2024

E-commerce data analysis

USING SQL

JAYANTI RASTOGI

STRUCTURE AND CHARACTERISTICS OF DATASET

Data type of all columns for given tables.

Query walkthrough:

We need to find out the datatypes of various fields that are in our dataset. Understanding the format in which data has been stored, will help us to plan how to analyse and gain insights from it. Here we are using Information Schema for the purpose.

Query: select
column_name, data_type
from project.INFORMATION_SCHEMA.COLUMNS
where table_name='demographics';

Column and their respective datatypes for table “demographics”:

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	column_name	data_type
1	AGE_DESC	STRING
2	MARITAL_STATUS_CODE	STRING
3	INCOME_DESC	STRING
4	HOMEOWNER_DESC	STRING
5	HH_COMP_DESC	STRING
6	HOUSEHOLD_SIZE_DESC	STRING
7	KID_CATEGORY_DESC	STRING
8	household_key	INT64

Query: select
column_name, data_type
from project.INFORMATION_SCHEMA.COLUMNS
where table_name='product';

Column and their respective datatypes for table “product”:

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	column_name	data_type
1	PRODUCT_ID	INT64
2	MANUFACTURER	INT64
3	DEPARTMENT	STRING
4	BRAND	STRING
5	COMMODITY_DESC	STRING
6	SUB_COMMODITY_DESC	STRING
7	CURR_SIZE_OF_PRODUCT	STRING

Query: select
column_name, data_type
from project.INFORMATION_SCHEMA.COLUMNS
where table_name='transactions';

Column and their respective datatypes for table “transactions”:

JOB INFORMATION | RESULTS | CHART | JSON

Row	column_name	data_type
4	DAY	INT64
5	PRODUCT_ID	INT64
6	QUANTITY	INT64
7	SALES_VALUE	FLOAT64
8	STORE_ID	INT64
9	RETAIL_DISC	FLOAT64
10	TRANS_TIME	INT64
11	WEEK_NO	INT64
12	COUPON_DISC	FLOAT64
13	COUPON_MATCH_DISC	FLOAT64

Insights:

- Household_key is primary key for demographics. It is foreign key in transactions table.
- Product_id is primary key in product table and serves foreign in transactions table.
- Trans_time in transactions table should in date time format.
- There should be proper Date column in transactions table to analyse month on month sales.

Question 1: *Find the number of orders that have small, medium or large order value (small:0-10 dollars, medium:10-20 dollars, large:20+).*

Query walkthrough:

We are required to find the count of orders for 3 categories: small, medium and large. As these categories are not predefined, we cannot directly apply count on these categories. To do this, we will be identifying if the order sales_value (sum of sales_value over basket_id) is within defined range of categories with help of case statements and then we will be using sum to find the total order count for the categories.

Query:
```
with cte as (select basket_id,
                    sum(sales_value) as sales_value
             from `project.transactions`
             group by BASKET_ID)

select
 sum(case when SALES_VALUE <10 then 1 else 0 end) AS small,
 sum(case when SALES_VALUE >=10 and SALES_VALUE<20 then 1 else 0 end)
       AS medium,
 sum(case when SALES_VALUE >=20 then 1 else 0 end) AS large
from cte;
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	small	medium	large
1	115045	49892	68419

Insights:

- Most of orders (approx. 50%) are below $10.
- 30% of orders have high sales_value (above $20 and more).

Question 2: *Find the number of orders that are small, medium or large order value (small:0-5 dollars, medium:5-10 dollars, large:10+)*

Query walkthrough:

We are required to find the count of orders for 3 categories: small, medium and large. This is same what we have done before, and the approach is also same. We are only redistributing the range to get a more meaningful insights for SALES_VALUE.

Query:

```
with cte as (select basket_id,
                    sum(sales_value) as sales_value
             from `project.transactions`
             group by BASKET_ID)

select
 sum(case when SALES_VALUE <5 then 1 else 0 end) AS small,
 sum(case when SALES_VALUE >=5 and SALES_VALUE<10 then 1 else 0 end)
       AS medium,
 sum(case when SALES_VALUE >=10 then 1 else 0 end) AS large
from cte;
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	small	medium	large
1	69752	45293	118311

Insights:

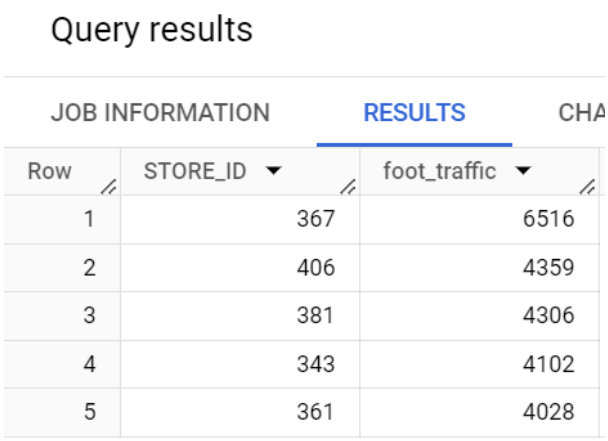
- After redefining the range for SALES_VALUE, we can see that for most orders (60% of 115045) in before considered range of below $10 have a total value of $5 or below.
- Orders with SALES_VALUE between $5 and $10 are only 19% of total orders.
- 50% of orders have SALES_VALUE above $10.

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

Query walkthrough:

We will calculating the total number of orders for every store. This will help us the identifying the busiest stores and hence identifying the stores will generate more revenue.

Query:
```
select
    STORE_ID,
    count(distinct basket_id) as foot_traffic
from project.transactions
group by STORE_ID
order by foot_traffic desc
limit 5 ;
```



Row	STORE_ID	foot_traffic
1	367	6516
2	406	4359
3	381	4306
4	343	4102
5	361	4028

Insights:

- STORE_ID 367, 406 and 381 are top 3 busiest stores.
- The foot traffic at STORE_ID 367 is quite a bit higher than at other stores.
- High foot traffic indicates that many customers are visiting and engaging with the business. It suggests that the business is attracting people, whether through marketing efforts, location advantages, or other factors.

Question 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 average money.*

Query walkthrough:

To obtain the first visit, last visit, number of visits, average money spent per visit, and total money spent by each customer, the query groups the transactions table by household_key. The first visit and last visit are determined by combining trans_time and day to accurately capture the earliest and latest transaction times.

Query:
```
with cte1 as (select
        household_key,
        basket_id,
        SALES_VALUE,
```

```
        (LPAD(CAST(DAY AS STRING), 3, '0')) as day,
        LPAD(CAST(TRANS_TIME AS STRING), 4, '0') as trans_time
        from project.transactions
        ),
    cte2 as (select household_key,basket_id,
        sum(sales_value) as sales_value
        from cte1
        group by household_key,basket_id),
    cte3 as (select
        household_key,
        count(distinct basket_id) as number_of_visits,
        min(concat("day ",day,"_",left(trans_time,2),":",right(trans_time,2)))
          as first_visit,
        max(concat("day " ,day,"_",left(trans_time,2),":",right(trans_time,2)))
          as last_visit,
        from cte1
        group by household_key),
     cte4 as (select household_key,
          round(sum(SALES_VALUE),3)as total_sales,
          round(avg(SALES_VALUE),3) as avg_sales
          from cte2
          group by household_key)

select c1.* , c2.total_sales,c2.avg_sales from cte3 c1 join cte4 c2
                using(household_key)
order by avg_sales desc
limit 10
```

Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	household_key	number_of_visits	first_visit	last_visit	total_sales	avg_sales
1	2042	26	day 052_18:42	day 683_16:18	2339.21	89.97
2	973	80	day 095_21:28	day 710_20:52	6875.89	85.949
3	1899	69	day 020_13:59	day 705_09:57	5789.59	83.907
4	1900	55	day 111_14:16	day 707_13:18	4227.72	76.868
5	1574	27	day 107_11:37	day 651_14:37	1843.3	68.27
6	1315	5	day 060_22:21	day 624_16:36	317.39	63.478
7	2479	111	day 111_09:22	day 706_18:12	6954.64	62.654
8	931	40	day 094_12:45	day 668_18:42	2455.29	61.382
9	1344	26	day 087_15:38	day 691_17:22	1570.37	60.399
10	248	53	day 029_14:15	day 704_16:34	3090.89	58.319

Insights:

- The customer with household_key 2042 has an average purchase amount of $90, which is higher than that of any other customer.
- The customer with household_key 1315 has a comparative average purchase amount but did not visit the store regularly. This shows need for customer engagement.
- The customer with household_key 2479 has made over 100 visits, with an average purchase amount of $62. The total sales for this customer are over $7000.

Question 5: *Do a 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 walkthrough:

In this query, we first group the data by household_key to calculate the total purchase amount for each customer. We then use a LEFT JOIN to include customers who are the highest spenders, even if their demographics information is missing. This approach ensures that all high-spending customers are considered, while an INNER JOIN would exclude those without available demographics information.

Query:

```
with cte1 as (select
                household_key,
                round(sum(SALES_VALUE),3)as total_sales,
                from project.transactions
                group by household_key
                )

select c.household_key,c.total_sales, d.*
from cte1 c left join project.demographics d
using(household_key)
order by c.total_sales desc
limit 2;
```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	household_key	total_sales	AGE_DESC	MARITAL_STATUS	INCOME_DESC	HOMEOWNER_DESC	HH_COMP_DESC	HOUSEHOLD_SI	KID_CATEGORY_	household_key
1	1023	18901.09	null	null	null	null	null	null	null	null
2	1609	13804.38	45-54	A	125-149K	Homeowner	2 Adults Kids	5+	3+	1609

Insights:

- The customer with household_key 1023 has the highest purchase amount, totaling over $18,000 for the considered period. However, demographic details for this customer are not available.
- The customer with household_key 1609 has the second highest purchase amount, totaling $13,000. The demographic information indicates that this customer is a middle-aged couple with a large household size (5+) and high income.

Question 6: *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 walkthrough:

We start by grouping the transactions and product information. In the first CTE, we extract product_id, BASKET_ID, and product names (SUB_COMMODITY_DESC). The second CTE generates all unique pairs of products within each basket, ensuring each combination is represented consistently. Finally, we count the occurrences of each product pair, group by the pair, and order the results by the count to highlight the most frequently bought combinations.

Query:
```
with cte1 as (select
                    t.product_id,
                    t.BASKET_ID,
                    p.SUB_COMMODITY_DESC
                        from project.transactions t join project.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;
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS

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

Insights:

- Bananas and Fluid Milk White are most frequently bought together combination.
- Fluid Milk White seems to be most bought product as it is appearing in all ten top combinations.

Question 7: *Find the weekly change in Revenue Per Account (RPA) (difference in spending by each customer compared to last week)(use lag function)*

Query walkthrough:

To calculate the weekly change in Revenue Per Account (RPA), first the weekly revenue for each customer (cte1) are summed for every customer. Then, it uses the LAG function to get the previous week's revenue and compute the difference from the current week's revenue (cte2). The results are selected and ordered by household_key.

Query:

```
with cte1 as (select
                household_key,WEEK_NO,
              round(sum(SALES_VALUE),2) as week_revenue
            from project.transactions
            group by household_key,WEEK_NO
            order by household_key,WEEK_NO),
cte2 as (select *,
            (round(lag(week_revenue,1)over(partition by
                household_key order by week_no),2))as prev_week_revenue,
                round(week_revenue- lag(week_revenue,1)over(partition by
                household_key order by week_no),2) as revenue_diff
 from cte1)

select *
from cte2
order by household_key
```

Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTI

Row	household_key	WEEK_NO	week_revenue	prev_week_revenue	revenue_diff
1	1	8	42.58	null	null
2	1	10	14.01	42.58	-28.57
3	1	13	14.03	14.01	0.02
4	1	14	25.71	14.03	11.68
5	1	15	10.98	25.71	-14.73
6	1	16	9.09	10.98	-1.89
7	1	17	13.98	9.09	4.89
8	1	19	47.35	13.98	33.37
9	1	20	31.77	47.35	-15.58
10	1	22	38.98	31.77	7.21

Insights:

- Customers do not make purchase every week.
- Weekly revenue also varies.

Additional questions:

Question 8. *For each product, calculate the total quantity sold each week, and determine the top 5 brands in terms of sales quantity.*

Query walkthrough:

The query groups the data by product_id and week_no to calculate the total quantity sold each week for each product. It then orders the results by the total units sold in descending order and by week_no to identify the top-selling products each week.

Query:

```
select
  product_id,week_no,
  sum(quantity) as total_units
from project.transactions
group by product_id,week_no
order by total_units desc, week_no;
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	product_id	week_no	total_units
1	6534178	76	1639672
2	6534178	22	1456713
3	6534178	98	1444271
4	6534178	74	1438054
5	6534178	69	1380032
6	6534178	88	1379704
7	6534178	58	1368262
8	6534178	78	1360031
9	6534178	101	1351866
10	6534178	87	1351690

Insights:

- It seems product_id 6534178 is most bought product. This product is GASOLINE REG UNLEADED.

For 2nd part:

Query walkthrough:

To get top 5 products names, the query first calculates the total quantity sold for each product by grouping the transactions by product_id in a Common Table Expression (CTE). Then, it joins this result with the product table to sum the total units sold for each product description (SUB_COMMODITY_DESC). Finally, it orders the products by the total units sold in descending order.

Query:

```
with cte1 as (select
                product_id,sum(quantity) as total_units
              from project.transactions
              group by product_id
```

```
)

select p.SUB_COMMODITY_DESC, sum(c.total_units) as total_units
from cte1 c join project.product p using (product_id)
group by p.SUB_COMMODITY_DESC
order by total_units desc
```

Query results

JOB INFORMATION | RESULTS | CHART | JSO

Row	SUB_COMMODITY_DESC	total_units
1	GASOLINE-REG UNLEADED	128146567
2	FLUID MILK WHITE ONLY	45613
3	YOGURT NOT MULTI-PACKS	32845
4	SOFT DRINKS 12/18&15PK CA...	27527
5	SFT DRNK 2 LITER BTL CARB I...	27216
6	CANDY BARS (SINGLES)(INCL...	23222
7	SHREDDED CHEESE	17843
8	MAINSTREAM WHITE BREAD	17437
9	CONDENSED SOUP	16601
10	BANANAS	15841

Insights:

- As seen above, Gasoline is most bought product.
- Milk is second most product which is quite relatable as milk with other products top the product combinations.

Question 9: *Investigate the peak transaction times by hour of the week. Calculate the average revenue and total revenue generated during these peak times.*

Query walkthrough:

We first extracts the hour from the trans_time field and calculates the sales value for each transaction in a Common Table Expression (CTE). Then, it is grouped by hour, calculating foot traffic (count of transactions), total sales, and average sales per hour. Finally, the results are ordered by foot traffic in descending order to identify the busiest hours.

Query:

```
with cte1 as (select
                sales_value,
                trans_time,
                basket_id,
                left(lpad(cast(trans_time as string),4,"0"),2) as  hour
                from project.transactions
                ),
     cte2 as (select
                basket_id, hour,
                round(sum(sales_value),2) as sales_value
                from cte1
                group by basket_id,hour ),
     cte3 as (select
```

```
                    hour,
                    count(distinct basket_id) as foot_traffic,
                    round(sum(sales_value),2) as total_sales,
                    round(avg(sales_value),2) as avg_sales
                    from cte2
                    group by hour
                    order by foot_traffic desc)
select * from cte3
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS

Row	hour	foot_traffic	total_sales	avg_sales
1	17	23530	413181.65	17.56
2	18	21961	381367.2	17.37
3	16	21222	374040.08	17.63
4	19	18632	321471.08	17.25
5	15	18445	336488.32	18.24
6	14	16845	319910.44	18.99
7	13	16465	315762.43	19.18
8	12	15680	295302.94	18.83
9	20	15026	247677.19	16.48
10	11	13545	250375.01	18.48

Insights:

- Foot traffic is notably higher between 4 pm and 6 pm compared to the rest of the day.
- The busiest hour is from 5 pm to 6 pm, where both foot traffic and revenue peak.
- Average revenue tends to be higher between 12 pm and 4 pm.

Question 10: *Determine how household characteristics (such as the number of children and homeownership) affect the total revenue generated.*

```
with cte1 as (select household_key,
            basket_id,
            sum(sales_value) as sales_value
            from project.transactions
            group by household_key,basket_id ),
    cte2 as (select p.household_key,
              p.homeowner_desc,
              p.kid_category_desc,
              t.sales_value
              from project.demographics p join cte1 t using(household_key)),

    cte3 as (select
            homeowner_desc,
            round(avg(sales_value),2) as household_avg_sale,
            round(sum(sales_value),2) as total_sales
```

```
        from cte2
        group by homeowner_desc),
    cte4 as (select
        kid_category_desc,
        round(avg(sales_value),2) as household_avg_sale,
        round(sum(sales_value),2) as total_sales
        from cte2
        group by kid_category_desc)

select * from cte3
order by household_avg_sale;

select * from cte4
order by household_avg_sale;
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTI

Row	kid_category_desc	household_avg_sale	total_sales
1	None/Unknown	17.89	1477240.38
2	1	19.58	336888.57
3	2	21.59	203033.72
4	3+	23.78	232914.18

Insights:

- Average sales are higher for household with more than 3 kids.
- As the kids increases in household, it is likely to get more sales from the household.

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTI

Row	homeowner_desc	household_avg_sale	total_sales
1	Probable Renter	11.36	23671.22
2	Probable Owner	14.5	27236.49
3	Renter	15.65	118735.77
4	Unknown	15.89	561266.78
5	Homeowner	21.07	1519166.59

Insights:

- Average as well as total sales are higher for household with home ownership.
- Homeowners have highest average sales as $21.

Question 11: *Calculate the year-over-year sales growth for each household, segmented by income and marital status.*

Query walkthrough:

First we aggregate annual sales data per basket and household in cte1, joins it with demographic details in cte2, and sums total sales by household, year, marital status, and income. It then calculates average and total sales by income and marital status for each year in cte3, reshapes the data to compare sales between two years in cte4, and formats marital status codes while computing growth percentages in cte5. The final results are ordered by income description.

Query:

```
with cte1 as (select
                household_key,
                basket_id,
                ceil(day/365) as year,
                sum(sales_value) as sales_value
   from `project.transactions`
   group by household_key, basket_id, year
),

cte2 as (
   select
      c.household_key,
      c.year,
      sum(c.sales_value) as total_sales,
      d.marital_status_code,
      d.income_desc
   from cte1 c
   join project.demographics d using (household_key)
   group by c.household_key, c.year, d.marital_status_code, d.income_desc
),

cte3 as (
   select
      income_desc,
      year,
      marital_status_code,
      round(avg(total_sales), 2) as avg_sales,
      round(sum(total_sales), 2) as total_sales
   from cte2
   group by income_desc, year, marital_status_code
),

cte4 as (
   select
      income_desc,
      marital_status_code,
      max(case when year = 1 then avg_sales end) as avg_sales_year1,
      max(case when year = 2 then avg_sales end) as avg_sales_year2,
      max(case when year = 1 then total_sales end) as total_sales_year1,
      max(case when year = 2 then total_sales end) as total_sales_year2
   from cte3
   group by income_desc, marital_status_code
),
```

```
cte5 as (
   select
      income_desc,
      case when marital_status_code="a" then "married"
           when marital_status_code="b" then "single"
           else "unknown"
           end as marital_status_code,
      avg_sales_year1,
      avg_sales_year2,
      total_sales_year1,
      total_sales_year2,
      round(((avg_sales_year2 - avg_sales_year1) / avg_sales_year1) * 100, 2) as
avg_sales_growth_pct,
      round(((total_sales_year2 - total_sales_year1) / total_sales_year1) * 100, 2) as
total_sales_growth_pct
   from cte4 )

select *
from cte5
order by income_desc;
```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	income_desc	marital_status_code	avg_sales_year1	avg_sales_year2	total_sales_year1	total_sales_year2	avg_sales_growth_pct	total_sales_growth_pct
1	100-124K	Single	1064.64	1364.2	5323.22	6820.98	28.14	28.14
2	100-124K	Married	1340.9	1676.16	30840.61	38551.57	25.0	25.0
3	100-124K	Unknown	1518.11	1714.4	9108.66	10286.43	12.93	12.93
4	125-149K	Married	1880.21	2334.43	45125.06	56026.36	24.16	24.16
5	125-149K	Single	1205.9	1321.51	4823.61	5286.03	9.59	9.59
6	125-149K	Unknown	1869.26	2051.1	18692.61	20511.03	9.73	9.73
7	15-24K	Married	829.77	947.71	19914.53	22745.04	14.21	14.21
8	15-24K	Unknown	956.01	1221.33	40152.38	51295.88	27.75	27.75
9	15-24K	Single	964.08	1189.98	7712.63	9519.82	23.43	23.43

Insights:

- The sales for each category have from last year. This shows growth.
- Maximum growth is seen for Single adult household with income range 100-124K.
- Considerable growth in purchase is also seen for income range 15-24K.

Question 12: *Analyze how coupon usage differs across households of different sizes. Determine if larger households are more likely to use coupons and if this leads to higher overall savings or spending.*

Query walkthrough:

The query aggregates sales and discount data per basket in cte1, calculates total discounts in cte2, and joins this with household size data in cte3. It then sums total sales and discounts by household size and orders the results to analyze how coupon usage and its impact vary across different household sizes.

Query:

```
with cte1 as (select
                  basket_id, household_key,
                  sum(Sales_value) as sales_value,
                  sum(Retail_disc) as retail_disc,
                  sum(coupon_match_disc) as match_disc,
```

```
        sum(coupon_disc) as coupon_disc
        from project.transactions
        group by basket_id, household_key),
    cte2 as (select
         basket_id, household_key,
         sales_value,
         retail_disc +match_disc+coupon_disc as total_discount
         from cte1
         ),
     cte3 as (select
          d.household_size_desc,
          round(sum(c.sales_value),2) as total_sales_value,
          round(sum(c.total_discount),2) as total_discount
          from cte2 c join project.demographics d
          using(household_key)
          group by d.household_size_desc)
select * from cte3
order by household_size_desc
```

Query results

SAVE RESUL

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | E

Row	household_size_desc	avg_sales_value	total_sales_value	total_discount
1	1	16.35	640187.6	-113083.86
2	2	19.29	880826.16	-153613.59
3	3	19.29	332013.32	-59549.29
4	4	22.41	175637.09	-33422.0
5	5+	24.38	221412.68	-39306.39

Insights:

- With increasing household_size, average purchase amount also increases.
- For household_size '4', the total sales is low suggesting less customers belongs to this category.
- With increase in household_size, the discounts are also increasing suggesting more use of coupons and other discounts.

Overall Insights:

- **Sales Distribution**: Approximately 50% of orders are below $10, with a significant portion (30%) having a sales value above $20. Orders between $5 and $10 make up 19% of total orders, while 50% of orders exceed $10.
- **Store Performance**: STORE_IDs 367, 406, and 381 are the busiest, with STORE_ID 367 having the highest foot traffic, indicating strong customer attraction and engagement.
- **Customer Spending**: The customer with household_key 2042 has the highest average purchase amount of $90. Other high-value customers like household_key 2479 have made substantial purchases, but some high-value customers do not visit regularly, suggesting potential for increased engagement.

- **Product Trends**: Gasoline (product_id 6534178) is the most frequently bought product, and milk is also a top product, frequently appearing in top product combinations.
- **Peak Times**: Foot traffic peaks between 4 pm and 6 pm, with the busiest hour being 5 pm to 6 pm. Average revenue is higher between 12 pm and 4 pm.
- **Demographic Insights**: Households with more than three kids and those with home ownership show higher average and total sales. The highest growth in sales is observed in single adult households with incomes between $100-124K and households with incomes between $15-24K.
- **Average Purchase Amount**: As household size increases, the average purchase amount also rises, indicating that larger households tend to spend more per transaction.
- **Household Size 4**: The total sales for households with a size of 4 are relatively low, suggesting that this category has fewer customers compared to other sizes.
- **Discount Usage**: Larger households receive more discounts, implying that they are more likely to use coupons and other promotional offers.

Recommendations:

1. **Include Date Column**: Add a date column to the transactions table to enable time-based analyses, such as month-on-month growth and pattern changes. This will facilitate more granular insights into sales trends and seasonal variations.
2. **Clarify Categorical Data**: Improve the naming conventions for categorical distributions, particularly for Home Ownership categories, to make their meanings clearer. This will enhance the interpretability of the data.
3. **Data Format Consistency**: Ensure that the trans_time column in the transactions table is in a proper date-time format to support accurate time-based analyses.
4. **Enhance Customer Engagement**: Focus on increasing engagement with high-value customers who visit infrequently. Implement targeted marketing strategies or loyalty programs to encourage more frequent visits.
5. **Analyze High Foot Traffic**: Given that STORE_ID 367 has the highest foot traffic, explore the factors contributing to its success, such as location advantages or marketing strategies, and consider replicating these factors in other stores.
6. **Regular Purchase Patterns**: Investigate and address the lack of regular purchasing among some customers, especially those with high average purchase amounts. Developing strategies to encourage more consistent purchasing habits could be beneficial.
7. **Leverage Product Data**: Capitalize on the popularity of products like Gasoline and Fluid Milk White by optimizing inventory and marketing strategies around these items. Given their high sales volume and frequent combinations, these products should be a focus.
8. **Time-Based Sales Strategies**: Adjust staffing and promotions to align with peak foot traffic hours (4 pm to 6 pm) and higher average revenue periods (12 pm to 4 pm). This can help maximize sales and improve customer service during peak times.

9. **Tailor Sales Strategies**: Recognize that households with more than three kids tend to spend more, and tailor marketing and sales strategies to target these larger households effectively. Additionally, emphasize the benefits of home ownership in marketing efforts, as it correlates with higher sales.
10. **Enhance Inventory Management**: Utilize the sales data to refine inventory management practices. Given that certain products like Gasoline and Fluid Milk White are consistently popular, ensure that these items are always well-stocked.
11. **Targeted Promotion**: Offer promotions tailored to high-value customers or provide incentives to increase visit frequency for infrequent high-spenders.
12. **Target Marketing for Medium-Sized Households**: Since households of size 4 have lower total sales, consider targeted marketing strategies or promotions to increase engagement with this group and boost sales.
13. **Optimize Discount Strategies**: Given that larger households use more discounts, enhance coupon and discount strategies to maximize their effectiveness. Tailor discount offers to encourage more frequent purchases and higher spending among these households.