Ecommerce Application case study

Data Dictionary:

Demographic table

Variable	Description
HOUSEHOLD_KEY	Uniquely identifies each household
AGE_DESC	Estimated age range
MARITAL_STATUS_CODE	Marital Status (A - Married, B- Single, U - Unknown)
INCOME_DESC	Household income
HOMEOWNER_DESC	Homeowner, renter, etc.
HH_COMP_DESC	Household composition
HOUSEHOLD_SIZE_DESC	Size of household up to 5+
KID_CATEGORY_DESC	Number of children present up to 3+

Transaction table:

Variable	Description
HOUSEHOLD_KEY	Uniquely identifies each household
BASKET_ID	Uniquely identifies a purchase occasion
DAY	Day when transaction occurred
PRODUCT_ID	Uniquely identifies each product
QUANTITY	Number of the products purchased during the trip
SALES_VALUE	Amount of dollars retailer receives from sale
STORE_ID	Identifies unique stores
COUPON_MATCH_DISC	Discount applied due to retailer's match of manufacturer coupon
COUPON_DISC	Discount applied due to manufacturer coupon
RETAIL_DISC	Discount applied due to retailer's loyalty card program
TRANS_TIME	Time of day when the transaction occurred
WEEK_NO	Week of the transaction. Ranges 1 - 102

Products Table:

Variable	Description
PRODUCT_ID	Number that uniquely identifies each product
DEPARTMENT	Groups similar products together
COMMODITY_DESC	Groups similar products together at a lower level
SUB_COMMODITY_DESC	Groups similar products together at the lowest level
MANUFACTURER	Code that links products with same manufacturer together
BRAND	Indicates Private or National label brand
CURR_SIZE_OF_PRODUCT	Indicates package size (not available for all products)

Basic Data Analyses:

Total Sales per department

Query :

```
SELECT p.DEPARTMENT,
round(sum((t.SALES_VALUE*t.QUANTITY)-
(t.RETAIL_DISC+t.COUPON_DISC+t.COUPON_MATCH_DISC)),2) AS TOTAL_SALES
FROM `ecommerce.transaction_data` t
JOIN `ecommerce.product` p
 ON t.PRODUCT_ID = p.PRODUCT_ID
GROUP BY p.DEPARTMENT
ORDER BY TOTAL_SALES DESC
limit 10
```

Output :

DEPARTMENT	TOTAL_SALES
KIOSK-GAS	3359893183.0
MISC SALES TRAN	700068796.74
GROCERY	3651525.79
DRUG GM	815374.81
MEAT	525143.89
PRODUCE	421579.77
MEAT-PCKGD	371871.49
DELI	162296.48
PASTRY	93457.01
NUTRITION	88193.74

Average Spending by Household Income Group:

Query:

```
SELECT d.INCOME_DESC, AVG(t.SALES_VALUE) AS AVERAGE_SPENDING
FROM `ecommerce.transaction_data` t
JOIN `ecommerce.hh_demographic` d ON t.household_key = d.household_key
GROUP BY d.INCOME_DESC
ORDER BY AVERAGE_SPENDING DESC
Limit 10
```

Output :

INCOME_DESC	AVERAGE_SPENDING
175-199K	3.681601370823...
250K+	3.679758991009...
200-249K	3.624220812182...
150-174K	3.546047261310...
125-149K	3.418564547643...
100-124K	3.392882546725...
75-99K	3.312785343786...
50-74K	3.138740979130...
35-49K	2.977249179315...
Under 15K	2.956208975568...

Find the number of orders that have small, medium or large order value (small:0-10 dollars, medium:10-20 dollars, large:20+)

Query:

```
select
case WHEN x.order_value BETWEEN 0 AND 10 THEN 'Small'
      WHEN x.order_value BETWEEN 10 AND 20 THEN 'Medium'
      ELSE 'Large'
    END AS order_value_category,
 count(*) order_count,
from
(
select t.BASKET_ID,
sum((t.QUANTITY*t.SALES_VALUE)-
(t.RETAIL_DISC+t.COUPON_DISC+t.COUPON_MATCH_DISC)) as order_value
FROM `ecommerce.transaction_data` t
left join `ecommerce.product` p
on t.PRODUCT_ID = p.PRODUCT_ID
group by t.BASKET_ID ) as x
group by order_value_category
```

Output:

order_value_category	order_count
Large	97409
Small	93087
Medium	42860

Find the number of orders that are small, medium or large order value(small:0-5 dollars, medium:5-10 dollars, large:10+)

Query:

```
select
case WHEN x.order_value BETWEEN 0 AND 5 THEN 'Small'
     WHEN x.order_value BETWEEN 5 AND 10 THEN 'Medium'
     ELSE 'Large'
   END AS order_value_category,
 count(*) order_count,
from
(
select t.BASKET_ID,
sum((t.QUANTITY*t.SALES_VALUE)-
(t.RETAIL_DISC+t.COUPON_DISC+t.COUPON_MATCH_DISC)) as order_value
FROM `ecommerce.transaction_data` t
left join `ecommerce.product` p
on t.PRODUCT_ID = p.PRODUCT_ID
group by t.BASKET_ID ) as x
group by order_value_category
```

Output:

order_value_category	order_count
Large	140269
Small	55823
Medium	37264

Find top 3 stores with highest foot traffic for each week (Foot traffic: number of customers transacting)

Query:

```
select p.STORE_ID,p.WEEK_NO
from (
select x.STORE_ID,x.WEEK_NO,x.Total_Transactions,
row_number() over (partition by WEEK_NO order by Total_Transactions) as rank_order
from (
select STORE_ID, WEEK_NO,count(BASKET_ID) Total_Transactions,
from `ecommerce.transaction_data`
group by STORE_ID, WEEK_NO
order by Total_Transactions desc) as x) as p
where p.rank_order <= 3
order by p.WEEK_NO
```

Output:

STORE_ID	WEEK_NO
286	1
790	1
295	1
3313	2
355	2
311	2
366	3
286	3
370	3
2915	4

Create a basic customer profiling with first, last visit, number of visits, average money spent per visit and total money spent order by highest avg money

Query:

```
select t.household_key,
    MIN(t.DAY) AS first_visit,
    MAX(t.DAY) AS last_visit,
    COUNT(DISTINCT t.BASKET_ID) AS number_of_visits,
    Round(SUM((t.SALES_VALUE*t.QUANTITY)-
(t.RETAIL_DISC+t.COUPON_DISC+t.COUPON_MATCH_DISC)) / COUNT(DISTINCT
t.BASKET_ID),2) AS avg_money_spent_per_visit,
    Round(SUM((t.SALES_VALUE*t.QUANTITY)-
(t.RETAIL_DISC+t.COUPON_DISC+t.COUPON_MATCH_DISC)),2) AS total_money_spent
from `ecommerce.transaction_data` t
group by household_key
ORDER BY
    avg_money_spent_per_visit DESC
```

Output:

household_key	first_visit	last_visit	number_of_visits	avg_money_spent_pe	total_money_spent
1727	109	118	2	1223390.73	2446781.46
2219	80	702	12	517007.5	6204090.04
755	36	709	201	359618.73	72283364.5
556	28	668	13	320927.77	4172061.07
218	109	711	145	281253.57	40781768.03
777	82	707	36	240131.14	8644721.15
1701	81	684	28	235526.87	6594752.44
2115	75	711	252	230390.76	58058471.64
1800	54	709	65	227885.11	14812532.32
1023	107	710	422	224386.83	94691240.76

Single customer analysis selecting most spending customer for whom we have demographic information(because not all customers in transaction data are present in demographic table)(show the demographic as well as total spent)

Query:

```
with CTE as (
select t.household_key,
round(sum((t.SALES_VALUE*t.QUANTITY)-
(t.RETAIL_DISC+t.COUPON_DISC+t.COUPON_MATCH_DISC)),2) as total_amount
from `ecommerce.transaction_data` t
inner join `ecommerce.hh_demographic` d
on t.household_key = d.household_key
group by t.household_key
order by total_amount desc)
select c.household_key,c.total_amount,
h.AGE_DESC,h.MARITAL_STATUS_CODE,h.INCOME_DESC,h.HOMEOWNER_DESC
from CTE c
inner join `ecommerce.hh_demographic` h
on c.household_key = h.household_key
limit 1
```

Output:

household_key	total_amount	AGE_DESC	MARITAL_STATUS_CODE	INCOME_DESC	HOMEOWNER_DESC
7	2371.29	45-54	A	50-74K	Homeowner

Find products(product table : SUB_COMMODITY_DESC) which are most frequently bought together and the count of each combination bought together. do not print a combination twice (A-B / B-A)

Query:

```
with cte1 as (select
        t.product_id,
        t.BASKET_ID,
        p.SUB_COMMODITY_DESC
from `ecommerce.transaction_data` t join `ecommerce.product` p using(product_id)),
    cte2 as (select
              distinct
             a.BASKET_ID,
             case
                 when a.SUB_COMMODITY_DESC < b.SUB_COMMODITY_DESC
then a.SUB_COMMODITY_DESC
                 else b.SUB_COMMODITY_DESC end as product1,
              case
                  when a.SUB_COMMODITY_DESC < b.SUB_COMMODITY_DESC
then b.SUB_COMMODITY_DESC
                   else a.SUB_COMMODITY_DESC end as product2
             from cte1 a  join cte1 b on a.BASKET_ID = b.BASKET_ID
             where a.SUB_COMMODITY_DESC <> b.SUB_COMMODITY_DESC)
select
    product1,
    product2,
    count(*) as combination_count
from cte2
group by product1, product2
order by combination_count desc;
```

Outpt:

product1	product2	combination_count
BANANAS	FLUID MILK WHITE ONLY	4131
FLUID MILK WHITE ONLY	MAINSTREAM WHITE BREAD	3753
FLUID MILK WHITE ONLY	SOFT DRINKS 12/18&15PK CA...	3328
FLUID MILK WHITE ONLY	SHREDDED CHEESE	3155
FLUID MILK WHITE ONLY	YOGURT NOT MULTI-PACKS	2805
DAIRY CASE 100% PURE JUICE...	FLUID MILK WHITE ONLY	2682
FLUID MILK WHITE ONLY	SFT DRNK 2 LITER BTL CARB I...	2579
FLUID MILK WHITE ONLY	KIDS CEREAL	2554
FLUID MILK WHITE ONLY	POTATO CHIPS	2200
EGGS - LARGE	FLUID MILK WHITE ONLY	1952

Weekly change in Revenue Per Account (RPA) (difference in spending by each customer compared to last week)(use lag function)

Query:

```
With CTE as
(
Select t.household_key, t.WEEK_NO,
round(sum((t.SALES_VALUE*t.QUANTITY)-
(t.RETAIL_DISC+t.COUPON_DISC+t.COUPON_MATCH_DISC)),2) as Total_revenue
from `ecommerce.transaction_data` t
group by t.household_key, t.WEEK_NO
order by t.household_key, t.WEEK_NO
)
select x.household_key, x.WEEK_NO,x.Total_revenue,
lag(x.Total_revenue) over (partition by x.household_key order by x.WEEK_NO) as pre_revenue,
Round((x.Total_revenue - (lag(x.Total_revenue) over (partition by x.household_key order by
x.WEEK_NO))),2) Revenue_change
from CTE x
order by x.household_key, x.WEEK_NO
```

Output:

household_key	WEEK_NO	Total_revenue	pre_revenue	Revenue_change
1	8	55.35	*null*	*null*
1	10	27.88	55.35	-27.47
1	13	15.44	27.88	-12.44
1	14	38.26	15.44	22.82
1	15	15.31	38.26	-22.95
1	16	19.85	15.31	4.54
1	17	22.16	19.85	2.31
1	19	53.51	22.16	31.35
1	20	60.24	53.51	6.73
1	22	52.82	60.24	-7.42

Recommendations and Insights:

- For customers whose spending has decreased, consider sending re-engagement offers, such as discounts or personalized messages, to encourage them to return.
- For customers with increased spending, send thank-you notes, reward points, or personalized product recommendations to reinforce the positive trend.
- Ensure sufficient stock for top-performing products to avoid stockouts, especially during peak seasons.
- Allocate more marketing resources to promote top-performing products and departments, while also considering strategies to boost awareness and sales of less popular items.
- Develop marketing campaigns tailored to different demographic groups. For example, families with children might be more interested in promotions for bulk purchases or children's products.
- Offer products that cater specifically to the needs of different demographic groups, such as age-specific items or products targeted at certain income brackets.