271	
9	10	2016	GO	367	
10	10	2016	MA	353	
11	10	2016	MG	11756	
12	10	2016	MT	322	

### INSIGHTS:

As per my analysis from the data the most number of sales in a month are done in SP , this state made the top 10 highest sales of month in this period. AP and RR states are not having much sales in this time period.

### RECOMMENDATIONS:

We can improve the sales in the low performing states by increasing discounts and combo offers and advertising products. We have to focus on states with low orders and have to improve them.

### Q2) How are the customers distributed across all the states?

```
select geolocation_state, count(*) as no_of_cust
from
(
select c.geolocation_state
from `geolocation.geolocation` as c
left join `geolocation.customers` as g
on g.customer_zip_code_prefix = c.geolocation_zip_code_prefix
)
group by geolocation_state
order by no_of_cust desc
```

## Query results



< DB INFORMATION RESULTS CHART JSON

Row	geolocation_state ▼	no_of_cust ▼
1	SP	5620430
2	RJ	3015690
3	MG	2878728
4	RS	805370
5	PR	626021
6	SC	538638
7	BA	365875
8	ES	316654
9	GO	133146
10	MT	122395
11	PE	114588
12	DF	93309
13	PA	83554
14	CE	63507

### INSIGHTS:

As per my analysis from the data the most number of customers are in SP state, we are getting more order from there too. But we are having less customers from AC, AM, AP and RR states.

### RECOMMENDATIONS:

We can improve the sales in the low performing states by increasing discounts and combo offers and advertising products. We can make advertisements with celebrities to attract people and make local announcements as well.

## Impact on Economy:



**Q1) 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.**

ANS:

QUERY:

```
with cte as
(
select p.payment_value,extract(year from o.order_purchase_timestamp) as year,
extract(month from o.order_purchase_timestamp) as month
from `geolocation.payments` p
left join `geolocation.orders` o
on o.order_id = p.order_id
),
sample as (
select distinct year, sum(payment_value) over(partition by cte.year) as
cost_of_orders
from cte
where year in (2017,2018) and month between 1 and 8
group by year, payment_value
order by year
)
select *,case
when year = 2018 then ROUND(((cost_of_orders - (select cost_of_orders from sample
where year = 2017))/((select cost_of_orders from sample where year = 2017))*100,2)
else null
end as increase_percentage
from sample
```

Query results					 SAVE RESULTS ▾	 E
<	JOB INFORMATION	RESULTS	CHART	JSON	EXECUT	
Row	year ▾	cost_of_orders ▾	increase_percentage			
1	2017	2340992.03	null			
2	2018	4944467.35	111.21			

**INSIGHTS:** As per my findings, sales in year 2018 between January and August increased by 111% (i.e,number of orders are double) compared to year 2017 in the period between January and August.

Recommendation:



Number of orders has doubled in year 2018, we can increase sales by advertisements and discounts.

## Q2) Calculate the Total & Average value of order price for each state.

ANS:

QUERY:

```
with cte as (
SELECT c.customer_state,i.price
from `geolocation.customers` c
left join `geolocation.orders` o
on c.customer_id = o.customer_id
left join `geolocation.order_items` i
on o.order_id = i.order_id
)
select customer_state, avg(price) as avg_price, sum(price) as total_price
from cte
group by customer_state
order by avg_price asc, total_price
```

Row	customer_state ▼	avg_price ▼	total_price ▼
1	SP	109.6536291597...	5202955.050001...
2	PR	119.0041393728...	683083.7600000...
3	RS	120.3374530874...	750304.0200000...
4	MG	120.7485741488...	1585308.029999...
5	ES	121.9137012411...	275037.3099999...
6	SC	124.6535775862...	520553.3400000...
7	RJ	125.1178180945...	1824092.669999...
8	DF	125.7705486284...	302603.9399999...
9	GO	126.2717316759...	294591.9499999...
10	BA	134.6012082126...	511349.9900000...
11	AM	135.4959999999...	22356.84000000...
12	MS	142.6283760683...	116812.6399999...
13	MA	145.2041504854...	119648.2199999...
14	PE	145.5083222591...	262788.0299999...

Results per page: 50 ▼ 1 – 27 of 27 |< >

## INSIGHTS:

As per my Analysis,

**Set1** : PB, AL, AC, RO, PA are having very good average value of orders but when it comes to Total value made from orders those 5 are not in the top of list. These locations are not making more orders to stand in the top list of highest total value orders.

**Set2**: SP, RJ, MG, RS, PR, SC are the top total value making states.

The reason is because the number of orders from Set1 states might be less than Set2 states but they are purchasing products with high prices. When it comes to Set2 states, those states are purchasing products with high prices and low prices as well.

#### RECOMMENDATION:

As we have seen from our insights some people might not purchasing low price products and some people are purchasing them too. We can send survey forms to our customers, what products they are not able to find in our retails, what products they are finding not helpful. So we can take those feedback and improve our total sales.

### Q3) Calculate the Total & Average value of order freight for each state.

ANS:

QUERY:

```
with cte as (
SELECT c.customer_state,i.freight_value
from `geolocation.customers` c
left join `geolocation.orders` o
on c.customer_id = o.customer_id
left join `geolocation.order_items` i
on o.order_id = i.order_id
)
select customer_state, avg(freight_value) as avg_freight_value, sum(freight_value)
as total_freight_value
from cte
group by customer_state
order by total_freight_value desc
```

customer_state ▼	avg_freight_value ▼	total_freight_value ▼
SP	15.14727539041...	718723.0699999...
RJ	20.96092393168...	305589.3100000...
MG	20.63016680630...	270853.4600000...
RS	21.73580433039...	135522.7400000...
PR	20.53165156794...	117851.6800000...
BA	26.36395893656...	100156.6799999...
SC	21.47036877394...	89660.26000000...
PE	32.91786267995...	59449.65999999...
GO	22.76681525932...	53114.97999999...
DF	21.04135494596...	50625.49999999...
ES	22.05877659574...	49764.59999999...
CE	32.71420162381...	48351.58999999...
PA	35.83268518518...	38699.30000000...
MA	38.25700242718...	31523.77000000...

Results per page: 50 ▼ 1 – 27 of 27

### INSIGHTS:

As per my Analysis, we are getting more orders from SP so we are having very high freight cost for that state. But we are spending less amount on each product(avg\_freight\_cost) for shipping and delivering. But in case of RR,PB,RO,AC,PI and few more states, we are spending more money on each product as number of orders being placed are very less.

### RECOMMENDATIONS:

So in order to decrease the freight cost we have to increase the number of sales in the states having low orders. We can achieve this through advertisements and we can also use local warehouses to reduce the cost of transportation.

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

**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

ANS:

QUERY:

```
SELECT distinct order_id,o.order_status, g.geolocation_state,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) as
`delivery_time`,
DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date,DAY) as
`diff_estimated_delivery`
FROM `geolocation.orders` o
left join `geolocation.customers` c
on o.customer_id = c.customer_id
left join `geolocation.geolocation` g
on g.geolocation_zip_code_prefix = c.customer_zip_code_prefix
where o.order_status = 'delivered'
```

Row	der_id	order_status	geolocation_state	delivery_time	diff_estimated_delive
1	i5c894d068ac37e6e03dc54e...	delivered	RS	30	-1
2	i97562c3aee8bdedcb5c2e45...	delivered	MT	32	0
3	if47f50f04c4cb6774570cfde...	delivered	SE	29	-1
4	i6e9ec344d3bf029ff83a161c...	delivered	CE	43	4
5	ie1a3c2b97fb0809da548a59...	delivered	SC	40	4
6	04fa4105ee8045f6a0139ca5...	delivered	PE	37	1
7	i2bb8109d097a9fc6e9cefc5...	delivered	RJ	33	5
8	io57d37308e787052a32828...	delivered	AL	38	6
9	i135c945c554eebfd7576c73...	delivered	PA	36	2
10	i93e45e7ca1084efcd38ddeb...	delivered	MA	34	0
11	ic77e51e0f179d75a64a6141...	delivered	RS	42	11
12	i918e406132d7c81f1b84527...	delivered	PB	35	3
13	if6604e77ce6433e7d68dd86...	delivered	RJ	32	7

**INSIGHTS:** As per my Analysis, some deliveries getting delivered faster than expected time and some are getting delayed. And some data is not available because the order might be in (created, shipped, approved, canceled, invoiced, processing, unavailable) state. So we are not able to find the delivery\_time and diff\_estimated\_time for that particular orders.

### RECOMMENDATIONS:

There are few states like RJ, ES, SP, SE, PA and some other states, for these states we are taking more time to deliver than expected time. If the reason is distance of transportation then it is better to install a warehouse to eliminate this high difference in delivery date. If order got delayed it might effect the order to be cancelled and impact customer satisfaction. If we are making good sales in those areas then it is better to install a warehouse for faster delivery.

**Q2) Find out the top 5 states with the highest & lowest average freight value.**

ANS:

**QUERY:**

```
with cte as (
SELECT c.customer_state,i.freight_value
from `geolocation.customers` c
left join `geolocation.orders` o
on c.customer_id = o.customer_id
left join `geolocation.order_items` i
on o.order_id = i.order_id
)
(select customer_state, avg(freight_value) as avg_freight_value
from cte
group by customer_state
order by avg_freight_value desc
LIMIT 5)
UNION ALL
(select customer_state, avg(freight_value) as avg_freight_value
from cte
group by customer_state
order by avg_freight_value ASC
LIMIT 5)
```

Row	customer_state	avg_freight_value	Row	customer_state	avg_freight_value
1	RR	42.98442307692...	1	SP	15.14727539041...
2	PB	42.72380398671...	2	PR	20.53165156794...
3	RO	41.06971223021...	3	MG	20.63016680630...
4	AC	40.07336956521...	4	RJ	20.96092393168...
5	PI	39.14797047970...	5	DF	21.04135494596...

Row	customer_state	avg_freight_value
1	RR	42.98442307692...
2	PB	42.72380398671...
3	RO	41.06971223021...
4	AC	40.07336956521...
5	PI	39.14797047970...
6	SP	15.14727539041...
7	PR	20.53165156794...
8	MG	20.63016680630...
9	RJ	20.96092393168...
10	DF	21.04135494596...

### INSIGHTS:

As per my Analysis, we are getting more orders from SP so we are having very low average freight cost for that state. But in case of RR,PB,RO,AC,PI, we are spending more money on delivering each product as number of orders being placed are very less.

### RECOMMENDATIONS:

So in order to decrease the freight cost we have to increase the number of sales in the states. We can achieve this through advertisements, discounts and we can also use local warehouses to reduce the cost of transportation.

### Q3) Find out the top 5 states with the highest & lowest average delivery time.

ANS:

### QUERY:

```

SELECT geolocation_state, avg(delivery_time) as avg_time
from(
SELECT g.geolocation_state,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) as
`delivery_time`
FROM `geolocation.orders` o
left join `geolocation.customers` c
on o.customer_id = c.customer_id
left join `geolocation.geolocation` g

```

```

on g.geolocation_zip_code_prefix = c.customer_zip_code_prefix
where order_status = 'delivered'
)
group by geolocation_state
order by avg_time desc
limit 5

```

JOB INFORMATION		RESULTS	CHART	JSON
Row	geolocation_state ▼	avg_time ▼		
1	AP	27.99122623772...		
2	AM	24.65119678421...		
3	RR	24.52060133630...		
4	AL	23.14352789271...		
5	PA	22.55023982441...		

```

SELECT geolocation_state, avg(delivery_time) as avg_time
from(
SELECT g.geolocation_state,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) as
`delivery_time`
FROM `geolocation.orders` o
left join `geolocation.customers` c
on o.customer_id = c.customer_id
left join `geolocation.geolocation` g
on g.geolocation_zip_code_prefix = c.customer_zip_code_prefix
where order_status = 'delivered'
)
group by geolocation_state
order by avg_time asc
limit 5

```

JOB INFORMATION		RESULTS	CHART	JSON
Row	geolocation_state ▼	avg_time ▼		
1	SP	8.470555045095...		
2	PR	11.03876404770...		
3	MG	11.41821678372...		
4	DF	12.49651789233...		
5	SC	14.48408434580...		

**INSIGHTS:** As per my Analysis, Average delivery time of SP, PR, MG, DF, SC is very good. But in case of AP, AM, RR, AL, PA delivery time is very high due to distance or low number of orders being placed.

### RECOMMENDATIONS:

For these states AP, AM, RR, AL, PA we are taking more time to deliver than expected time. If the reason is transportation then it is better to install a warehouse to eliminate this high

difference in delivery date. If order got delayed it might affect the order to be cancelled and impact customer satisfaction. If we are making good sales in those areas then it is better to install a warehouse for faster delivery.

**Q4) 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.**

ANS:

QUERY:

```
SELECT distinct g.geolocation_state,
AVG(DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date,DAY)) as
`diff_estimated_delivery`
FROM `geolocation.orders` o
left join `geolocation.customers` c
on o.customer_id = c.customer_id
left join `geolocation.geolocation` g
on g.geolocation_zip_code_prefix = c.customer_zip_code_prefix
where o.order_status = 'delivered'and order_delivered_customer_date is not null and
order_estimated_delivery_date is not null
GROUP by g.geolocation_state
order by diff_estimated_delivery asc
LIMIT 5
```

Row	geolocation_state	diff_estimated_delive
1	RR	-20.4203786191...
2	AM	-20.1326511967...
3	RO	-18.6520972167...
4	AC	-18.4614566719...
5	AP	-18.1825778149...

INSIGHTS:

For these states we are delivering faster than expected time of delivery. It seems like a good thing but if we display the estimated time of delivery to the customer, they may cancel their order out of concern for the projected delivery duration.

RECOMMENDATION:

My suggestion is to re-evaluate the delivery times for orders going to those states. This way, we can provide customers with a more accurate estimate of the delivery time.

**Analysis based on the payments:**

**Q1) Find the month on month no. of orders placed using different payment types.**

ANS:

**QUERY:**

```
select MofSale, YofSale, payment_type, count(*) as no_of_sales
from
(
select o.order_id, p.payment_type, extract(month from o.order_purchase_timestamp) as
MofSale, extract(year from o.order_purchase_timestamp) as YofSale
from `geolocation.payments` as p
left join `geolocation.orders` as o
on o.order_id = p.order_id
)
group by YofSale, MofSale, payment_type
order by YofSale, MofSale, payment_type
```

MofSale	YofSale	payment_type	no_of_sales
9	2016	credit_card	3
10	2016	UPI	63
10	2016	credit_card	254
10	2016	debit_card	2
10	2016	voucher	23
12	2016	credit_card	1
1	2017	UPI	197
1	2017	credit_card	583
1	2017	debit_card	9
1	2017	voucher	61
2	2017	UPI	398
2	2017	credit_card	1356
2	2017	debit_card	13
2	2017	voucher	119

**INSIGHTS:** As per my analysis, most of payments are done using credit cards, the reason might be because of the offers available through the credit cards. And UPI is the second favourite payment method for our customers. And the other thing I've noticed was there is payment status of cash.

**RECOMMENDATION:** Some people may not credit card like people of lower age. So we have to make sure they also purchase good goods from us. We can give different discounts for children, so they can love purchasing from us.

**Q2) Find the no. of orders placed on the basis of the payment installments that have been paid.**

**ANS:**

**QUERY:**



```

select payment_installments, count(*) as no_of_sales
from
(
select o.order_id,p.payment_installments,extract(month from
o.order_purchase_timestamp) as MofSale,extract(year from
o.order_purchase_timestamp) as YofSale
from `geolocation.payments` as p
left join `geolocation.orders` as o
on p.order_id = o.order_id
)
group by payment_installments
order by payment_installments

```

Row	payment_installment	no_of_sales
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
12	11	23
13	12	133
14	13	16

**INSIGHTS:** As per my Analysis, Most of the customers are paying their bills in one installment. And many customers are paying their Installments within an year.

**RECOMMENDATION:** N/A