

Target-Business Case

About Target:

- Target is a general merchandise retailer with stores in all 50 U.S. states and the District of Columbia.
- 75% of the U.S. population lives within 10 miles of a Target store. **locations**
- We employ 400,000+ **team members**
- Our tagline is "Expect More. Pay Less." We've been using it since 1994!
- Target's CEO since August 2014 is Brian Cornell. **leadership**
- The Target Corporation also owns Shipt and Roundel. More to love!
- Target is headquartered in Minneapolis, Minnesota, its hometown since the first Target store opened in 1962 under The Dayton Company. **history**

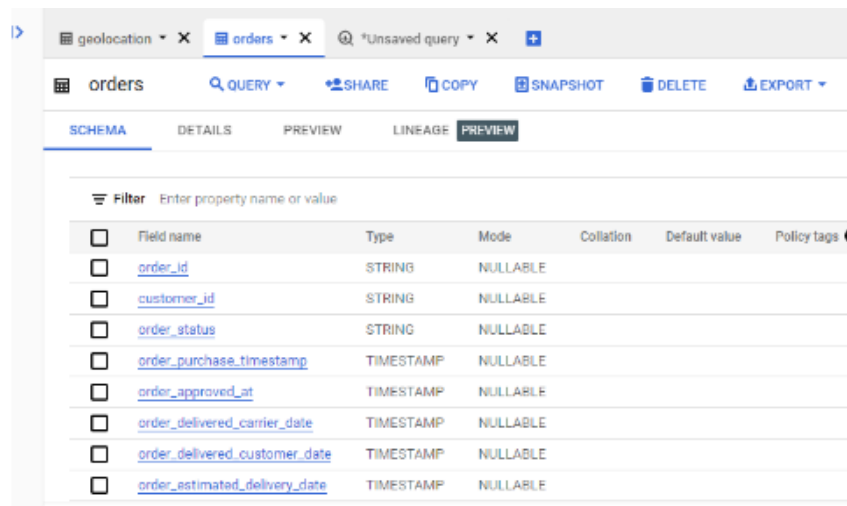
Data:

1. customers.csv
2. geolocation.csv
3. order_items.csv
4. payments.csv
5. reviews.csv
6. orders.csv
7. products.csv
8. sellers.csv

My Findings:

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



The screenshot shows a database interface with a tab for 'orders'. Below the tab, there are buttons for 'QUERY', 'SHARE', 'COPY', 'SNAPSHOT', 'DELETE', and 'EXPORT'. Underneath these buttons are tabs for 'SCHEMA', 'DETAILS', 'PREVIEW', 'LINEAGE', and 'PREVIEW'. The 'SCHEMA' tab is selected, displaying a table with columns: Field name, Type, Mode, Collation, Default value, and Policy tags. The table lists the following columns:

Field name	Type	Mode	Collation	Default value	Policy tags
<input type="checkbox"/> order_id	STRING	NULLABLE			
<input type="checkbox"/> customer_id	STRING	NULLABLE			
<input type="checkbox"/> order_status	STRING	NULLABLE			
<input type="checkbox"/> order_purchase_timestamp	TIMESTAMP	NULLABLE			
<input type="checkbox"/> order_approved_at	TIMESTAMP	NULLABLE			
<input type="checkbox"/> order_delivered_carrier_date	TIMESTAMP	NULLABLE			
<input type="checkbox"/> order_delivered_customer_date	TIMESTAMP	NULLABLE			
<input type="checkbox"/> order_estimated_delivery_date	TIMESTAMP	NULLABLE			

1.2. Time period for which the data is given

```
“select
    min(order_purchase_timestamp) as min,
    max(order_purchase_timestamp) as max
from `Business_Case_Target_SQL.orders`”
```

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS	
Row	//	min	//	max	//
1		2016-09-04 21:15:19 UTC		2018-10-17 17:30:18 UTC	

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

```
“select
    c.customer_city,
    c.customer_state
from `Business_Case_Target_SQL.orders` o
inner join `Business_Case_Target_SQL.customers` c
```

```
on o.customer_id = c.customer_id"
```

Row	customer_city	customer_state
1	maceio	AL
2	aracaju	SE
3	aracaju	SE
4	maceio	AL
5	teresina	PI
6	pau d'arco	AL
7	natal	RN
8	teresina	PI
9	sao joao do piaui	PI

2. In-depth Exploration:

2.1. 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?

```
"select
    extract(year from order_purchase_timestamp) as Year,
    extract(month from order_purchase_timestamp) as
Month,
    count(order_id) as count
from `Business_Case_Target_SQL.orders`
group by Year, Month
order by 1,2"
```

Year	Month	count
2016	9	4
2016	10	324
2016	12	1
2017	1	800
2017	2	1780
2017	3	2682
2017	4	2404
2017	5	3700
2017	6	3245

2017	7	4026
2017	8	4331
2017	9	4285
2017	10	4631
2017	11	7544
2017	12	5673
2018	1	7269
2018	2	6728
2018	3	7211
2018	4	6939
2018	5	6873
2018	6	6167
2018	7	6292
2018	8	6512
2018	9	16
2018	10	4

SUM of count	Month											
Year	1	2	3	4	5	6	7	8	9	10	11	12
2016									4	324		1
2017	800	1780	2682	2404	3700	3245	4026	4331	4285	4631	7544	5673
2018	7269	6728	7211	6939	6873	6167	6292	6512	16	4		

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

```

select
  case
    when extract(hour from
order_purchase_timestamp) > 3 and extract(hour from
order_purchase_timestamp) < 6
      then "Dawn"
    when extract(hour from
order_purchase_timestamp) >= 6 and extract(hour from
order_purchase_timestamp) < 11
      then "Morning"
    when extract(hour from
order_purchase_timestamp) >= 11 and extract(hour from
order_purchase_timestamp) <= 16
      then "Afternoon"

```

```

        when extract(hour from
order_purchase_timestamp) >16 and extract(hour from
order_purchase_timestamp) < 20
        then "Evening"
        else "Night"
        End as Greetings ,
        count(order_id) as count
from `Business_Case_Target_SQL.orders`
group by Greetings
order by 2"

```

Greetings	Count of orders
Dawn	394
Morning	15662
Evening	17901
Night	26695
Afternoon	38789

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

3.1. Get month on month orders by states

```

"select distinct
    C.customer_state,
    extract(month from order_purchase_timestamp) as
Month,
    count(order_id) as Count
from `Business_Case_Target_SQL.orders` O
Inner join `Business_Case_Target_SQL.customers` C
on O.customer_id = C.customer_id
group by 1, 2"

```

MAX of Count	Month											
customer_state	1	2	3	4	5	6	7	8	9	10	11	12
AC	8	6	4	9	10	7	9	7	5	6	5	5
AL	39	39	40	51	46	34	40	34	20	30	26	14

AM	12	16	14	19	19	8	23	9	9	3	10	6
AP	11	4	8	5	11	4	7	5	2	3	4	4
BA	264	273	340	318	368	307	405	323	170	170	250	192
CE	99	101	126	143	136	121	140	130	77	74	108	81
DF	151	196	207	183	208	220	243	232	97	104	168	131
ES	159	186	182	188	228	204	206	200	93	104	170	113
GO	164	176	199	177	226	184	192	213	88	117	157	127
MA	66	67	77	73	65	59	79	70	42	52	56	41
MG	971	1063	1237	1061	1190	1080	1111	1177	511	600	943	691
MS	71	75	79	58	74	76	74	59	33	34	46	36
MT	96	84	71	92	104	83	85	78	35	55	74	50
PA	82	83	109	107	75	92	96	104	41	58	70	58
PB	33	47	55	51	47	51	79	46	29	31	30	37
PE	113	146	153	154	174	140	210	170	76	87	126	103
PI	55	46	48	50	56	43	52	43	23	25	31	23
PR	443	460	504	500	524	478	523	556	183	225	378	271
RJ	990	1176	1302	1172	1321	1128	1288	1307	612	725	1048	783
RN	51	31	52	42	39	49	56	40	24	27	44	30
RO	23	25	29	20	26	22	27	23	16	14	17	11
RR	2	7	8	4	3	8	6		2	4	2	
RS	427	473	569	488	559	526	565	599	279	276	422	283
SC	345	316	362	351	379	321	356	365	157	189	303	193
SE	24	27	43	27	19	37	42	43	16	25	27	20
SP	3351	3357	4047	3967	4632	4104	4381	4982	1648	1908	3012	2357
TO	19	28	28	33	34	26	23	28	17	13	17	14

3.2. Distribution of customers across the states in Brazil

```

select
    customer_state,
    count(customer_id) Count
from `Business_Case_Target_SQL.customers`
group by 1
order by 2 desc

```

customer_state	Count
SP	41746
RJ	12852
MG	11635

RS	5466
PR	5045
SC	3637

4. Impact on Economy: Analyse 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

```

"with cte1 as(
    with cte as
    (
        select *
        from `Business_Case_Target_SQL.orders` O
        full join `Business_Case_Target_SQL.payments` P
        on O.order_id = P.order_id
        where (extract(month from order_purchase_timestamp)
between 1 and 8)
    )
    select
    distinct
    extract(year from order_purchase_timestamp) as Year,
    sum(payment_value) over(partition by extract(year from
order_purchase_timestamp) order by extract(year from
order_purchase_timestamp))as Payments_Total,
    from cte
)
select
    *,
    round(((Payments_Total - lag(Payments_Total)
over(order by Year))/ Payments_Total)*100,2) as Percent
from cte1"

```

Row	Year	Payments_Total	Percent
1	2017	3669022.11...	null
2	2018	8694733.84	57.8

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

```

"select distinct
    customer_state,
    Round(sum(price) over(partition by
customer_state),2) as Sum_Price,
    Round(Avg(price) over(partition by
customer_state),2) as Mean_Price,
    Round(sum(freight_value) over(partition by
customer_state),2) as Sum_freight_value,
    Round(Avg(freight_value) over(partition by
customer_state),2) as Mean_freight_value,
    count(0.order_id) over(partition by customer_state)
count_of_orders
from `Business_Case_Target_SQL.order_items` Oi
full join `Business_Case_Target_SQL.orders` O
on Oi.order_id = O.order_id
full join `Business_Case_Target_SQL.customers` C
on O.customer_id = C.customer_id"

```

customer_state	Sum_Price	Mean_Price	Sum_freight_value	Mean_freight_value	count_of_orders
PA	178947.81	165.69	38699.3	35.83	1085
PB	115268.08	191.48	25719.73	42.72	606
TO	49621.74	157.53	11732.68	37.25	316
RS	750304.02	120.34	135522.74	21.74	6269
CE	227254.71	153.76	48351.59	32.71	1487
RR	7829.43	150.57	2235.19	42.98	52
RJ	1824092.67	125.12	305589.31	20.96	14669
SP	5202955.05	109.65	718723.07	15.15	47820
AL	80314.81	180.89	15914.59	35.84	446

5. Analysis on sales, freight and delivery time

5.1. Calculate days between purchasing, delivering and estimated delivery

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

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

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

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

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

```
"with cte as
(
  select
    *,
    datetime_diff(order_delivered_customer_date,
order_purchase_timestamp, day) as time_to_delivery,
    abs(datetime_diff(order_estimated_delivery_date,
order_delivered_customer_date, day)) as
diff_estimated_delivery
  from `Business_Case_Target_SQL.orders` o
  inner join `Business_Case_Target_SQL.order_items` oi
  on o.order_id = oi.order_id
  inner join `Business_Case_Target_SQL.customers` c
  on o.customer_id = c.customer_id
)
select distinct
  customer_state,
  avg(freight_value) over(partition by customer_state) as
Mean_freight,
  avg(time_to_delivery) over(partition by customer_state)
as Mean_time_to_delivery,
  avg(diff_estimated_delivery) over(partition by
customer_state) as Mean_diff_estimated_delivery
```

```

from cte
order by 3 desc
limit 5"

```

Top five states with highest freight value

customer_state	Mean_freight	Mean_time_to_delivery	Mean_diff_estimated_delivery
RR	42.98442308	27.82608696	25.34782609
AP	34.00609756	27.75308642	24.58024691
AM	33.20539394	25.96319018	20.47239264
AL	35.84367117	23.99297424	12.06088993
PA	35.83268519	23.30170778	16.20208729

6. Payment type analysis:

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

```

"with cte as
(
select
    *,
    extract(year from order_purchase_timestamp) as Year,
    extract(month from order_purchase_timestamp) as Month
from `Business_Case_Target_SQL.orders` O
right join `Business_Case_Target_SQL.payments` P
on O.order_id = P.order_id
)
select
    distinct
    Year,
    Month,
    payment_type,
    count(*) as count
from cte
group by 1,2,3
order by 1,2"

```

Row	Year	Month	payment_type	count
1	2016	9	credit_card	3
2	2016	10	credit_card	254
3	2016	10	voucher	23
4	2016	10	debit_card	2
5	2016	10	UPI	63
6	2016	12	credit_card	1

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

```

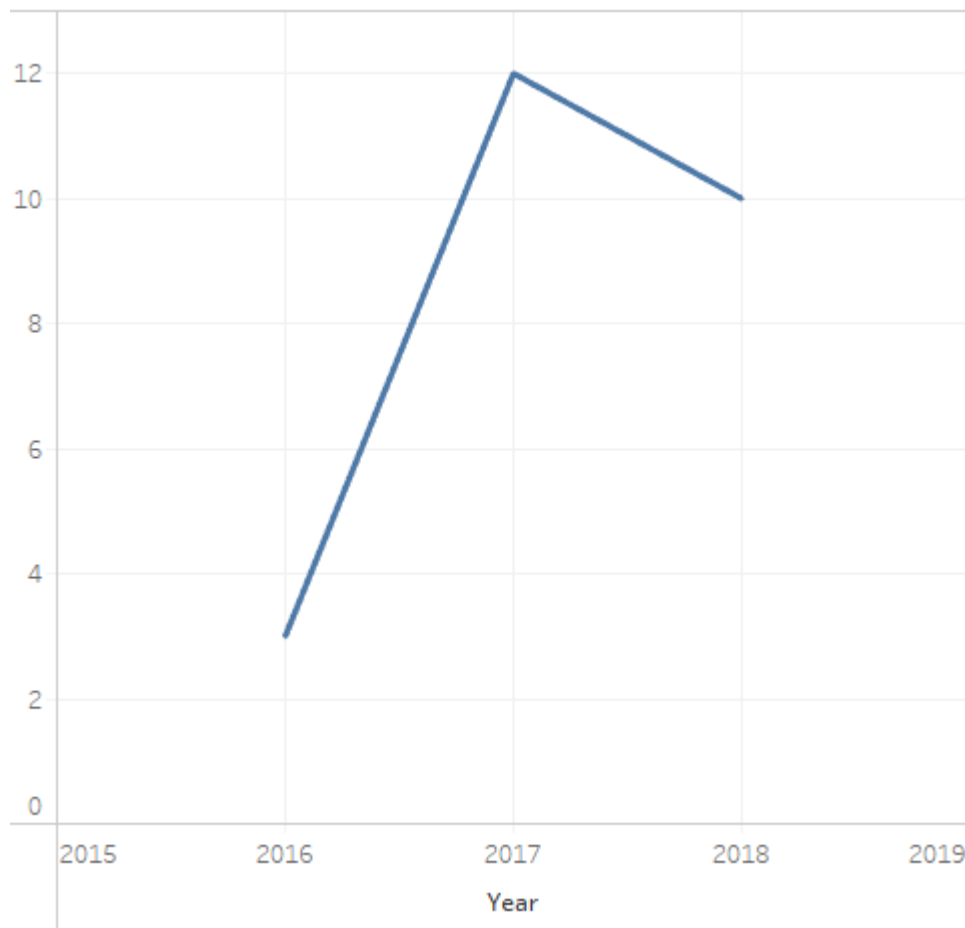
select
    payment_installments,
    count(*) as count
from `Business_Case_Target_SQL.payments`
group by 1

```

payment_installments	count
0	2
1	52546
2	12413
3	10461
4	7098
5	5239
6	3920
7	1626
8	4268
9	644
10	5328
11	23
12	133
13	16
14	15
15	74
16	5

Actionable Insights

1. There is increasing growth of sales from 2017 Feb to 2018 August. We can see the increasing trend from 2017 Feb to 2017 Dec. After that there are predictable decreasing trends. The peak month was November 2017 with an order of 7544. Single digit of orders was ordered in 2018 Oct.

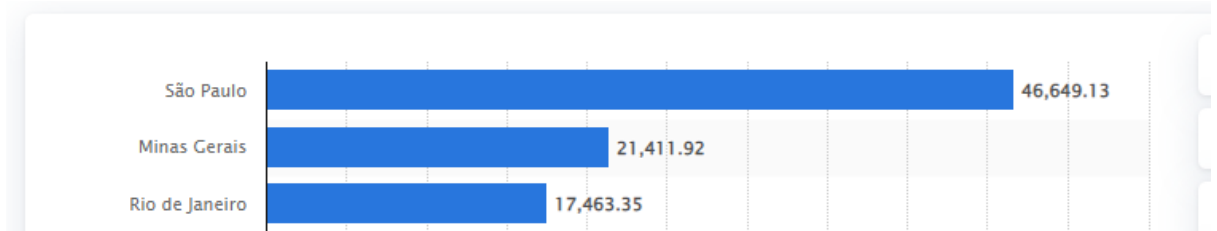


2. Most Brazilians used to buy in Afternoon time, we can say that a large amount of orders were ordered in the time frame of 8 am to 11 pm .
3. The top 3 highest orders placed in a states are Rio de Janeiro, São Paulo, Minas Gerais respectively.

These 3 are the major cities in Brazil, which most of the population occurs in these 3 cities only.

Most number of people = Most count of orders.

Population:



Count of orders:

customer_state	Count
SP	41746
RJ	12852
MG	11635

4. The 57.8% of orders were increased from 2017 to 2018 in the range of Jan to Aug. The orders count increased **2x** from the past year.
5. The mean price was higher in Paraíba and very lower in São Paulo. Similarly the average freight value is higher in Roraima and lower in São Paulo.
6. Target is delivering orders very quickly in São Paulo within 8 days on average. Roraima is the state where the order got delivered very late compared to other states.
7. In São Paulo, Target is delivering **90%** products within the estimated time. More customers are there in São Paulo. But in the same, target performance is not that much fast as compared to são paulo in

roraima , because they are taking extra 3 to 4 days for delivery in those areas.

8. Most Brazilians used to buy in credit cards, upi was second place in that. Mostly they used to buy in one time payments.

Actionable items:

1. We can recommend that if they can be proactive in delivery for the major cities like Rio de Janeiro, São Paulo, Minas Gerais, it will be useful for business, because the average time for delivery in those states are 8 days. As an E-Commerce site they have to reduce the delivery time within 3 to 5 days. We are lagging behind in delivery and we need to improve in delivery timings.
2. From the data, we can see that most of the orders are placed from 8 am to 11 pm. For 100% availability, we need to ensure there is no outage in the website or app. We can add more servers in those timings for avoiding any orders due to any failure from our side.
3. Most orders are ordered through credit card, we can provide more offers for credit cards for increasing the count of orders. Then we can tie up with the credit card bank for improving target business.