Customers and Products Analysis Using SQL

Author: Dr Lamia Ghozy

Date: 10/11/2025

Project Type: Business Intelligence / Data Analysis

---

1. Introduction

Sales data analysis is a powerful tool to drive smarter business decisions, optimise resources, and increase revenue. This project focuses on analysing the sales records database of a scale model car retailer to answer key business questions.

The project aims to provide actionable insights for:

1. Inventory optimisation – Which products should be ordered more or less?
2. Customer engagement – How to tailor marketing strategies to customer behaviours?
3. Marketing spend – How much can we invest in acquiring new customers?

By leveraging SQL queries, key performance indicators (KPIs), and calculated metrics, we extract data-driven recommendations that can inform business strategy.

---

2. Question 1: Inventory Analysis – Products to Restock

Business Objective:
Identify priority products for restocking to avoid out-of-stock situations while optimising high-selling product availability.

Methodology:

- Compute **low stock rate** for each product: `SUM(quantityOrdered) ÷ quantityInStock`.
- Compute **product performance**: `SUM(quantityOrdered × priceEach)`.
- Identify products that are both low in stock and high-performing.

Results – Priority Products for Restocking:

Product Name	Product Line
1968 Ford Mustang	Classic Cars
1911 Ford Town Car	Vintage Cars
1928 Mercedes-Benz SSK	Vintage Cars
1960 BSA Gold Star DBD34	Motorcycles
1997 BMW F650 ST	Motorcycles
1928 Ford Phaeton Deluxe	Vintage Cars
2002 Yamaha YZR M1	Motorcycles
The Mayflower	Ships
F/A 18 Hornet 1/72	Planes
Pont Yacht	Ships

Key Insight:

Vintage cars and motorcycles are the highest-demand, low-stock items. Prioritising their restocking will prevent lost sales and improve customer satisfaction.

---

3. Question 2: Customer Segmentation – Tailoring Marketing Strategies

Business Objective:
Classify customers into VIP (high-profit) and less-engaged (low-profit) segments to target marketing and loyalty initiatives effectively.

Methodology:

- Calculate profit per customer:
  `Profit = SUM(quantityOrdered × (priceEach – buyPrice))`.
- Identify **VIP customers** (top 5 by profit) and **least-engaged customers** (bottom 5 by profit).
- Include customer contact information for actionable outreach.

Results – VIP Customers:

Last Name	First Name	City	Country	Profit
Freyre	Diego	Madrid	Spain	326,519.66
Nelson	Susan	San Rafael	USA	236,769.39
Young	Jeff	NYC	USA	72,370.09
Ferguson	Peter	Melbourne	Australia	70,311.07
Labrune	Janine	Nantes	France	60,875.30

Results – Least-Engaged Customers:

Last Name	First Name	City	Country	Profit
Young	Mary	Glendale	USA	2,610.87
Taylor	Leslie	Brickhaven	USA	6,586.02
Ricotti	Franco	Milan	Italy	9,532.93
Schmitt	Carine	Nantes	France	10,063.80
Smith	Thomas	London	UK	10,868.04

Key Insight:
VIP customers should be targeted with loyalty programs, while less-engaged customers should be approached with engagement campaigns to increase retention and profitability.

---

4. Question 3: Customer Acquisition – Budgeting Marketing Spend

Business Objective:
Determine the optimal marketing spend for acquiring new customers by calculating the Customer Lifetime Value (LTV).

Methodology:

- Compute **profit per customer** using previous queries.
- Calculate **average profit across all customers** to determine LTV.

Result:

- **Customer Lifetime Value (LTV):** $39,039.59

Key Insight:

The LTV informs how much can be invested in acquiring new customers. For instance, acquiring 10 new customers next month could generate ~$390,395 in future profit, guiding data-driven marketing investment decisions.

---

5. Conclusion

This project demonstrates how SQL can transform raw sales data into actionable business intelligence.

Key Takeaways:

1. **Inventory Optimisation:** Prioritise restocking high-performing, low-stock products.
2. **Customer Segmentation:** Focus loyalty programs on VIPs and engagement campaigns on less-engaged customers.
3. **Marketing Budgeting:** Use LTV to guide customer acquisition investments and forecast revenue.

The combination of SQL skills and business insight showcased here highlights both technical proficiency and strategic thinking—qualities that are highly valuable in data-driven roles.

---

6. Appendix: SQL Queries

```
-- Question 1: Low stock & Product Performance
WITH Low_stock AS (
    SELECT
        p.productCode,
        p.productName,
        ROUND(SUM(o.quantityOrdered)/p.quantityInStock,2) AS low_stock
    FROM products AS p
    JOIN orderdetails AS o
      ON p.productCode = o.productCode
    GROUP BY p.productCode, p.productName
    ORDER BY low_stock DESC
    LIMIT 10
)
SELECT *
FROM Low_stock
WHERE productCode IN (
    SELECT productCode
    FROM (
        SELECT
            p.productCode,
            SUM(o.quantityOrdered * o.priceEach) AS performance
```

```
        FROM products AS p
        JOIN orderdetails AS o
          ON p.productCode = o.productCode
        GROUP BY p.productCode
        ORDER BY performance DESC
        LIMIT 10
    )
);

-- Question 2: VIP and Least-Engaged Customers
WITH customer_profit AS (
    SELECT o.customerNumber, SUM(quantityOrdered * (priceEach - buyPrice)) AS
profit
    FROM products p
    JOIN orderdetails od
      ON p.productCode = od.productCode
    JOIN orders of
      ON o.orderNumber = od.orderNumber
    GROUP BY o.customerNumber
)
SELECT c.contactLastName, c.contactFirstName, c.city, c.country, cp.profit
FROM customers AS c
JOIN customer_profit AS cp
  ON c.customerNumber = cp.customerNumber
WHERE cp.customerNumber IN (
    SELECT customerNumber
    FROM customer_profit
    ORDER BY profit DESC
    LIMIT 5
)
UNION ALL
SELECT c.contactLastName, c.contactFirstName, c.city, c.country, cp.profit
FROM customers AS c
JOIN customer_profit AS cp
  ON c.customerNumber = cp.customerNumber
WHERE cp.customerNumber IN (
    SELECT customerNumber
    FROM customer_profit
    ORDER BY profit ASC
    LIMIT 5
);

-- Question 3: Customer Lifetime Value
WITH customer_profit AS (
    SELECT o.customerNumber, SUM(quantityOrdered * (priceEach - buyPrice)) AS
profit
    FROM products p
    JOIN orderdetails od
      ON p.productCode = od.productCode
    JOIN orders o
      ON o.orderNumber = od.orderNumber
    GROUP BY o.customerNumber
)
SELECT ROUND(AVG(profit),2) AS LTV
FROM customer_profit;
```
