Learning Objectives

1) Introduction
2) Usual exploratory analysis of Target's e-commerce dataset
3) In-depth Exploration of dataset
4) Evolution of E-commerce orders in the Brazil region
5) Impact on Economy: Analysis of the money movement by e-commerce by looking at order prices, freight and others
6) Analysis based on sales, freight and delivery time
7) Analysis based on the payments
8) Conclusion and recommendations

1) Introduction

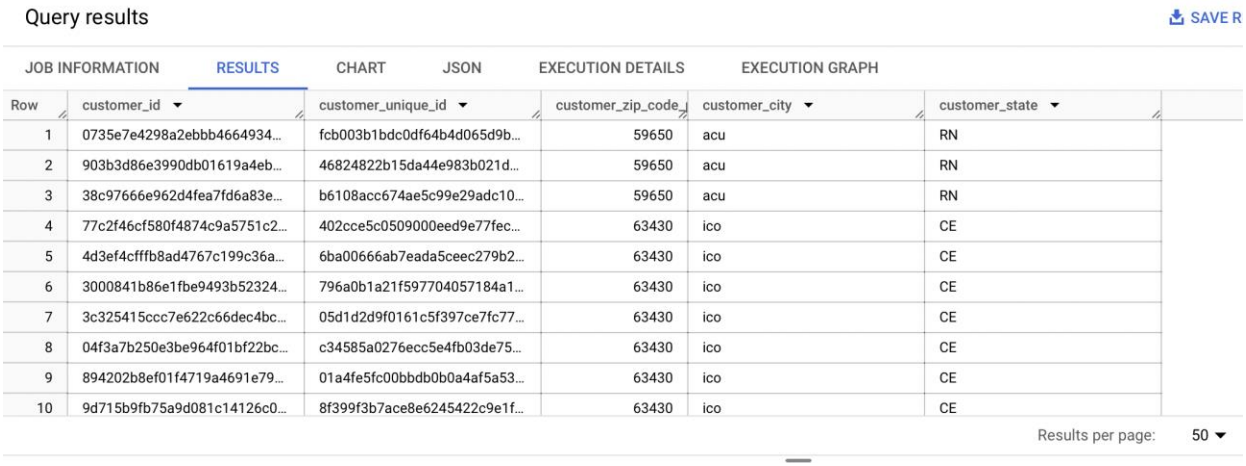
"Target" is a company which has expanded its e-commerce operations in Brazil for growing the online shopping trend. To make success, it is important for Target to understand the working of e-commerce in Brazil and make planning for growing it. In this report, it is an analysis of Target's e-commerce dataset using SQL queries.

2) Usual exploratory analysis of Target's e-commerce dataset

For analysis of dataset we must check the data type of columns of customer table to make it easy for further analysis in SQL. The query we can use is:

```
select * from `target.customers`;
```

And we get output as:


Query results

SAVE R

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	customer_id	customer_unique_id	customer_zip_code_	customer_city	customer_state
1	0735e7e4298a2ebbb4664934...	fcb003b1bdc0df64b4d065d9b...	59650	acu	RN
2	903b3d86e3990db01619a4eb...	46824822b15da44e983b021d...	59650	acu	RN
3	38c97666e962d4fea7fd6a83e...	b6108acc674ae5c99e29adc10...	59650	acu	RN
4	77c2f46cf580f4874c9a5751c2...	402cce5c0509000eed9e77fec...	63430	ico	CE
5	4d3ef4cfffb8ad4767c199c36a...	6ba00666ab7eada5ceec279b2...	63430	ico	CE
6	3000841b86e1fbe9493b52324...	796a0b1a21f597704057184a1...	63430	ico	CE
7	3c325415ccc7e622c66dec4bc...	05d1d2d9f0161c5f397ce7fc77...	63430	ico	CE
8	04f3a7b250e3be964f01bf22bc...	c34585a0276ecc5e4fb03de75...	63430	ico	CE
9	894202b8ef01f4719a4691e79...	01a4fe5fc00bbdb0b0a4af5a53...	63430	ico	CE
10	9d715b9fb75a9d081c14126c0...	8f399f3b7ace8e6245422c9e1f...	63430	ico	CE

Results per page: 50

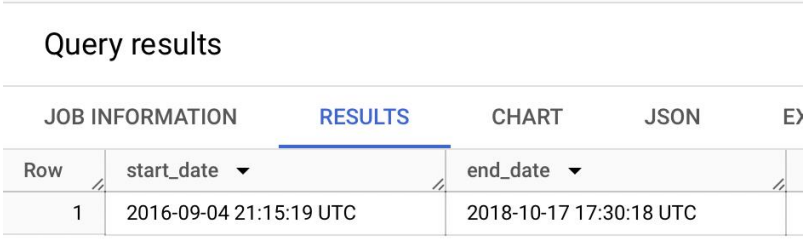
In this output we observe that columns are

Columns	Data Type
1) Customer_id	STRING
2) customer_unique_id	STRING
3) Customer_zip_code	INT
4) Customer_city	STRING
5) Customer_state	STRING
6) row	INT

From the analysis of first and last order placed in this data set, the start and end of the data are 4th SEPTEMBER 2016 and 17th OCTOBER 2018. The SQL query is used:

```
SELECT
DISTINCT MIN(order_purchase_timestamp) start_date,
MAX(order_purchase_timestamp) end_date
FROM
`target.orders`;
```

And the output we get as :



Then we analyze the count of the number of unique cities & states where orders were placed by the customers during the given time period. The SQL query used:

```
SELECT DISTINCT c.customer_city, c.customer_state, COUNT(o.customer_id) order_count
FROM target.orders o
JOIN target.customers c
ON o.customer_id = c.customer_id
GROUP BY 1, 2
ORDER BY 3 DESC;
```

The result we get as:

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXI

Row	customer_city	customer_state	order_count
1	sao paulo	SP	15540
2	rio de janeiro	RJ	6882
3	belo horizonte	MG	2773
4	brasilia	DF	2131
5	curitiba	PR	1521
6	campinas	SP	1444
7	porto alegre	RS	1379
8	salvador	BA	1245
9	guarulhos	SP	1189
10	sao bernardo do campo	SP	938

The graphical representation :

Here we observe that Sao paulo city from SP state has highest order counts which is more than the sum of next 5 cities. Because the sao paulo city is very rich city.

3) In-depth Exploration of Dataset

Here we have to explore the growing trend of online shopping in Brazil. To dermine growing trend we have to count the number of orders placed over the past years. We use the SQL query as:

```
SELECT
EXTRACT(YEAR FROM o.order_purchase_timestamp) AS year,
EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
COUNT(DISTINCT o.order_id) AS order_count
FROM `target.orders` o
JOIN `target.customers` c
ON o.customer_id = c.customer_id
GROUP BY year, month
ORDER BY year, month;
```

The result we get as:

Row	year	month	order_count
1	2016	9	4
2	2016	10	324
3	2016	12	1
4	2017	1	800
5	2017	2	1780
6	2017	3	2682
7	2017	4	2404
8	2017	5	3700
9	2017	6	3245
10	2017	7	4026

From the analysis we observe that the trend of online shopping increases from 2016 to 2018 on the basis of order count. But only order counting doesn't imply business growth. To increase the business, we must focus on revenue growth.

Now we have to analyze any seasonality on the basis of orders. To analyze seasonality by month we use SQL query as:

SELECT

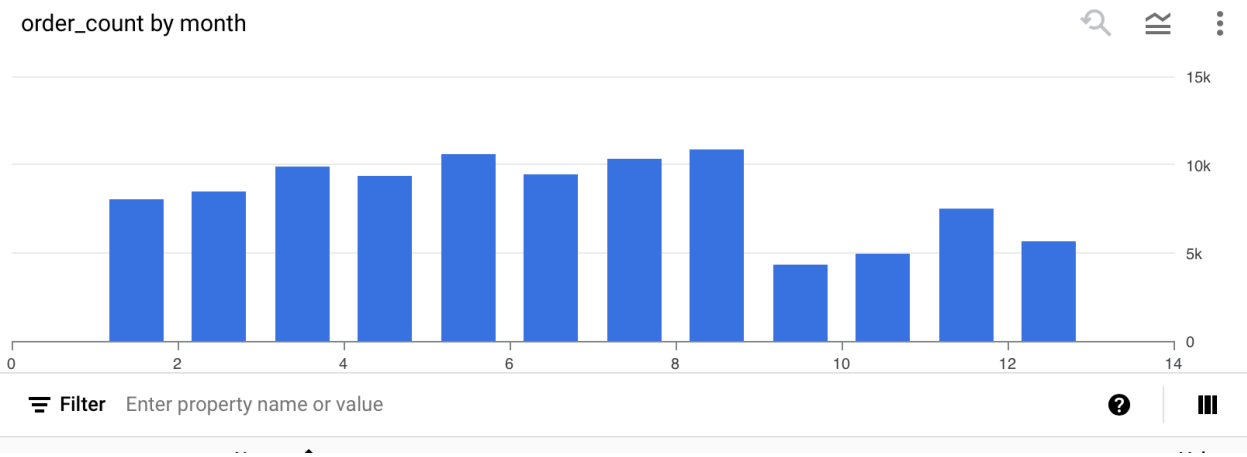
```
EXTRACT(MONTH FROM order_purchase_timestamp) AS month,
COUNT(DISTINCT order_id) AS order_count
FROM `target.orders`
GROUP BY month
ORDER By month;
```

The result we get as:

JOB INFORMATION | RESULTS | CHART

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

To be more precise we analyze it by bar diagram:



From the results we observe that the months MAY, JULY and AUGUST has highest orders place in these years. It implies that some national or cultural festivals are celebrated in Brazil in these months and the orders suddenly rises.

Now we have to analyze the time range in a day at which most orders are placed. To analyse it we use SQL query as:

```
SELECT
CASE
WHEN EXTRACT(HOUR FROM o.order_purchase_timestamp) BETWEEN 0 AND 5 THEN
'Dawn'
WHEN EXTRACT(HOUR FROM o.order_purchase_timestamp) BETWEEN 6 AND 11 THEN
'Morning'
WHEN EXTRACT(HOUR FROM o.order_purchase_timestamp) BETWEEN 12 AND 17
THEN 'Afternoon'
WHEN EXTRACT(HOUR FROM o.order_purchase_timestamp) BETWEEN 18 AND 23
THEN 'Night'
END AS hour,
COUNT(o.order_id) AS order_count
FROM `target.orders` o
JOIN `target.customers` c
ON o.customer_id = c.customer_id
GROUP BY hour
ORDER BY order_count DESC;
```

The result we get as :

JOB INFORMATION | RESULTS | CHART | JSON

Row	hour	order_count
1	Afternoon	38361
2	Night	34100
3	Morning	22240
4	Dawn	4740

From this analysis we observe that the maximum orders placed during 12:00:00 to 17:00:00 hours. And the minimum orders placed between 00:00:00 to 05:00:00 hours.

4) Evolution of e-commerce orders in Brazil region

To understand the evolution of e-commerce orders in Brazil region we analyzed the month-on-month orders counts for each state. The SQL query used as:

```
SELECT
c.customer_state,
EXTRACT(month FROM o.order_purchase_timestamp) AS month,
COUNT(o.order_purchase_timestamp) AS order_count
FROM `target.orders` o
JOIN `target.customers` c
ON o.customer_id = c.customer_id
GROUP BY c.customer_state, month
ORDER BY c.customer_state, month;
```

The output result we get as:

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS

Row	customer_state	month	order_count
1	AC	1	8
2	AC	2	6
3	AC	3	4
4	AC	4	9
5	AC	5	10
6	AC	6	7
7	AC	7	9
8	AC	8	7
9	AC	9	5
10	AC	10	6

From the result we get, the number of order counts of each state in each month.

To further explore the e-commerce landscape in Brazil, we explained the distribution of customers across the states. The SQL query is used as:

```
SELECT
c.customer_state,
COUNT(c.customer_id) AS no_of_customers
FROM `target.customers` c
GROUP BY c.customer_state
ORDER BY no_of_customers DESC;
```

The result we get as:

JOB INFORMATION | RESULTS | CHART | JSON

Row	customer_state	no_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

From the result we observe that the state SP has the highest number of customers.

5) Impact on Economy: Analysis of the money movement by e-commerce by looking at order prices, freight and others

To analyze the money movement by e-commerce, we have to calculate the percentage increase in the cost of orders from 2017 to 2018 in the months between January to August. We use the SQL query as :

```
SELECT
EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
(
(
SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2018 AND
EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8 THEN
p.payment_value END) -
SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017 AND
EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8 THEN
p.payment_value END)
)
/
SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017 AND
EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8 THEN
p.payment_value END)
)*100 AS percent_increase
FROM `target.orders` o
JOIN `target.payments` p
ON o.order_id = p.order_id
WHERE
EXTRACT(YEAR FROM o.order_purchase_timestamp) IN (2017, 2018) AND
EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8
GROUP BY 1
ORDER BY 1;
```

The result obtained from this query as:

JOB INFORMATION | RESULTS | CHART

Row	month	percent_increase
1	1	705.1266954171...
2	2	239.9918145445...
3	3	157.7786066709...
4	4	177.8407701149...
5	5	94.62734375677...
6	6	100.2596912456...
7	7	80.04245463390...
8	8	51.60600520477...

From this information we observe that the highest percentage increase has been done in the cost of orders from 2017 to 2018 in January month. And the lowest is in August month.

Now to analyze total and average price of orders for each state also we have to analyze total and average price of order freight for each state. We used SQL query as :

```
SELECT
c.customer_state,
ROUND(AVG(i.price), 2) AS mean_price,
ROUND(SUM(i.price), 2) AS total_price,
ROUND(AVG(i.freight_value), 2) AS mean_freight_value,
ROUND(SUM(i.freight_value), 2) AS total_freight_value
FROM `target.orders` o
JOIN `target.order_items` i ON o.order_id = i.order_id
JOIN `target.customers` c ON o.customer_id = c.customer_id
GROUP BY c.customer_state;
```

We get the result as :

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	customer_state	mean_price	total_price	mean_freight_value	total_freight_value
1	MT	148.3	156453.53	28.17	29715.43
2	MA	145.2	119648.22	38.26	31523.77
3	AL	180.89	80314.81	35.84	15914.59
4	SP	109.65	5202955.05	15.15	718723.07
5	MG	120.75	1585308.03	20.63	270853.46
6	PE	145.51	262788.03	32.92	59449.66
7	RJ	125.12	1824092.67	20.96	305589.31
8	DF	125.77	302603.94	21.04	50625.5
9	RS	120.34	750304.02	21.74	135522.74
10	SE	153.04	58920.85	36.65	14111.47

The analysis reveals interesting findings. While São Paulo (SP) has the highest total price value and total freight value, it surprisingly has the lowest average price value and average freight value among all states. On the other hand, the state of Paraíba (PB) has the highest average price value and average freight value.

6) Analysis based on sales, freight and delivery time

To understand the time duration between purchasing an order, its delivery, and the estimated delivery, we calculated the number of days using the following SQL query:

```
SELECT
order_id,
DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY)
AS delivered_in_days,
DATE_DIFF(order_estimated_delivery_date, order_purchase_timestamp, DAY)
AS estimated_delivery_in_days,
DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date, DAY) AS
estimated_minus_actual_delivery_days
FROM `target.orders`
WHERE DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY) IS
NOT NULL
ORDER BY delivered_in_days;
```

The result of this query we get as:

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION G

Row	order_id	delivered_in_days	estimated_delivery_j	estimated_minus_ac
1	e65f1eeee1f52024ad1dcd034...	0	10	9
2	bb5a519e352b45b714192a02f...	0	26	25
3	434cecee7d1a65fc65358a632...	0	20	19
4	d3ca7b82c922817b06e5ca211...	0	12	11
5	1d893dd7ca5f77ebf5f59f0d20...	0	10	10
6	d5fbeedc85190ba88580d6f82...	0	8	7
7	79e324907160caea526fd8b94...	0	9	8
8	38c1e3d4ed6a13cd0cf612d4c...	0	17	16
9	8339b608be0d84fca9d8da68b...	0	28	27
10	f349cdb62f69c3fae5c4d7d3f3...	0	13	12

Finding average time to delivery and average difference in estimated delivery state-wise:
To gain insights the SQL query is used as:

```
SELECT
c.customer_state,
ROUND(AVG(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,
DAY)), 2)
AS avg_time_to_delivery,
ROUND(AVG(DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date,
DAY)), 2)
AS avg_diff_estimated_delivery
FROM `target.orders` o
JOIN `target.customers` c ON o.customer_id = c.customer_id
WHERE
DATE_DIFF(order_purchase_timestamp, order_delivered_customer_date, DAY) IS NOT
NULL
AND
DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date, DAY) IS NOT
NULL
GROUP BY c.customer_state
ORDER BY avg_time_to_delivery;
```

The result of this query is :

Row	customer_state	avg_time_to_delivery	avg_diff_estimated_d
1	SP	8.3	10.14
2	PR	11.53	12.36
3	MG	11.54	12.3
4	DF	12.51	11.12
5	SC	14.48	10.61
6	RS	14.82	12.98
7	RJ	14.85	10.9
8	GO	15.15	11.27
9	MS	15.19	10.17
10	ES	15.33	9.62

From the results we can observe that state SP has lowest average time to delivery while the state RR has highest average time to delivery.

• Top five states with highest average freight value: the SQL query used as

```
SELECT
c.customer_state,
ROUND(AVG(i.freight_value), 2) AS mean_freight_value,
FROM `target.orders` o
JOIN `target.order_items` i ON o.order_id = i.order_id
JOIN `target.customers` c ON o.customer_id = c.customer_id
GROUP BY c.customer_state
order by mean_freight_value desc
limit 5;
```

The result is :

JOB INFORMATION | RESULTS | CHART | JSON

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

• Bottom five states with lowest average freight value: the SQL query is used

```
SELECT
c.customer_state,
ROUND(AVG(i.freight_value), 2) AS mean_freight_value,
FROM `target.orders` o
JOIN `target.order_items` i ON o.order_id = i.order_id
JOIN `target.customers` c ON o.customer_id = c.customer_id
GROUP BY c.customer_state
order by mean_freight_value
limit 5;
```

The result is

JOB INFORMATION | RESULTS | CHART | JSON

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

7) Analysis based on payments

We have to analyze month on month number of orders placed using different payment types. The SQL query is executed as:

```
SELECT
p.payment_type,
EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
COUNT(DISTINCT o.order_id) AS order_count
FROM `target.orders` o
JOIN `target.payments` p
ON o.order_id = p.order_id
GROUP BY 1, 2
ORDER BY 1, 2;
```

The result we got as :

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION

Row	payment_type	month	order_count
1	UPI	1	1715
2	UPI	2	1723
3	UPI	3	1942
4	UPI	4	1783
5	UPI	5	2035
6	UPI	6	1807
7	UPI	7	2074
8	UPI	8	2077
9	UPI	9	903
10	UPI	10	1056

From the analysis it shows that the credit card UPI type of payment mode is preferable by customers.

Now we have to calculate the number of orders placed on the basis of the payment installments that have been paid. The SQL query is used as:

```
SELECT
p.payment_installments,
COUNT(o.order_id) AS order_count
FROM `target.orders` o
JOIN `target.payments` p
ON o.order_id = p.order_id
WHERE o.order_status != 'canceled'
GROUP BY 1
ORDER BY 2 DESC;
```

Row	payment_installment	order_count
1	1	52184
2	2	12353
3	3	10392
4	4	7056
5	10	5292
6	5	5209
7	8	4239
8	6	3898
9	7	1620
10	9	638

The analysis reveals that the majority of orders (maximum count) have only one payment installment. The highest number of installments is 24, which is associated with 18 orders.

8) Conclusion

From the analysis of e-commerce data in the Brazilian market provides valuable insights into customer buying pattern, sales trends, payments preferences, delivery experiences etc. By understanding these patterns and trends, business can make and implement strategies to optimize their operations and growth. Here are some key outputs:

• The state SP dominates e-commerce market of Brazil. It implies need to focus on other states for potential growth.

• Offering the discounts during off–peak season can attract the customers and boost the sale during slower period

Recommendations

• Improve the logistic and shipping processes to reduce delivery time.

• Avail the offers, discounts, referrals for the customer to attract customers.

• Collaborate with sellers and design the issues regarding products and improve product quality all related issues