

# The Look E-Commerce

Modern Clothing Site



# Table of content

**Business  
Background  
&  
Problems**

**Data  
Preparation  
&  
ERD of Dataset**

**Profit  
&  
Revenue  
Analysis**

**Retention rate  
Analysis**

# *Business Background*



## ***The Look***

---

The Look is an e-commerce business in the fashion industry that can be affected by the potential crisis in 2023. To get a handle on the circumstances, the company needs to be in optimization mode. The company is planning to cut off resources in some categories and increase the retention rate by understanding the retention behaviors of the users.



# *Business* Problem

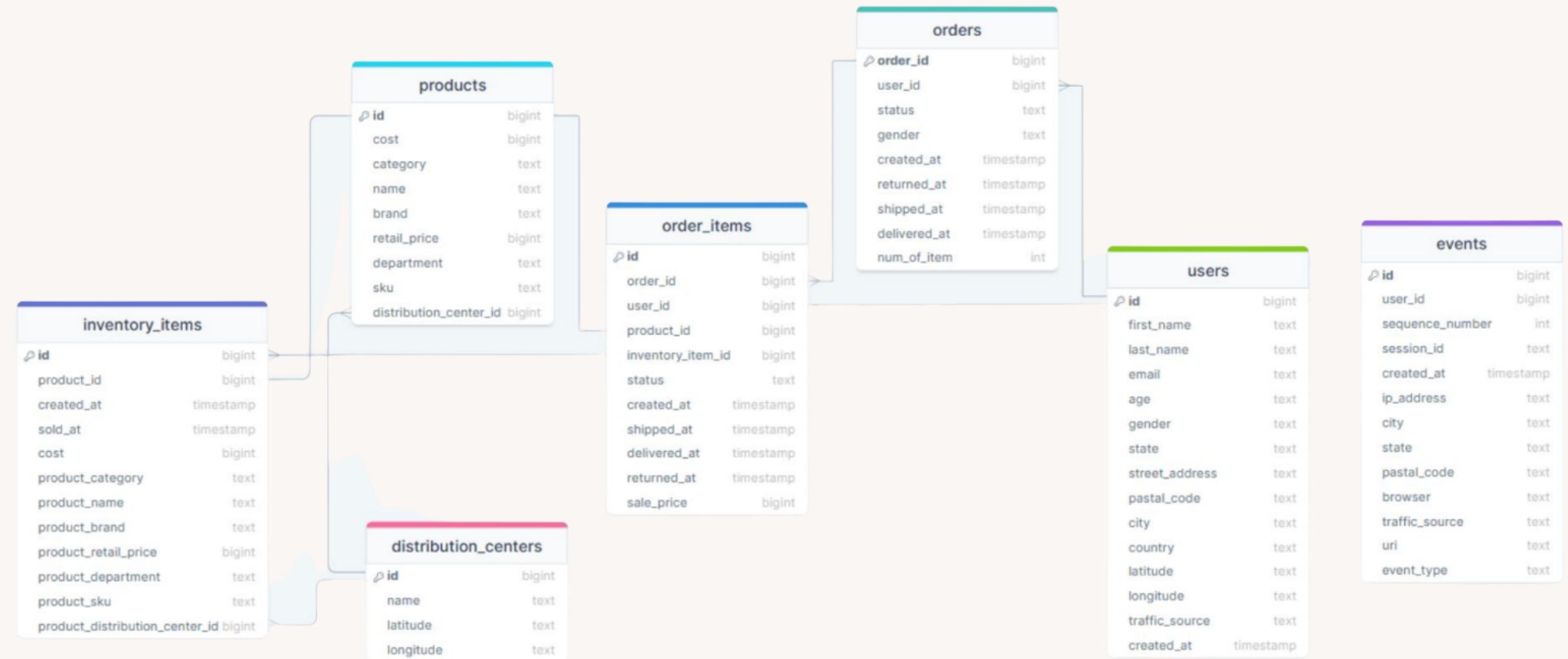
---

What categories need to be deprioritized?  
And how to increase the retention rate by the  
following month?

# *Dataset & Entity Relationship Diagram*

The company has given data analyst to access their database containing information about users and order activity, also product and inventory distribution center.

ERD of this dataset shown here :



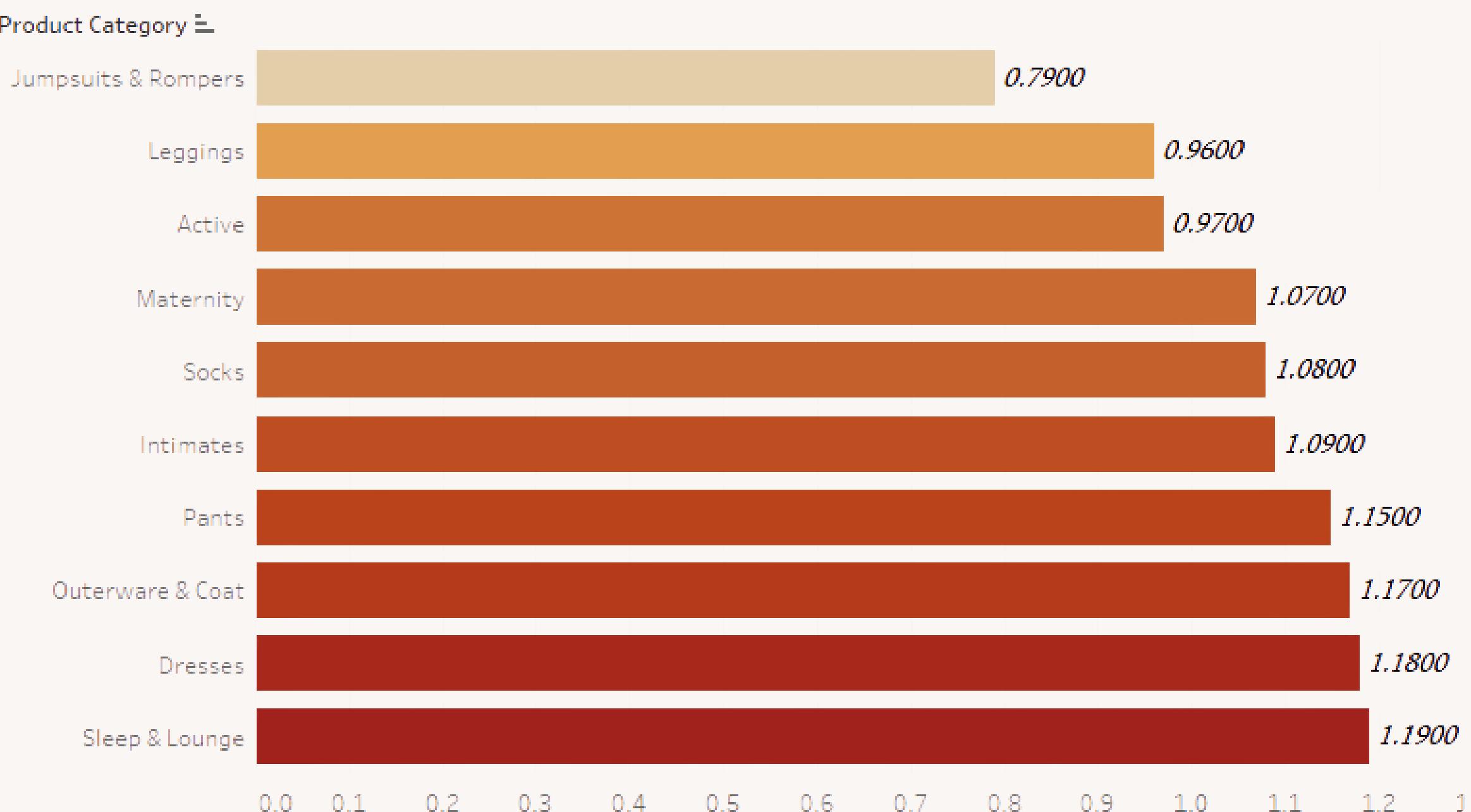
# *Lowest to Highest Profit Growth*

Based on the result, we can see that

"Jumpsuits & Rompers" has the **lowest profit growth** compared to others.

Growth revenue can be calculated by following function :

$$\text{Profit} / \text{LAG(Profit)} - 1$$



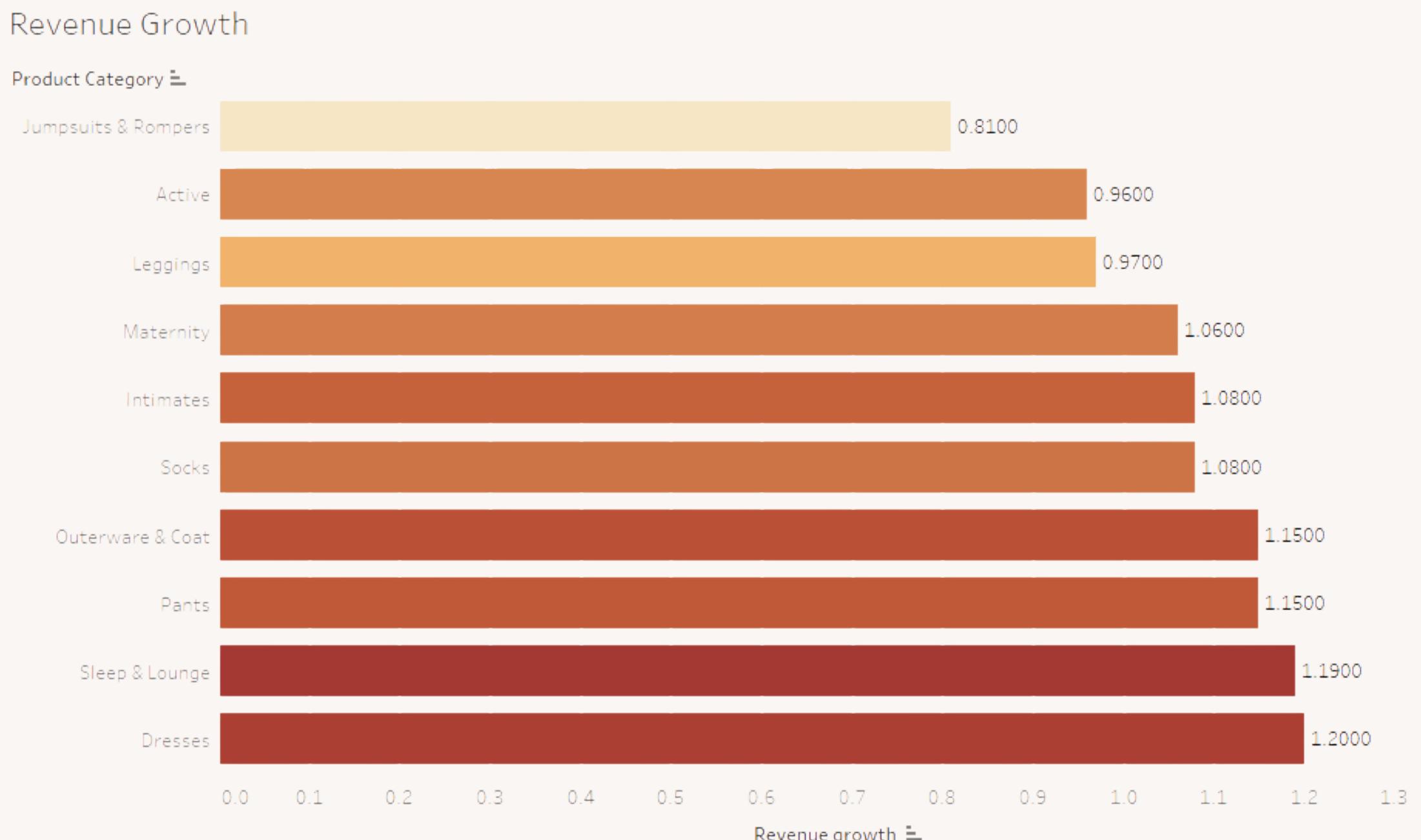
## *Lowest to Highest Revenue Growth*

Based on the result, we can see that

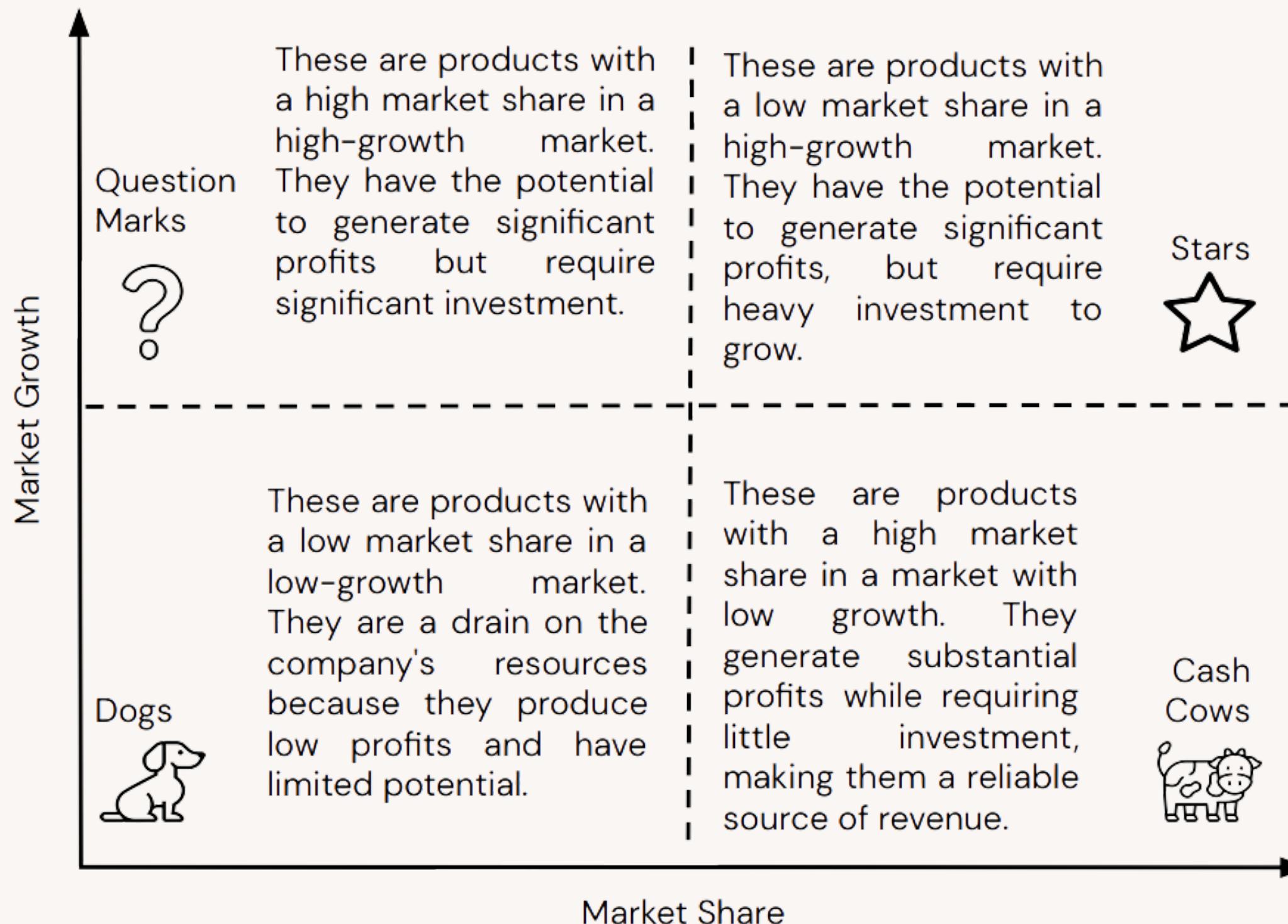
"Jumpsuits & Rompers" has the **lowest revenue growth** compared to others.

Growth revenue can be calculated by following function :

$$\text{Revenue}/\text{LAG}(\text{Revenue}) - 1$$



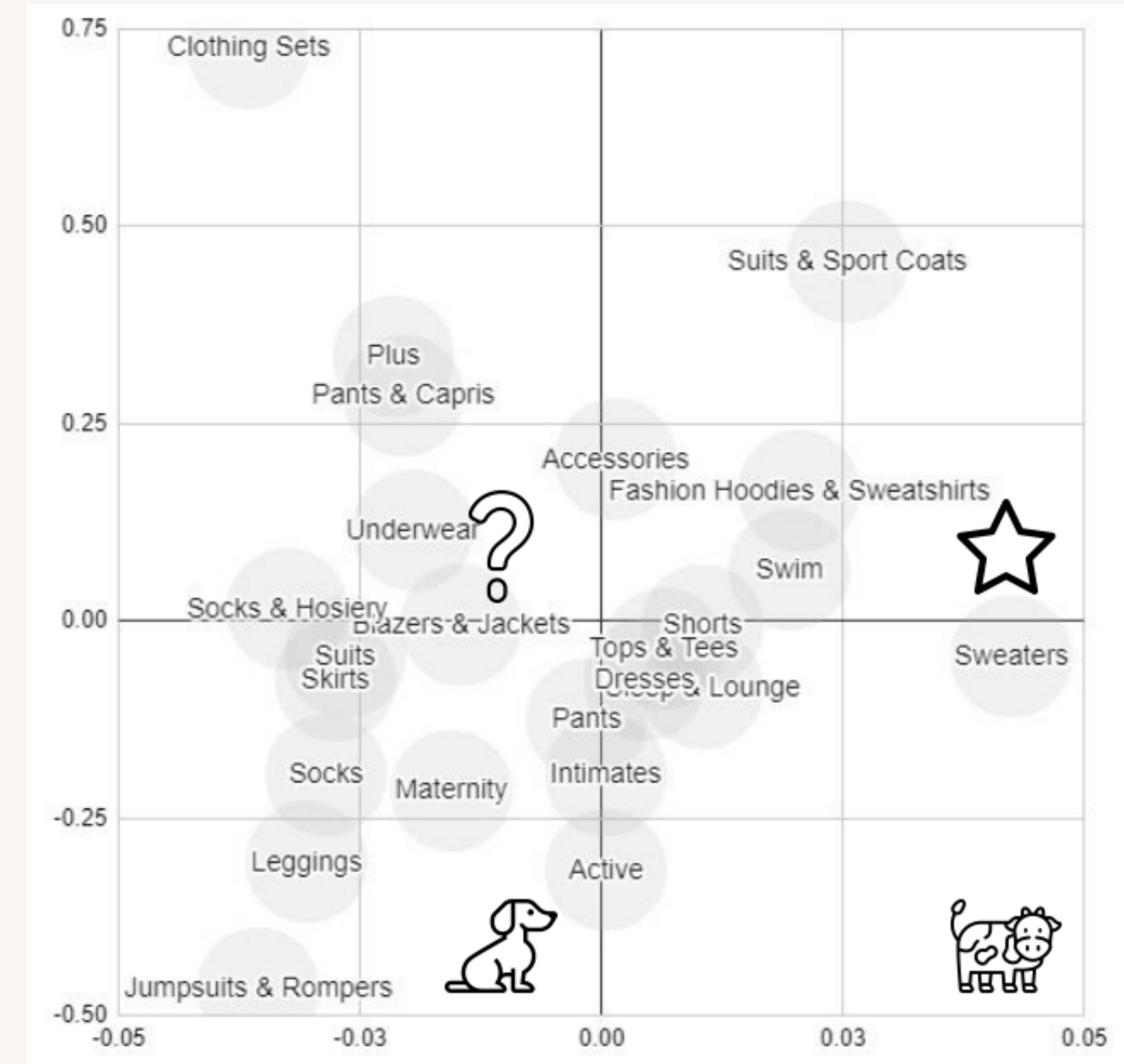
# BCG Matrix



We projected the results into the BCG Matrix to better understand the category's growth. We examine our product portfolio using the BCG Matrix to determine how to allocate strategic resources and order investments.

# *BCG Matrix*

Based on the BCG Matrix, we can deprioritize **Jumpsuits & Rompers** and **Leggings** that fall near the bottom of the dog quadrant. It could imply cutting marketing budgets or reallocating resources to focus on higher-potential categories like **Suits & Sport Coats** or **Fashion Hoodies & Sweatshirts** in the star quadrant.



# Cohort and User Retention Analysis

first_purchase_cohort_sizes	SUM of perc_retention	diff_month_active												
		0	1	2	3	4	5	6	7	8	9	10	11	12
2022-01-01	963	100	1.45	1.66	1.45	1.25	1.45	1.45	1.25	1.77	1.66	1.14	1.56	1.04
2022-02-01	869	100	1.04	1.27	1.27	1.38	1.04	1.15	1.73	0.58	0.81	1.38	1.73	
2022-03-01	1002	100	1.10	1.40	1.90	2.59	2.20	1.50	1.10	1.40	1.30	1.20		
2022-04-01	1052	100	1.52	1.52	2.28	1.14	1.05	1.90	1.43	1.62	1.90			
2022-05-01	1143	100	1.66	1.92	2.27	2.19	1.75	1.84	1.49	1.22				
2022-06-01	1172	100	2.39	1.79	2.22	1.37	1.62	1.62	1.37					
2022-07-01	1318	100	2.50	1.52	2.12	1.82	1.67	1.52						
2022-08-01	1390	100	2.01	2.73	1.94	2.88	1.51							
2022-09-01	1528	100	2.75	1.83	2.03	2.29								
2022-10-01	1594	100	3.32	3.83	2.95									
2022-11-01	1838	100	3.70	3.59										
2022-12-01	2168	100	6.09											

- User retention rate is relatively low (around 1% - 6%) in the following months.
- User retention rate in the following month stagnan over time period.
- The number of new user increase from March to December 2022.

# *Recommendations*

- Before making decisions to dropping or deprioritizing a “**Jumpsuit and rompers** ” product category, company need to conduct a through analysis to understand the reason behind those possibility. Here is the recommendations :
  - Company need to conduct market research to understand the latest trend in the dress category, including styles, colors, or materials that are popular.
  - Offer promotions, such as discounts, bundles and limited-time offers to make customer make a purchases and increases sales.
  - In order to improve loyal customers (increase retention rate), company need to improve quality of customer service, giving a discounts or promo events and loyal points program



---

# Thank You !

Let's connect !

 [awalraiss@gmail.com](mailto:awalraiss@gmail.com)

 [/awalraiss](#)

 [awalraiss](#)

# SQL Syntax #1

```
WITH OBS1 AS
(
SELECT
    EXTRACT(MONTH FROM DATE_TRUNC(oi.created_at, MONTH)) AS month,
    product_category as Kategori_Produk,
    SUM(sale_price-cost) as profit,
    SUM(sale_price) as revenue,
    FROM `sql-project-376612.thelook_ecommerce.order_items` as oi
    JOIN `sql-project-376612.thelook_ecommerce.inventory_items` as
    inv_items
    ON oi.inventory_item_id = inv_items.id
    AND DATE(oi.created_at) BETWEEN "2022-01-01" AND "2022-12-31"
    GROUP BY 2,1
    ORDER BY 1,2
),
```

```
Growth AS
(
SELECT
    OBS1.month,
    Kategori_Produk,
    profit/LAG(profit) OVER(PARTITION BY Kategori_Produk ORDER BY
month) -1 as profit_growth,
    revenue/LAG(revenue) OVER(PARTITION BY Kategori_Produk ORDER
BY month)-1 as revenue_growth
FROM OBS1
)
SELECT
    DISTINCT Kategori_Produk,
    ROUND(AVG(Growth.profit_growth) OVER (PARTITION BY
Kategori_Produk)*100 ,2) as avg_profit,
    ROUND(AVG(Growth.revenue_growth) OVER (PARTITION BY
Kategori_Produk)*100 ,2) as avg_revenue_growth
FROM Growth
ORDER BY 2,1
```

# SQL Syntax #2

```
WITH user_raw AS(
  SELECT DISTINCT
    user_id,
    DATE_TRUNC(DATE(created_at), MONTH) as active_month,
    MIN(DATE_TRUNC(DATE(created_at), MONTH)) OVER (PARTITION
      BY user_id) as first_purchase_month
  FROM `sql-project-376612.thelook_ecommerce.orders`
  WHERE status = "Shipped"
)
,
ddiff_activity AS(
  SELECT *
  ,DATE_DIFF(active_month, first_purchase_month, MONTH) as
  diff_month_active
  FROM user_raw
)
,
user_active_monthly AS (
  SELECT
    first_purchase_month,
    diff_month_active,
    COUNT(DISTINCT user_id) as total_user
  FROM ddiff_activity
  WHERE first_purchase_month >= "2022-01-01"
  GROUP BY 1,2
  ORDER BY 1,2
)
,
cohort_size AS(
  SELECT
    first_purchase_month,
    COUNT(DISTINCT user_id) cohort_sizes
  FROM ddiff_activity
  GROUP BY 1
)
SELECT
  t1.*,
  t2.diff_month_active,
  t2.total_user,
  t2.total_user/ t1.cohort_sizes * 100 as perc_retention
FROM cohort_size t1
LEFT JOIN user_active_monthly as t2
ON t1.first_purchase_month = t2.first_purchase_month
ORDER BY 1,3
```