Contents

1. Objective

This case study is to analyze the sales performance of a single product in a retail store using business metrics. The goal is to derive meaningful insights that can help the store make data-driven decisions about pricing, promotions, and profitability

2. Tools used for Analysis

- Snowflake
- Power BI

3. Formulas used for the project

- **Daily Sales Price Per Unit** = $\left(\frac{Sales}{Quantity\_Sold}\right)$

- **Average Unit Sales Price** = $\left(\frac{Total\ Sales}{Total\ Quantity\_Sold}\right)$

- **Daily Gross Profit (%)** = $\left(\frac{Sales - Cost\ of\ Sales}{Sales}\right) * 100$

- **Gross Profit Per Unit (%)** = $\left(\frac{Sales\ Per\ Unit - Cost\ of\ Sales\ Per\ Unit}{Sales}\right) * 100$

- **PED** = $\left(\frac{\Delta Quantity\_Sold}{\Delta Price}\right)$

4. Methodology

1. Checking the completeness of the data

1) The number of records

2) Checking data types

The date column has a Text type, so we need to update it.

```
1  SHOW COLUMNS IN "CASESTUDIES_DB"."PUBLIC"."SALES_CASESTUDY1_TBL";
```

Results | Chart

column_name	data_type
DATE	{"type":"TEXT","length":16777216,"byteLength":16777216,"nullable":true,"fixed":fals
SALES	{"type":"FIXED","precision":38,"scale":5,"nullable":true}
COST_OF_SALES	{"type":"FIXED","precision":38,"scale":5,"nullable":true}
QUANTITY_SOLD	{"type":"FIXED","precision":38,"scale":0,"nullable":true}

3) Checking Nulls

Checked the nulls on excel before uploading the data on snowflake

4) Checking duplicates

If Number of rows = Number of Rows without duplicates that means, there are no duplicates

CASESTUDIES_DB.PUBLIC | Settings

```
1  SELECT count(distinct(*)) as Count_without_duplicates from sales_casestudy1_tbl
```

Results | Chart

	COUNT_WITHOUT_DUPLICATES
1	1053

```
CASESTUDIES_DB.PUBLIC    Settings

1  SELECT
2     count(*) as Number_of_Records
3  FROM
4    "CASESTUDIES_DB"."PUBLIC"."SALES_CASESTUDY1_TBL";
```

Results | Chart

	NUMBER_OF_RECORDS
1	1053

2. Creating the Temporary table

Creating the Temp table because I don't want to temper with the main table when I am adding my flagging columns

```
CASESTUDIES_DB.PUBLIC    Settings

1  CREATE TEMPORARY TABLE SALES_CASESTUDY1_TEMP_TBL AS
2  SELECT *
3  FROM SALES_CASESTUDY1_TBL;
4
5
```

Results | Chart

	status
1	Table SALES_CASESTUDY1_TEMP_TBL successfully created.

3. Date Formatting

Changing the format of the date so that I will be able to work with it

```
CASESTUDIES_DB.PUBLIC    Settings

1  UPDATE SALES_CASESTUDY1_TEMP_TBL
2  SET date = to_date(date,'YYYY-MM-DD');
3
4
5
```

Results | Chart

	number of rows updated	number of multi-joined rows updated
1	1053	

4. The Metrics

❖ **Daily Sales Price Per Unit**

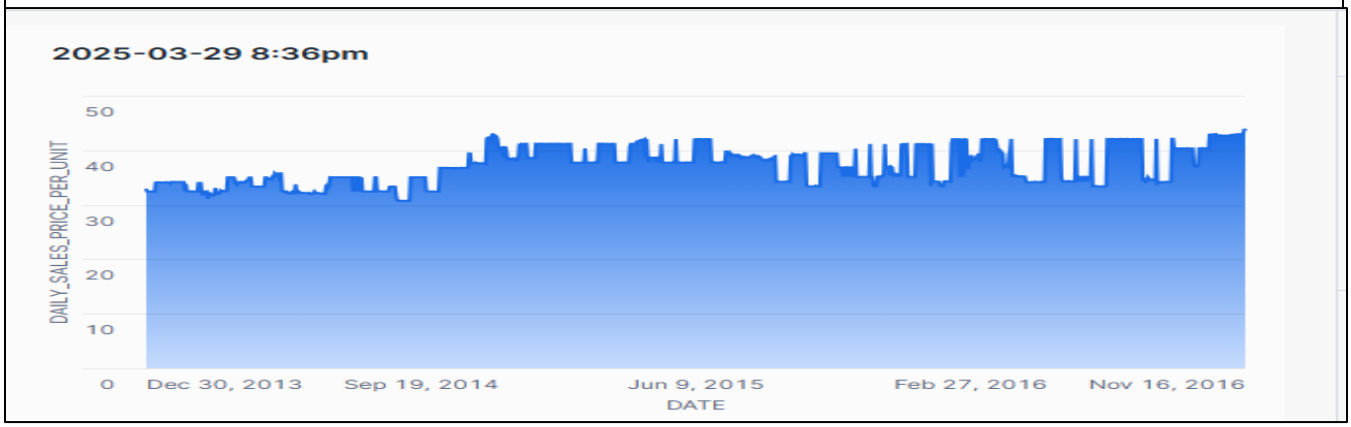
The price at which a single unit of a product is sold on a particular day

$$\textbf{Daily Sales Price Per Unit} = \left(\frac{Sales}{Quantity\_Sold}\right)$$

The daily unit price has been fluctuating because of Promotions, discounts and other external factors however it went up from R32.80 in 2013 to R43.81 in 2016. The upward trend suggests that despite fluctuations, the overall pricing strategy allowed for an increase in unit sales price

```
SELECT
    date,
    ROUND((Sales / Quantity_Sold), 2) AS Daily_Sales_Price_Per_Unit
    FROM sales_casestudy1_temp_tbl

ORDER BY date ASC;
```



Calculating the Daily sales price and showing First 5 and Last 5 records

```
SELECT
    date,
    ROUND((Sales / Quantity_Sold), 2) AS Daily_Sales_Price_Per_Unit
FROM (
    SELECT
        date,
        Sales,
        Quantity_Sold,
        ROW_NUMBER() OVER (ORDER BY date ASC) as rownumber_asc,  -- adding ascending temp marking column
    ROW_NUMBER() OVER (ORDER BY date DESC) as rownumber_desc ---- adding descending temp marking  column
    FROM sales_casestudy1_temp_tbl
)
WHERE rownumber_asc <= 5 OR rownumber_desc <= 5
ORDER BY date ASC;
```

Results | Chart

	DATE	DAILY_SALES_PRICE_PER_UNIT
1	2013-12-30	32.80
2	2013-12-31	32.41
3	2014-01-01	32.41
4	2014-01-02	32.42
5	2014-01-03	32.42
6	2016-11-12	42.93
7	2016-11-13	42.94
8	2016-11-14	42.93
9	2016-11-15	43.79
10	2016-11-16	43.81

❖ Average Unit Sales Price

The average price at which a single unit of a product is sold over a specific period.

$$\textbf{Average Unit Sales Price} = \left(\frac{Total\ Sales}{Total\ Quantity\_Sold}\right)$$

CASESTUDIES_DB.PUBLIC | Settings

```
7  SELECT
8      ROUND(AVG((Sales /Quantity_Sold)), 2) as Average_Unit_Sales_Price
9      FROM sales_casestudy1_temp_tbl
10
11
```

Results | Chart

	AVERAGE_UNIT_SALES_PRICE
1	37.07

CASESTUDIES_DB.PUBLIC | Settings

```
7  SELECT
8      ROUND((sum(Sales) /sum(Quantity_Sold)), 2) as Average_Unit_Sales_Price
9      FROM sales_casestudy1_temp_tbl
10
11
```

Results | Chart

	AVERAGE_UNIT_SALES_PRICE
1	35.40

❖ Daily Gross Profit (%)

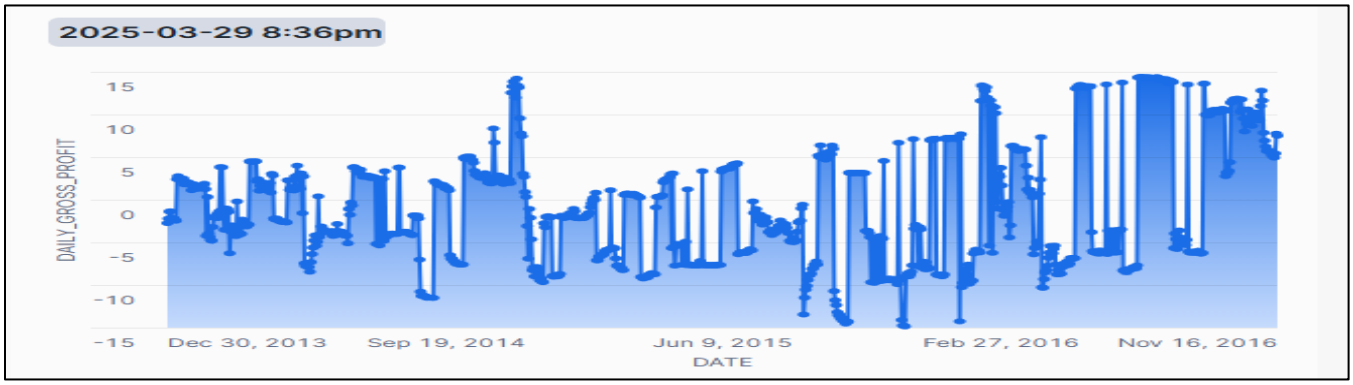
This measures the profitability of sales on a given day. It shows how much of the revenue remains after covering.

Daily Gross Profit (%) = $\left(\frac{Sales - Cost\ of\ Sales}{Sales}\right) * 100$

Daily gross profit varied significantly over the days, even showing a quite number of negative values which is not good for the business. This usually happens due to excessive discounts, cost price increases, or operational inefficiencies.

```
SELECT
    Date,
    ROUND(((Sales - Cost_of_Sales)/sales)*100, 2) as Daily_Gross_Profit

    from sales_casestudy1_temp_tbl
```



Calculating the Daily sales price and showing First 5 and Last 5 records

```
SELECT
    ROUND((sum(Sales) /sum(Quantity_Sold)), 2) as Average_Unit_Sales_Price
    FROM sales_casestudy1_temp_tbl;

--Daily Gross Profit (%): ((Sales - Cost of Sales) / Sales) * 100
SELECT
    Date,
    ROUND(((Sales - Cost_of_Sales)/sales)*100, 2) as Daily_Gross_Profit
    FROM (
    select
        date ,
        Sales,
        Cost_of_Sales,
        ROW_NUMBER() OVER (order by date ASC) as rn_asc,
        ROW_NUMBER() OVER (order by date desc) as rn_desc
    from sales_casestudy1_temp_tbl
    )
    where rn_asc <=5 or rn_desc <=5
    order by date asc
```

	DATE	DAILY_GROSS_PROFIT
1	2013-12-30	-2.74
2	2013-12-31	-2.21
3	2014-01-01	-1.39
4	2014-01-02	-1.35
5	2014-01-03	-2.11
6	2016-11-12	5.29
7	2016-11-13	4.94
8	2016-11-14	5.43
9	2016-11-15	7.75
10	2016-11-16	7.51

❖ Daily Gross Profit Per Unit (%)

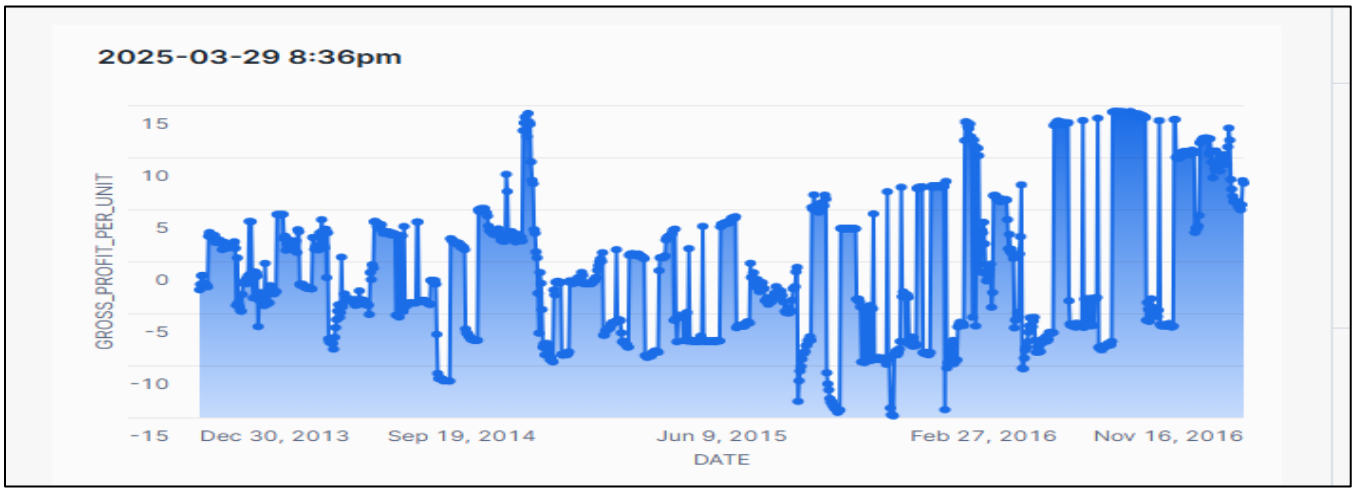
It measures the profit margin made on each unit sold, expressed as a percentage of the selling price.

$$\textbf{Gross Profit Per Unit (\%)} = \left(\frac{Sales\ Per\ Unit - Cost\ of\ Sales\ Per\ Unit}{Sales\ \text{Per Unit}}\right) * 100$$

```
SELECT
    Date,
    (sales/quantity_sold) as Sales_per_unit ,
    (Cost_of_sales/quantity_sold) as Cost_sales_per_unit ,
    ROUND(((Sales_per_unit-Cost_sales_per_unit)/Sales_per_unit)*100, 2) as Gross_Profit_Per_Unit

    from sales_casestudy1_temp_tbl
```

The daily gross profit per unit is the same as Daily gross profit



Calculating the Daily sales price and showing First 5 and Last 5 records

```
SELECT
    Date,
    (sales/quantity_sold) as Sales_per_unit ,
    (Cost_of_sales/quantity_sold) as Cost_sales_per_unit ,
    ROUND(((Sales_per_unit-Cost_sales_per_unit)/Sales_per_unit)*100, 2) as Gross_Profit_Per_Unit
    from( select
            date,
            sales,
            Cost_of_sales,
            Quantity_sold,
            ROW_NUMBER() OVER (order by date asc) as rn_asc,
            ROW_NUMBER() OVER (order by date desc) as rn_desc
    from sales_casestudy1_temp_tbl)
 where rn_asc <= 5 or rn_desc <= 5
 order by date asc
```

❖ **PED** (Price Elasticity of Demand)

It measures how the quantity demanded of a good or service changes in response to a change in its price.

If PED > 1, demand is considered elastic (consumers are highly responsive to price changes).

If PED < 1, demand is considered inelastic (consumers are less responsive to price changes).

If PED = 1, demand is unitary elastic (the percentage change in quantity demanded is equal to the percentage change in price)

```
SELECT
    Date,
    Sales,
    Quantity_sold,
    (sales/quantity_sold) as Sales_per_unit ,
    (Cost_of_sales/quantity_sold) as Cost_sales_per_unit ,
    ROUND(((Sales_per_unit- Cost_sales_per_unit)/Sales_per_unit)*100, 2) as Gross_Profit_Per_Unit,
    ROUND(((Sales - Cost_of_Sales)/sales)*100, 2) as Daily_Gross_Profit
    from sales_casestudy1_temp_tbl
```

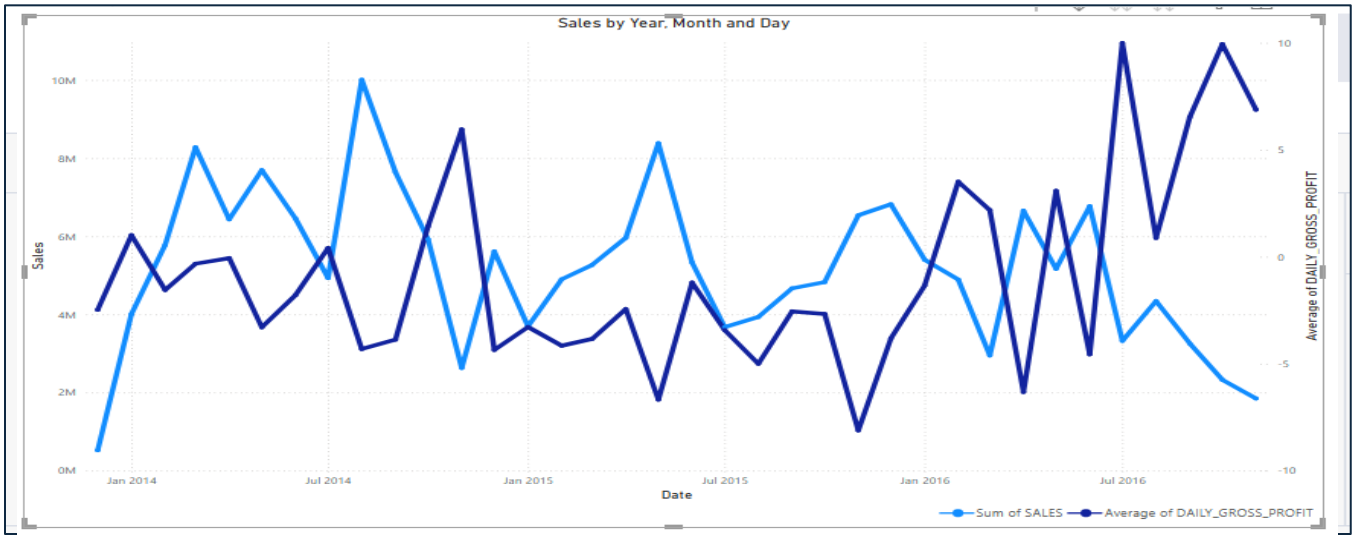
	DATE	SALES_PER_UNIT	COST_SALES_PER_UNIT	GROSS_PROFIT_PER_UNIT
1	2013-12-30	32.80181161564	33.70142390508	-2.74
2	2013-12-31	32.40672039275	33.12323268235	-2.21
3	2014-01-01	32.40570116878	32.85523458178	-1.39
4	2014-01-02	32.41669907692	32.85525144939	-1.35
5	2014-01-03	32.41829375248	33.10256762198	-2.11
6	2016-11-12	42.93490621910	40.66176151444	5.29
7	2016-11-13	42.94028191583	40.82095962736	4.94
8	2016-11-14	42.92974275660	40.60032699413	5.43
9	2016-11-15	43.79144198991	40.39624233379	7.75
10	2016-11-16	43.80599653432	40.51552467385	7.51

$$\text{PED} = \left(\frac{\Delta Quantity\_Sold}{\Delta Price}\right)$$

We have – PED This means that the consumers are less responsive to price changes And We can see that there is a correlation between Sales and quantity sold and few outliers

Periods: May 2014, August 2014, May 2015

We see that Promotions drive higher sales volume but often result in lower profit margins due to discounts. While non-promotional days may see fewer sales, the higher profit margins per unit sold contribute to better overall profitability.



Calculated PED on excel

PED < 1 on these 3 periods, this means that the demand is not sensitive to the price

Year	Month	Average of DAILY_GROSS_PROFIT	Sum of SALES	Sum of QUANTITY_SOLD	Average of SALES_PER_UNIT	CHANGE IN QUANTITY	CHANGE IN PRICE
2014	May	-3.30	7694906.15	235793	33.11	0.24	-0.03
2014	August	-4.31	10010378.29	310572	32.50	1.11	-0.05
2015	May	-6.68	8384367.24	220775	38.12	0.44	-0.05

Period	PED
2014 May PED	-8.638610166
2014 August PED	-22.18426249
2015 May PED	-9.410153525

5. The Insights

Month-over-Month (MoM) and Year-over-Year (YoY) changes help track sales trends and performance over time

$$\textbf{MoM (\%)} = \left(\frac{Current\ Month\ Sales - Previous\ Month\ Sales}{Previous\ Month\ Sales}\right)$$

Calculated the mentioned measures using PowerBI

```
MoM_Sales_Change =
VAR CurrentMonthSales = CALCULATE(SUM('2025-03-29 8_36pm (2)'[SALES]), SAMEPERIODLASTYEAR(DATEADD('2025-03-29 8_36pm (2)'[DATE], -1, MONTH)))
VAR PreviousMonthSales = CALCULATE(SUM('2025-03-29 8_36pm (2)'[SALES]), DATEADD('2025-03-29 8_36pm (2)'[DATE], -1, MONTH))
RETURN
IF(
    NOT ISBLANK(SUM('2025-03-29 8_36pm (2)'[SALES])),
    DIVIDE(
        CurrentMonthSales - PreviousMonthSales,
        PreviousMonthSales
    )
)
```

$$\textbf{YoY (\%)} = \left(\frac{Current\ \text{Year}\ Sales\ \text{for the Month} - Previous\ \text{Year}\ Sales\ \text{for the same Month}}{Previous\ \text{Year}\ Sales\ \text{for the same Month}}\right)$$

```
YoY_Sales_Change =
VAR CurrentYearSales = CALCULATE(SUM('2025-03-29 8_36pm (2)'[SALES]))
VAR PreviousYearSales = CALCULATE(SUM('2025-03-29 8_36pm (2)'[SALES]), SAMEPERIODLASTYEAR('2025-03-29 8_36pm (2)'[DATE]))
RETURN
IF(
    NOT ISBLANK(SUM('2025-03-29 8_36pm (2)'[SALES])),
    DIVIDE(
        CurrentYearSales - PreviousYearSales,
        PreviousYearSales
    )
)
```

MoM is negative but YoY is positive, it might mean long-term growth with short-term fluctuations, The business made a huge sales in 2016 and it had a positive AVG Daily gross profit meaning the business manage to make a profit as it has been losing from 2013-2015