Target SQL

Analysis & Insights
for
America's leading Retail Chain

By SAYED ASHFAQ

About Target

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.

Basic Analysis

1. Initial Exploration of Dataset

1.1 Show all the tables and all the columns present in each table along with its data type.

Query:

```
SELECT
  table_schema,
  table_name,
  column_name,
  data_type
FROM `scaler-project-1-439305.Target_SQL`.INFORMATION_SCHEMA.COLUMNS;
```

Query results

SAVE RESULTS

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

Row	table_schema	table_name	column_name	data_type
1	Target_SQL	order_items	order_id	STRING
2	Target_SQL	order_items	order_item_id	INT64
3	Target_SQL	order_items	product_id	STRING
4	Target_SQL	order_items	seller_id	STRING
5	Target_SQL	order_items	shipping_limit_date	TIMESTAMP
6	Target_SQL	order_items	price	FLOAT64
7	Target_SQL	order_items	freight_value	FLOAT64
8	Target_SQL	sellers	seller_id	STRING
9	Target_SQL	sellers	seller_zip_code_prefix	INT64
10	Target_SQL	sellers	seller_city	STRING
11	Target_SQL	sellers	seller_state	STRING
12	Target_SQL	geolocation	geolocation_zip_code_prefix	INT64
13	Target_SQL	geolocation	geolocation_lat	FLOAT64
14	Target_SQL	geolocation	geolocation_lng	FLOAT64

Results per page: 50 1 – 49 of 49

1.2 Time period for which the data is given

Query:

```
SELECT
  MIN(DATE(order_purchase_timestamp)) as Start_date,
  MAX(DATE(order_purchase_timestamp)) as End_date
FROM `scaler-project-1-439305.Target_SQL.orders`;
```

Query results:

Row	Start_date	End_date
1	2016-09-04	2018-10-17

1.3. From which Cities and States , orders were placed during the given period.

Query:

```
SELECT
  DISTINCT c.customer_city as Cities,
  c.customer_state AS States
FROM `scaler-project-1-439305.Target_SQL.customers`c
RIGHT JOIN `scaler-project-1-439305.Target_SQL.orders` o
```

```
USING
(customer_id)
ORDER BY Cities, States;
```

Query results:

Row	Cities	States
1	abadia dos dourados	MG
2	abadiania	GO
3	abaete	MG
4	abaetetuba	PA
5	abaiara	CE
6	abaira	BA
7	abare	BA
8	abatia	PR
9	abdon batista	SC
10	abelardo luz	SC
11	abrantes	BA
12	abre campo	MG
13	abreu e lima	PE

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:

```
WITH MonthlyOrders AS (
  SELECT
    FORMAT_DATE('%b %Y',order_purchase_timestamp) AS MonthYear,
    EXTRACT(MONTH FROM order_purchase_timestamp) as Month,
    EXTRACT(YEAR FROM order_purchase_timestamp) as Year,
    count(order_id) as Orders
  FROM `scaler-project-1-439305.Target_SQL.orders`
  GROUP BY MonthYear, Month, Year
  ORDER BY Year , Month),

prev_month_orders as (SELECT
  my.MonthYear,
  my.Orders,
  LAG(my.Orders, 1) OVER (ORDER BY my.Year ASC, my.Month ASC) as Prev_order
FROM MonthlyOrders my
ORDER BY my.Year ASC, my.Month ASC)

SELECT
  MonthYear,
  Orders,
  ROUND(((Orders-Prev_order)*100.0/orders),2) as Monthly_Turnover
FROM prev_month_orders;
```

Query results:

MonthYear	Orders	Monthly_Turnover
Sep 2016	4	*null*
Oct 2016	324	98.77
Dec 2016	1	-32300.0
Jan 2017	800	99.88
Feb 2017	1780	55.06
Mar 2017	2682	33.63
Apr 2017	2404	-11.56
May 2017	3700	35.03
Jun 2017	3245	-14.02
Jul 2017	4026	19.4
Aug 2017	4331	7.04
Sep 2017	4285	-1.07
Oct 2017	4631	7.47
Nov 2017	7544	38.61

Insights:

The analysis of e-commerce data shows a clear growing trend in order volumes over time, indicating the sector's consistent expansion. While there are noticeable peaks in May and November, likely driven by specific events like sales or promotions (e.g., Black Friday), However there is no recurring seasonal pattern observed across other months, suggesting that the increase in activity during these months is event-specific rather than seasonal.

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

Query:

```
WITH TimePeriods AS (
  SELECT
    order_id,
    CASE
      WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 0 AND 5 THEN 'Dawn'
      WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 6 AND 11 THEN 'Morning'
      WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 12 AND 17 THEN 'Afternoon'
      WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 18 AND 23 THEN 'Night'
    END AS TimePeriod
  FROM `scaler-project-1-439305.Target_SQL.orders`
)

SELECT
  TimePeriod,
  COUNT(order_id) AS OrderCount,
  ROUND((COUNT(order_id) * 100.0) / SUM(COUNT(order_id)) OVER (), 2) AS Percentage
FROM TimePeriods
GROUP BY TimePeriod
ORDER BY OrderCount DESC;
```

Query results:

TimePeriod	OrderCount	Percentage
Afternoon	38361	38.58
Night	34100	34.29
Morning	22240	22.37
Dawn	4740	4.77

Insights:

From the analysis, we can conclude that Afternoon has the highest number of purchases, followed by Night, suggesting that customers likely utilize their free time in the afternoon or get reminded of pending purchases during work hours, prompting them to place orders.

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

1. Get month on month orders by region, states

Query:

```
WITH MonthlyOrderStates as (
  SELECT
    FORMAT_DATETIME('%b-%Y', o.order_purchase_timestamp) as MonthYear,
    EXTRACT(MONTH FROM O.order_purchase_timestamp) AS Month,
    EXTRACT(YEAR FROM O.order_purchase_timestamp) AS Year,
    c.customer_city,
    c.customer_state,
    COUNT(o.order_id) as orders
  FROM `scaler-project-1-439305.Target_SQL.orders` o
  JOIN `scaler-project-1-439305.Target_SQL.customers` c
  USING (customer_id)
  GROUP BY MonthYear, Month, Year, customer_city, customer_state
)

SELECT
  Monthyear,
  customer_city,
  customer_state,
  orders
FROM MonthlyOrderStates
ORDER BY orders DESC;
```

Query Result:

Monthyear	customer_city	customer_state	orders
Aug-2018	sao paulo	SP	1308
May-2018	sao paulo	SP	1222
Apr-2018	sao paulo	SP	1166
Mar-2018	sao paulo	SP	1151
Nov-2017	sao paulo	SP	1118
Jan-2018	sao paulo	SP	1098
Jul-2018	sao paulo	SP	1084
Jun-2018	sao paulo	SP	1054
Feb-2018	sao paulo	SP	1037
Dec-2017	sao paulo	SP	840
Aug-2017	sao paulo	SP	646
Oct-2017	sao paulo	SP	624
Sep-2017	sao paulo	SP	598
Nov-2017	rio de janeiro	RJ	578

Insights:

The dominance of orders from Sao Paulo (SP) and Rio de Janeiro (RJ) highlights their large populations, higher purchasing power, and strong e-commerce adoption. To capitalize, the company should strengthen logistics and inventory in these cities, while expanding marketing efforts to nearby regions and underperforming Tier-2 and Tier-3 cities to grow the customer base. Personalized marketing and exclusive offers can enhance retention in these core markets

2. How are customers distributed in Brazil

Query:

```
SELECT
  customer_state,
  count(*) as Total_Customers
FROM `scaler-project-1-439305.Target_SQL.customers`
GROUP BY customer_state
ORDER BY COUNT(*) DESC;
```

Query results

JOB INFORMATION | RESULTS | CHART

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

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)

Query:

```
SELECT
  Year,
  Cost_of_Orders,
  ROUND(((Cost_of_Orders-(LAG(Cost_of_Orders, 1) OVER (ORDER BY YEAR)))*100.0/(LAG(Cost_of_Orders, 1)
OVER (ORDER BY YEAR))),2) AS Percetage_Increased

FROM (
  SELECT
    EXTRACT(YEAR FROM o.order_purchase_timestamp) AS Year,
    sum(oi.price) as Cost_of_Orders
  FROM `scaler-project-1-439305.Target_SQL.orders` o
  JOIN `scaler-project-1-439305.Target_SQL.order_items` oi
  USING (order_id)
  WHERE EXTRACT(MONTH FROM order_purchase_timestamp) BETWEEN 1 AND 8
  group by year) temp
  ORDER BY Year;
```

Query result: There is 137.26% increase in cost of orders from 2017 to 2018, Rising Inflation and other factors might have affected the Cost to Increase.

Year	Cost_of_Orders	Percetage_Increased
2017	3113000.31999...	null
2018	7385905.80000...	137.26

2. Mean & Sum of price and freight value by customer state 💵🚚

Query:

```
SELECT
  c.customer_state,
  ROUND(AVG(oi.price),2) AS Price_mean,
  ROUND(AVG(oi.freight_value),2) AS freight_mean,
  ROUND(SUM(oi.price),2) AS Total_price,
  ROUND(SUM(oi.freight_value),2) AS total_freight_value,
FROM `scaler-project-1-439305.Target_SQL.customers` c
JOIN `scaler-project-1-439305.Target_SQL.orders` o
ON o.customer_id = c.customer_id
JOIN `scaler-project-1-439305.Target_SQL.order_items` oi
USING (order_id)
GROUP BY c.customer_state
ORDER BY Price_mean desc, freight_mean DESC;
```

Query result:

customer_state	Price_mean	freight_mean	Total_price	total_freight_value
PB	191.48	42.72	115268.08	25719.73
AL	180.89	35.84	80314.81	15914.59
AC	173.73	40.07	15982.95	3686.75
RO	165.97	41.07	46140.64	11417.38
PA	165.69	35.83	178947.81	38699.3
AP	164.32	34.01	13474.3	2788.5
PI	160.36	39.15	86914.08	21218.2
TO	157.53	37.25	49621.74	11732.68
RN	156.97	35.65	83034.98	18860.1

5. Analysis on sales, freight and delivery time

1. Calculate days between purchasing, delivering and estimated delivery

Query:

```
SELECT
  Order_id,
  DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) AS Actual_delivery_time,
  DATE_DIFF(order_estimated_delivery_date,order_purchase_timestamp, DAY) AS
Estimated_delivery_time
FROM `scaler-project-1-439305.Target_SQL.orders`
WHERE order_status = 'delivered'
ORDER BY Actual_delivery_time DESC;
```

Query Results:

Order_id	Actual_delivery_time	Estimated_delivery_
ca07593549f1816d26a572e06...	209	28
1b3190b2dfa9d789e1f14c05b...	208	19
440d0d17af552815d15a9e41a...	195	30
0f4519c5f1c541ddec9f21b3bd...	194	32
285ab9426d6982034523a855f...	194	28
2fb597c2f772eca01b1f5c561b...	194	39
47b40429ed8cce3aee9199792...	191	15
2fe324febf907e3ea3f2aa9650...	189	22
2d7561026d542c8dbd8f0daea...	188	28
437222e3fd1b07396f1d9ba8c...	187	42
c27815f7e3dd0b926b5855262...	187	25
dfe5f68118e2576143240b8d7	186	32

Insights:

The delivery performance analysis shows significant variability between estimated and actual delivery times, with some orders taking up to **209 days** despite an estimated delivery of **28 days**, risking customer trust and satisfaction. Conversely, in states like **São Paulo (SP)** and **Rio de Janeiro (RJ)**, some orders with an estimated delivery of **29-30 days** were delivered on the same day, potentially raising customer expectations for faster delivery. This inconsistency highlights the need to optimize logistics for delayed orders while setting more realistic and reliable delivery timeframes to balance customer expectations and operational capabilities.

2. Create columns:

time_to_delivery = order_purchase_timestamp - order_delivered_customer_date

diff_estimated_delivery = order_estimated_delivery_date - order_delivered_customer_date

Query:

```
SELECT *,
  DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY) AS time_to_deliver,
  DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date,DAY) AS
diff_estimated_time_to_deliver
FROM `scaler-project-1-439305.Target_SQL.orders`
WHERE order_status = 'delivered'
LIMIT 1000
```

Query results:

order_estimated_delivery_date	time_to_deliver	estimated_time_to_d
2017-05-18 00:00:00 UTC	23	9
2017-05-18 00:00:00 UTC	12	-5
2017-05-18 00:00:00 UTC	12	12
2017-05-18 00:00:00 UTC	7	1
2017-05-18 00:00:00 UTC	12	9
2017-05-18 00:00:00 UTC	1	5
2017-05-18 00:00:00 UTC	6	0
2017-05-18 00:00:00 UTC	21	7
2017-05-18 00:00:00 UTC	7	0
2017-05-18 00:00:00 UTC	30	1

Insights:

The newly created columns, **time_to_deliver** and **diff_estimated_time_to_deliver**, provide valuable insights into actual delivery durations and the variance from estimated delivery times. These metrics enable a deeper understanding of delivery performance and inconsistencies. By conducting statistical analyses on these columns, the business can identify patterns, outliers, and factors affecting delivery delays, leading to the development of a more accurate and reliable method for estimating delivery times, ultimately enhancing customer satisfaction and operational efficiency.

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

Query:

```
WITH DeliveredOrders as (
  SELECT *,
    DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY) AS time_to_deliver,
    DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date,DAY) AS
     diff_estimated_time_to_deliver
  FROM `scaler-project-1-439305.Target_SQL.orders`
  WHERE order_status = 'delivered'
)

SELECT
  c.customer_state,
  ROUND(AVG(oi.freight_value),2) AS mean_freight_value,
  ROUND(AVG(o.time_to_deliver),2) AS mean_time_to_deliver,
  ROUND(AVG(o.diff_estimated_time_to_deliver),2) AS diff_estimated_delivery
FROM `scaler-project-1-439305.Target_SQL.customers` c
JOIN DeliveredOrders o
USING (customer_id)
JOIN `scaler-project-1-439305.Target_SQL.order_items` oi
USING (order_id)
GROUP BY c.customer_state;
```

Query Results:

customer_state	mean_freight_value	mean_time_to_delive	diff_estimated_deliv
RN	35.72	18.87	13.06
CE	32.73	20.54	10.26
RS	21.61	14.71	13.2
SC	21.51	14.52	10.66
SP	15.12	8.26	10.26
MG	20.63	11.51	12.4
BA	26.49	18.77	10.12
RJ	20.91	14.69	11.14
GO	22.56	14.95	11.37
MA	38.49	21.2	9.11
PE	32.69	17.79	12.55

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

Query:

```
-- Converting above query into CTE named averages and getting TOP 5 states with highest/lowest mean
freight_values

-- states with high freight_value
        SELECT
          customer_state,
          mean_freight_value
```

customer_state	mean_freight_value
PB	43.09
RR	43.09
RO	41.33
AC	40.05
PI	39.12

```
FROM Averages
ORDER BY mean_freight_value DESC
LIMIT 5;
```

```
-- states with low freight_value
SELECT
  customer_state,
  mean_freight_value
FROM Averages
ORDER BY mean_freight_value
LIMIT 5;
```

customer_state	mean_freight_value
SP	15.12
PR	20.47
MG	20.63
RJ	20.91
DF	21.07

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

```
-- states that took HIGHEST time to deliver
SELECT
  customer_state,
  mean_time_to_deliver
FROM Averages
ORDER BY mean_time_to_deliver DESC
LIMIT 5;
```

customer_state	mean_time_to_delive
PB	20.12
RR	27.83
RO	19.28
AC	20.33
PI	18.93

```
-- states that took least time to deliver
SELECT
  customer_state,
  mean_time_to_deliver
FROM Averages
ORDER BY mean_time_to_deliver
LIMIT 5;
```

customer_state	mean_time_to_delive
SP	8.26
PR	11.48
MG	11.51
RJ	14.69
DF	12.5

Insights:
States like **PP, RR, RO, AC, and PI** have the highest mean freight values, while **SP, PR, MG, RJ, and DF** have the lowest. Interestingly, states with the highest freight values also experience the longest delivery times, whereas states with lower freight values enjoy quicker deliveries. This suggests a **positive correlation** between delivery time and freight cost, indicating that as delivery time increases, the cost of delivery also rises. Addressing logistical inefficiencies in high-freight regions could reduce both delivery time and costs, improving overall customer satisfaction and operational efficiency

6. Payment type analysis:

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

Query:

```
WITH PaymentTypes AS (
  SELECT
    FORMAT_DATETIME('%b %Y',o.order_purchase_timestamp) AS MonthYear,
    EXTRACT(YEAR FROM o.order_purchase_timestamp) AS year,
    EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
```

```
    p.payment_type,
    COUNT(o.order_id) AS orders
  FROM `scaler-project-1-439305.Target_SQL.orders` o
  JOIN `scaler-project-1-439305.Target_SQL.payments` p
  USING (order_id)
  GROUP BY MonthYear, year, month, payment_type
)

SELECT
  MonthYear,
  payment_type,
  orders
FROM PaymentTypes
ORDER BY year, month;
```

Query Results:

MonthYear	payment_type	orders
Sep 2016	credit_card	3
Oct 2016	credit_card	254
Oct 2016	UPI	63
Oct 2016	voucher	23
Oct 2016	debit_card	2
Dec 2016	credit_card	1
Jan 2017	credit_card	583
Jan 2017	UPI	197
Jan 2017	voucher	61
Jan 2017	debit_card	9
Feb 2017	credit_card	1356

2. Distribution of payment installments and count of orders

Query:

```
SELECT
  p.payment_type,
  count(o.order_id) AS orders
FROM `scaler-project-1-439305.Target_SQL.payments` p
LEFT JOIN `scaler-project-1-439305.Target_SQL.orders` o
ON o.order_id = p.order_id
GROUP BY payment_type
ORDER BY orders DESC;
```

Query Results:

payment_type	orders
credit_card	76795
UPI	19784
voucher	5775
debit_card	1529
not_defined	3

Insights:

Credit cards dominate as the preferred payment method, used by 77% of customers, followed by UPI payments at 19%, with the trend of credit card usage increasing monthly. Vouchers are used by 5% of customers, while debit cards, being the least popular, are used by only 1.5%. This highlights the need to focus on enhancing credit card and UPI payment experiences while exploring strategies to promote alternative payment options like vouchers and debit cards for wider adoption.

Key Insights and Recommendations from the SQL Analysis:

1. Growing E-commerce Trend in Brazil

- **Insight:** The e-commerce market shows consistent growth over time with peaks in May and November, driven by sales events like Black Friday.
- **Recommendation**:
    - Plan major promotional campaigns around May and November.
    - Focus on customer acquisition strategies in non-peak months to ensure sustained growth.

2. Customer Purchase Behavior

- Insight: Most purchases occur in the afternoon and evening, aligning with customer free time or reminders during work hours.
- **Recommendation**:
    - Schedule push notifications and email campaigns in the early afternoon to maximize engagement.
    - Optimize website performance during high-traffic hours to enhance the user experience.

3. Regional Insights

- **Insight**: Sao Paulo (SP) and Rio de Janeiro (RJ) dominate orders due to high populations and purchasing power.
- **Recommendation**:
    - Enhance inventory and logistics in SP and RJ.
    - Launch targeted marketing efforts to penetrate Tier-2 and Tier-3 cities.

4. Delivery Performance

- **Insight**: Delivery times vary significantly, with delays risking customer trust in certain regions.
- **Recommendation**:
    - Invest in better logistics infrastructure in underperforming states to reduce delays.
    - Provide realistic delivery timelines to manage customer expectations effectively.

5. Economic Analysis

- **Insight**: There was a 137.26% increase in order costs from 2017 to 2018, likely due to inflation and rising operational expenses.
- **Recommendation**:
    - Introduce cost-effective product ranges to attract price-sensitive customers.
    - Monitor market trends and adjust pricing strategies accordingly.

6. Freight and Delivery Costs

- **Insight**: States with higher freight costs also experience longer delivery times, indicating inefficiencies.
- **Recommendation**:
    - Optimize supply chain routes and warehouse distribution to reduce freight costs in high-expense regions.

7. Payment Behavior

- **Insight**: Credit cards dominate payment methods (77%), followed by UPI (19%). Vouchers and debit cards are underutilized.
- **Recommendation**:
    - Introduce exclusive offers for UPI users to increase adoption.
    - Promote the use of debit cards and vouchers with special discounts to diversify payment preferences.

8. Seasonal Trends

- **Insight**: Sales peaks are event-driven rather than seasonal.
- **Recommendation**:
    - Experiment with new sales events in non-peak months to identify additional growth opportunities.