Analyzing Customers Transactions using SQL

By/Nagwa Ahmed

Introduction

Customers have purchasing transactions that we shall be monitoring to get intuition behind each customer's behavior to target the customers most efficiently and proactively, to increase sales/revenue, improve customer retention, and decrease churn which will be achieved through investigating customers' behaviors by answering some questions regarding customers, transactions, sales ...etc

Data Analysis

The analysis part consists of some questions to be answered regarding our customers

Q1) What is the number of orders per year?

Here we aim to know whether a specific year has more sales than the other or in other words if our sales are increasing or decreasing over the years

INVOICE_YEAR	NUM_ORDERS
2010	1138
2011	11720

Our guess comes true , it seems that we have started with few orders around 1000 while we achieved about 10 times more orders in the following year which is very promising

Q2) What is the total sales per year?

Was this surge in order associated with an accompanying surge in sales?

INVOICE_YEAR	SALES
2010	13422.96
2011	242295.42

Fortunately, this was true and we have achieved better sales in the following year

Q3) Avg price for product over the years?

Is this success of a result from an increase in prices?

STOCKCODE	2010	2011
17107D	2.55	2.55
20727	1.65	1.65
84970S	0.85	0.85
21240	0.85	0.835555555555555
22554	1.65	1.65
22151		0.42
21363	4.95	4.95
22300	2.55	2.55
22523		0.85
21497		0.42
79066K		0.85
22295	1.45	1.65
22862	4.25	4.25
22645		1.026
21172		1.43
22573	0.85	0.85
23307		0.55

From a quick look at the output, it seems that most of the products have unchanged prices while some didn't exist in 2010 therefore there was an expansion in number of products offered by the owner

Q4) Top 10 products in their sales?

Which products contribute most to this success?

STOCKCODE	SALES	PRODUCT_RANK
84879	4176	1
22191	2662.2	2
84879	1740	3
21787	1519.8	4
84879	1403.6	5
22900	1392	6
22197	1368	7
21430	1220.4	8
21977	1134	9
21479	1125	10

These are the top 10 products that need more attention as they produce the best sales for us

Q5) When was the last time each customer made an order?

This question's goal is to know if customers have recent transaction or not

CUSTOMER_ID	LAST_INVOICE_DATE	TOTAL_SALES
12820	12/6/2011	374.91
12830	11/2/2011	3279.52
12847	11/17/2011	573.69
12853	7/28/2011	989.28
12867	11/13/2011	1179.31
12913	12/5/2011	1663.21
12919	12/1/2011	927.71

In order to go deeper in our analysis , we are going to provide some customer segements based on recency(how recent the last transaction?) , frequency (how many times the customer has bought from our store?) and monetary (how much each customer has paid for our products).

By combining all these measurements , we can divide our customers into the following groups (Champions - Loyal Customers - Potential Loyalists – Recent Customers – Promising - Customers Needing Attention - At Risk - Cant Lose Them – Hibernating – Lost)

After that we have tried answering some other questions using another customer data

Q1) What is the maximum number of consecutive days a customer made purchases?

```
with next_purchase as (
   select cust_id,calendar_dt,row_number() over(partition by cust_id order by
calendar_dt) as ranked_order
   from customers
) , prev_purchase as (
   select cust_id , count(*) over(partition by cust_id , calendar_dt-ranked_order  ) as
consecutive_days
   from next_purchase
)
select cust_id ,max(consecutive_days)
from prev_purchase
group by cust_id
order by cust_id ;
```

Q2) On average, How many days/transactions does it take a customer to reach a spent threshold of 250 L.E?

```
with total_amount  as (
    select cust_id, calendar_dt,sum(amt_le) over(partition by cust_id order by calendar_dt) as
total_spent_amout
    from customers
    order by cust_id
), threshold as (
select cust_id,min(calendar_dt) as start_trans,max(calendar_dt) as last_trans
from total_amount
where total_spent_amout>=250
group by cust_id
)
select avg(last_trans-start_trans) as avg_days_threshold
 from threshold;
```