**Pet Supply Store eCommerce Analysis**

-- Total number of sales

SELECT COUNT(\*) FROM sales

A screenshot of a computer

Description automatically generated

-- Top 5 customers (in terms of sales revenue)

SELECT s.customer\_id as customer, SUM(round(sales::numeric,2)) as sales

FROM sales s LEFT JOIN customers c USING (customer\_id)

GROUP BY s.customer\_id

ORDER BY SUM(sales::numeric) DESC

LIMIT 5;

A screenshot of a computer

Description automatically generated

-- Top 5 customers (in terms of # of sales transactions)

SELECT s.customer\_id as customer, COUNT(sales) as no\_of\_sales

FROM sales s LEFT JOIN customers c USING (customer\_id)

GROUP BY s.customer\_id

ORDER BY COUNT(sales) DESC

LIMIT 5;

A screenshot of a computer

Description automatically generated

-- Top 5 products (in terms of sales revenue)

SELECT p.description as product, SUM(sales::numeric) as sales

FROM SALES s JOIN PRODUCTS p USING (stock\_code)

GROUP BY p.description

ORDER BY SUM(sales::numeric) DESC

LIMIT 5;

A screenshot of a computer

Description automatically generated

-- Top 5 products (in terms of # of sales transactions)

SELECT p.description as product, COUNT(sales) as no\_of\_sales

FROM SALES s JOIN PRODUCTS p USING (stock\_code)

GROUP BY p.description

ORDER BY COUNT(sales) DESC

LIMIT 5;

A screenshot of a computer

Description automatically generated

-- What region, state and city had the highest revenue? (Top 5 for state and city)

SELECT region, SUM(sales::numeric) as sales

FROM sales s LEFT JOIN customers c USING (customer\_id)

GROUP BY region

HAVING region IS NOT NULL  --Filtering out guests

ORDER BY SUM(sales) DESC;

Filtering guest customers

A screenshot of a computer

Description automatically generated

Without filtering guest customers, we can see guest customers account for a large number of sales. Guest customers region shows as [null] since we don’t have their information.

A screenshot of a computer

Description automatically generated

SELECT order\_state as state, SUM(sales::numeric) as sales

FROM sales s LEFT JOIN customers c USING (customer\_id)

GROUP BY order\_state

HAVING order\_state IS NOT NULL  --Filtering out guests

ORDER BY SUM(sales) DESC

LIMIT 5;

Filtering guest customers

A screenshot of a computer

Description automatically generated

SELECT order\_city as city, order\_state as state, SUM(round(sales::numeric,2)) as sales

FROM sales s LEFT JOIN customers c USING (customer\_id)

GROUP BY order\_city, order\_state

HAVING order\_city IS NOT NULL  --Filtering out guests

ORDER BY SUM(sales) DESC

LIMIT 5;

Filtering out guest customers

A screenshot of a computer

Description automatically generated

-- Which products got more returns?

SELECT DISTINCT(p.description),

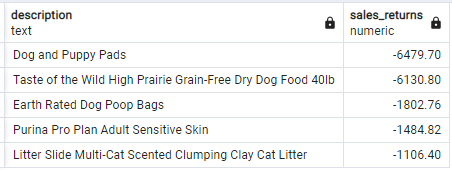
SUM(ROUND(sales::numeric,2)) OVER (PARTITION BY p.description) as sales\_returns

FROM SALES s JOIN PRODUCTS p USING (stock\_code)

WHERE invoice\_no = 'return'

ORDER BY sales\_returns

LIMIT 5;



-- Which days of the month generate more sales?

SELECT day, SUM(sales::numeric) as sales

FROM sales

GROUP BY day

ORDER BY SUM(sales) DESC

LIMIT 5;

A screenshot of a computer

Description automatically generated

-- Which days of the month generate less sales?

SELECT day, SUM(sales::numeric) as sales

FROM sales

GROUP BY day

ORDER BY SUM(sales)

LIMIT 5;

A screenshot of a computer

Description automatically generated

A graph with blue bars

Description automatically generated

A graph with blue lines

Description automatically generated

It seems the days with higher sales are usually in the first half of the month.

-- Which days of the week generate more sales?

SELECT day\_of\_week, SUM(sales::numeric) as sales

FROM sales

GROUP BY day\_of\_week

ORDER BY SUM(sales) DESC

A screenshot of a computer

Description automatically generated

We can see weekdays generate more sales than weekends.

-- Which months generate more sales?

SELECT month, SUM(sales::numeric) as sales

FROM sales

GROUP BY month

ORDER BY SUM(sales) DESC

A screenshot of a calendar

Description automatically generated

We can see Nov/Dec (holiday season) as the months with more sales.

-- Top 5 highest shipping cost products?

SELECT description as product, weight, shipping\_cost\_1000\_mile

FROM products

ORDER BY shipping\_cost\_1000\_mile DESC

LIMIT 5;

A screenshot of a computer

Description automatically generated

-- Correlation between product weight and shipping cost?

SELECT round(corr(weight, shipping\_cost\_1000\_mile)::numeric,2) as weight\_shipping\_corr

FROM products

A screen shot of a website

Description automatically generated

As expected, there is a high correlation between the product’s weight and shipping cost.

Market Basket Analysis

As a last step, we’ll calculate the product association and confidence taking as guide this very useful article: <https://medium.com/@samratjain/explained-market-basket-analysis-using-sql-a7434f30e649>

Confidence basically states what is the probability of buying product\_A given that you bought product\_B and the formula is:

Confidence (A -> B) = Support of A and B / Support of B

Where:

Support of A= Number of transactions A was bought in

Support of A and B= Number of transactions A and B was bought together in

\*Be aware that C(A -> B) is not the same as C(B -> A)

-- Product association and confidence of buying product 1 given customer bought product 2

WITH product\_association AS

(

    SELECT product\_1, product\_2, COUNT(transaction) as transaction\_count

    FROM

    (

        SELECT s1.invoice\_no as transaction, s1.description as product\_1, s2.description as product\_2, s2.invoice\_no

        FROM sales s1, sales s2

        WHERE s1.invoice\_no = s2.invoice\_no

        AND s1.description <> s2.description

        AND s1.description < s2.description

        ORDER BY s1.description

    ) as temp

    GROUP BY product\_1, product\_2

 ),

 transactions\_per\_product AS

 (

    SELECT description as product, COUNT(\*) as no\_of\_transactions

    FROM sales

    GROUP BY description

 )

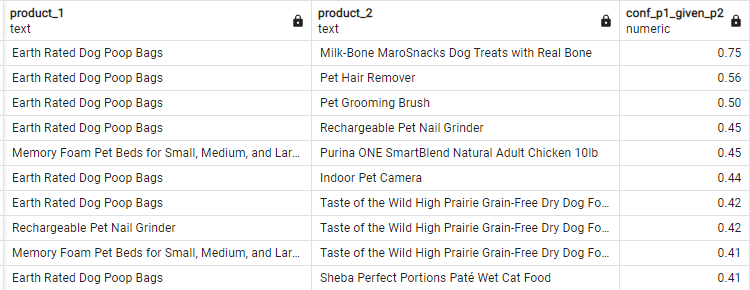
 SELECT product\_1, product\_2,

 round(transaction\_count/no\_of\_transactions::numeric,2) AS conf\_p1\_given\_p2 -- conf(p1->p2) = support of p1 and p2/support of p2

 FROM product\_association pa, transactions\_per\_product tp

 WHERE pa.product\_2 = tp.product

 ORDER BY conf\_p1\_given\_p2 DESC



Now doing from the other direction:

-- Product association and confidence of buying product 2 given customer bought product 1

WITH product\_association AS

(

    SELECT product\_1, product\_2, COUNT(transaction) as transaction\_count

    FROM

    (

        SELECT s1.invoice\_no as transaction, s1.description as product\_1, s2.description as product\_2, s2.invoice\_no

        FROM sales s1, sales s2

        WHERE s1.invoice\_no = s2.invoice\_no

        AND s1.description <> s2.description

        AND s1.description < s2.description

        ORDER BY s1.description

    ) as temp

    GROUP BY product\_1, product\_2

 ),

 transactions\_per\_product AS

 (

    SELECT description as product, COUNT(\*) as no\_of\_transactions

    FROM sales

    GROUP BY description

 )

 SELECT product\_1, product\_2,

 round(transaction\_count/no\_of\_transactions::numeric,2) AS conf\_p2\_given\_p1 -- conf(p2->p1) = support of p1 and p2/support of p1

 FROM product\_association pa, transactions\_per\_product tp

 WHERE pa.product\_1 = tp.product

 ORDER BY conf\_p2\_given\_p1 DESC

A screenshot of a computer

Description automatically generated