Albert Ripoll Batlló

Máster Big Data La Salle, Barcelona

Práctica Introducción BI

Feb 10, 2024

Resumen

"TheLook" es una empresa de moda que ha experimentado una transformación significativa en los últimos años, pasando de un enfoque tradicional en tiendas físicas a una estrategia de comercio electrónico desde 2020. Durante este período, la empresa ha logrado vender productos de moda de más de 2000 marcas en 16 países a través de su plataforma de comercio electrónico. Con el objetivo de gestionar eficientemente el gran volumen de datos generado por estas operaciones, la empresa planea implementar un proyecto de Business Intelligence en 2024.

Actualmente, "TheLook" opera tres sitios web, cada uno enfocado en un continente específico, y centraliza los datos en una plataforma de comercio electrónico, utilizando Shopify. El equipo de e-commerce es responsable de la gestión de datos y actualiza los registros consolidados de ventas en la plataforma de BigQuery cada 15 minutos. Posteriormente, el equipo de Business Intelligence se encarga de procesar estos datos a través de un proceso de extracción, carga y transformación (ELT) en BigQuery. Finalmente, generan contenido analítico utilizando la plataforma de dashboarding Holistics.

El objetivo principal de este proyecto de Business Intelligence es permitir una monitorización efectiva de las ventas y facilitar la identificación de nuevas oportunidades de crecimiento para la empresa. Al consolidar y analizar los datos de manera más eficiente, se espera que "TheLook" esté mejor posicionada para tomar decisiones estratégicas informadas y continuar expandiendo su presencia global en el competitivo sector de la moda.

Implementar vista para cada entidad del diseño estrella

Vista DT_PRODUCT

2.1

```
Create Or Replace View lasalle-albert-ripoll.caso_practico.DT_PRODUCT As
   Select id as product_id,
       Case
           When (name Is Null Or Trim(name)='')
           Then 'SENSE-NOM'
           Else Trim(name)
       End as name,
       Case
           When (category Is Null Or Trim(category)='')
           Then 'SENSE-CATEGORIA'
           Else Trim(category)
       End as category,
       Case
           When (brand Is Null Or Trim(brand)='')
           Then 'SENSE-MARCA'
           Else Trim(brand)
       End as brand
   From `bigquery-public-data.thelook_ecommerce.products`;
```

Vista DT_TIME

2.2

```
Create Or Replace View lasalle-albert-ripoll.caso_practico.DT_TIME AS
   Select Distinct created_at                                          As time_id,
     Cast(Extract(YEAR From created_at) As String)                     As year,
     LPad(Cast(Extract(MONTH From created_at) As String), 2, '0') As month,
     LPad(Cast(Extract(DAY From created_at) As String), 2, '0')   As day,
     FORMAT_TIMESTAMP('%Y_%m', created_at)                             As yyyy_mm,
     FORMAT_TIMESTAMP('%Y_%m_%d', created_at)                          As yyyy_mm_dd
   from `bigquery-public-data.thelook_ecommerce.orders`;
```

Vista DT_TICKET_LINE

3.1

```
Create Or Replace View lasalle-albert-ripoll.caso_practico.DT_TICKET_LINE As
   Select items.id      As ticket_line_id,
     items.order_id     As ticket_id,
     orders.status      As order_status,
     items.status       As order_line_status,
     orders.created_at  As order_created_at
   From `bigquery-public-data.thelook_ecommerce.order_items` items
   Inner Join `bigquery-public-data.thelook_ecommerce.orders` orders
   on (orders.order_id=items.order_id);
```

Vista DT_CUSTOMERS

3.2

```
Create Or Replace View lasalle-albert-ripoll.caso_practico.DT_CUSTOMER As
With users_ids As (
   Select email, max(id) as max_user_id, min(id) As min_user_id
   From `bigquery-public-data.thelook_ecommerce.users`
   Group By email
),
 users_info_max As (
   Select cust.email,
   cust.first_name,
   cust.last_name,
   concat(trim(cust.first_name),', ',trim(cust.last_name)) As full_name,
   cust.age,
   Case
           When (cust.age < 18) Then '(0,18)'
           When (cust.age >=18 And cust.age<25) Then '[18,25)'
           When (cust.age >=25 And cust.age<35) Then '[25,35)'
           When (cust.age >=35 And cust.age<45) Then '[35,45)'
           When (cust.age >=45 And cust.age<60) Then '[45,60)'
           When (cust.age >=60) Then '[60,-)'
           Else '<N/A>'
       End As age_range,
```

```
   cust.gender,
   cust.country,
   cust.city,
   cust.traffic_source
   From `bigquery-public-data.thelook_ecommerce.users` cust
   Inner Join users_ids ids
       On (cust.id=ids.max_user_id)
),
 users_info_min AS (
   Select cust.email, cust.created_at,
   ROUND(CEILING(TIMESTAMP_DIFF(CURRENT_TIMESTAMP(), cust.created_at, DAY)/365),0)
       As creation_lifespan_years
   From `bigquery-public-data.thelook_ecommerce.users` cust
   Inner Join users_ids ids
       On (cust.id=ids.min_user_id)
)
Select max.*, min.* EXCEPT(email)
From users_info_max max Inner Join users_info_min min
On (max.email=min.email);
```

Vista Tabla de Hechos FT_SALES

4.1

```
Create Or Replace View lasalle-albert-ripoll.caso_practico.FT_SALES As
With orders_ext As (
   Select users.email, orders.order_id As ticket_id, orders.created_at As time_id,
   ROUND(CEIL(TIMESTAMP_DIFF(orders.delivered_at, orders.created_at,DAY)),0)
      As days_from_creation_to_delivery,
   ROUND(CEIL(TIMESTAMP_DIFF(orders.shipped_at, orders.created_at,DAY)),0)
      As days_from_creation_to_shipment,
   ROUND(CEIL(TIMESTAMP_DIFF(orders.delivered_at, orders.shipped_at,DAY)),0)
      As days_from_shipment_to_delivery
   From `bigquery-public-data.thelook_ecommerce.orders` orders
   Join `bigquery-public-data.thelook_ecommerce.users` users
       on (orders.user_id=users.id)
), items_ext As (
```

5

```
   Select items.order_id   As ticket_id,
   items.id                As ticket_line_id,
   products.id             As product_id,
   1                       As total_products,
   Case
     When items.status in ('Complete','Shipped','Processing')
     Then items.sale_price Else 0
   End                     As net_sales,
   products.retail_price   As gross_sales,
   (products.retail_price - items.sale_price) As total_discount
   From `bigquery-public-data.thelook_ecommerce.order_items` items
   Inner Join `bigquery-public-data.thelook_ecommerce.products` products
       On (items.product_id=products.id)
)
Select
   items.ticket_line_id, orders.email, items.product_id, orders.time_id,
   orders.days_from_creation_to_delivery, orders.days_from_creation_to_shipment,
   orders.days_from_shipment_to_delivery, items.total_products,
   items.net_sales, items.gross_sales, items.total_discount
From orders_ext orders Left Join items_ext items
On (orders.ticket_id = items.ticket_id);
```

Capa semántica

Crear una nueva vista (V_SALES_SEMANTIC_LAYER) que exponga la capa semántica a partir de las 5 tablas base en una única vista.

6.1

V_SALES_SEMANTIC_LAYER

```
Create Or Replace View lasalle-albert-ripoll.caso_practico.V_SALES_SEMANTIC_LAYER As
Select FT_SALES.email, * EXCEPT(ticket_id,ticket_line_id,product_id,time_id,email)
From lasalle-albert-ripoll.caso_practico.FT_SALES
Inner Join lasalle-albert-ripoll.caso_practico.DT_TICKET_LINE
   On (FT_SALES.ticket_line_id=DT_TICKET_LINE.ticket_line_id)
Left Join lasalle-albert-ripoll.caso_practico.DT_PRODUCT
   On (FT_SALES.product_id=DT_PRODUCT.product_id)
Left Join lasalle-albert-ripoll.caso_practico.DT_TIME
   On (FT_SALES.time_id=DT_TIME.time_id)
Left Join lasalle-albert-ripoll.caso_practico.DT_CUSTOMER
   On (FT_SALES.email=DT_CUSTOMER.email);
```

Preguntas sobre la capa semántica

a) ¿Cuál es el producto más vendido (gross_sales)? ¿A qué marca (brand) pertenece? ¿A qué categoría? ¿Cuántos ingresos (net sales) ha proporcionado? Asumiremos todos los importes están en EUR.

```
-- Si la pregunta realmente es: el producto más vendido (max productos_vendidos):
Select name, category, brand,
   Sum(total_products) as productos_vendidos,
   Sum(gross_sales) as sum_ventas_brutas,
   Sum(net_sales)   as sum_ventas_netas
From lasalle-albert-ripoll.caso_practico.V_SALES_SEMANTIC_LAYER
Group By name, category, brand
Order By Sum(total_products) Desc
Limit 1;

-- Si la pregunta realmente es: el producto que más ventas brutas generó (max sum_ventas_brutas):
Select name, category, brand,
   Sum(total_products) as productos_vendidos,
   Sum(gross_sales) as sum_ventas_brutas,
   Sum(net_sales)   as sum_ventas_netas
From lasalle-albert-ripoll.caso_practico.V_SALES_SEMANTIC_LAYER      7.1
Group By name, category, brand
Order By Sum(gross_sales) Desc
Limit 1;
```

El producto más veces vendido es el “Jeans” “Wrangler Men’s Premium” ya que ha vendido 53 veces.

Fila	name	category	brand	productos_vendidos	sum_ventas_brutas	sum_ventas_netas
1	Wrangler Men's Premium Perfo...	Jeans	Wrangler	53	2523.960052490...	1852.310039520...

El producto con más ingresos brutos es el “Outwear & Coat” “Canada Goose Men’s The Chat…” ya que ha generado un total de 10595 EUR de ingresos brutos con solo 13 productos vendidos.

Fila	name	category	brand	productos_vendidos	sum_ventas_brutas	sum_ventas_netas
1	Canada Goose Men's The Chat...	Outerwear & Coats	Canada Goose	13	10595.0	7335.0

b) ¿Quién es el mejor cliente? ¿Cuántos ingresos ha generado? Utilizar indicador net_sales

```
Select email, full_name, sum(net_sales) as sum_net_sales
From lasalle-albert-ripoll.caso_practico.V_SALES_SEMANTIC_LAYER
Group By email, full_name
Order By sum(net_sales) Desc
Limit 1;
```

8.1

Fila	email	full_name	sum_net_sales
1	nicholaswilson@example.net	Nicholas, Wilson	2409.979997634...

c) ¿Qué año ha sido el que “theLook” ha ingresado más? Utilizar indicador net_sales

```
Select year, SUM(net_sales) As sum_net_sales
From lasalle-albert-ripoll.caso_practico.V_SALES_SEMANTIC_LAYER
Group By year
Order By SUM(net_sales) Desc
Limit 1;
```

8.2

Fila	year	sum_net_sales
1	2023	3533968.1034493

d) ¿Cuántos clientes únicos tiene “theLook”?

```
Select count(distinct email) as clientes_unicos
From lasalle-albert-ripoll.caso_practico.V_SALES_SEMANTIC_LAYER;
```

8.3

Fila	clientes_unicos
1	68892

e) ¿Cuál es el promedio en días entre creación y envío, creación y entrega, y entre envío y entrega de los pedidos? En caso “theLook” quiera optimizar el tiempo total del pedido, ¿en qué proceso de la cadena de envío debería invertir recursos para mejorar?

```
Select avg(days_from_creation_to_shipment) as creacion_envio
    ,  avg(days_from_creation_to_delivery) as creacion_entrega
    ,  avg(days_from_shipment_to_delivery) as envio_entrega
From lasalle-albert-ripoll.caso_practico.V_SALES_SEMANTIC_LAYER;
```

De promedio los tiempos de

• Compra a Creación del Envío son de 1 día: Esta etapa implica el tiempo desde que se realiza la compra hasta que se genera la etiqueta de envío. Aquí se podría evaluar si hay posibles demoras en la confirmación del pedido, en la preparación del producto para el envío o en la generación de la etiqueta de envío. Mejorar la eficiencia en estas áreas podría reducir este tiempo.

• Envío hasta Entrega son de 2 días: Esta etapa implica el tiempo desde que el paquete se envía hasta que se entrega al destinatario. Aquí se podría evaluar la eficiencia de las opciones de transporte y logística, así como la velocidad de procesamiento en los centros de distribución y los tiempos de tránsito. Mejorar la eficiencia en estas áreas podría acortar este período.

9.1

Fila	creacion_envio	creacion_entrega	envio_entrega
1	1.002441348100...	3.503351787883...	2.000189277433...

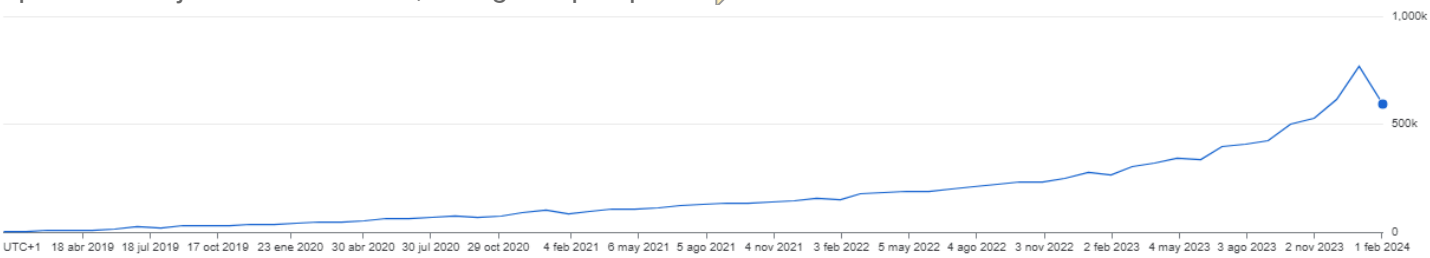
En el análisis resulta sorprendente que los tiempos de creación hasta envío más los tiempos de envío hasta la entrega no sean iguales al tiempo total del envío a la entrega (1+2=3,5). Eso es debido a que los resultados son aproximados porque en el análisis los valores han perdido precisión al redondearlos a días enteros. Además puede ser que algunos paquetes se hayan enviado pero no se hayan llegado a entregar por lo que se podría tener un desajuste de valores nulos entre las distintas columnas. Por ejemplo valores nulos en envío_entrega y creación_entrega pero no nulos en creación_envío.

Implementación de dashboard en preset, que permite generar conocimiento

a) ¿El negocio prospera? Mostrar a nivel temporal la evolución de ingresos.

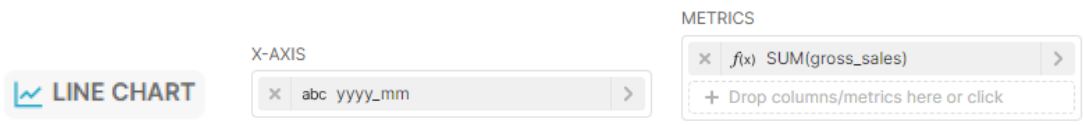
```sql
SELECT
  DATE_TRUNC(time_id, MONTH) as month, --trunca la columna time_id al primer día del mes, lo que te permite agrupar por mes.
  SUM(gross_sales) as total_sales --calcula la suma de las ventas brutas para cada mes.
FROM
  `lasalle-albert-ripoll.caso_practico.FT_SALES`
GROUP BY -- agrupa los resultados por mes.
  month
ORDER BY -- ordena los resultados en orden cronológico.
  month;
```

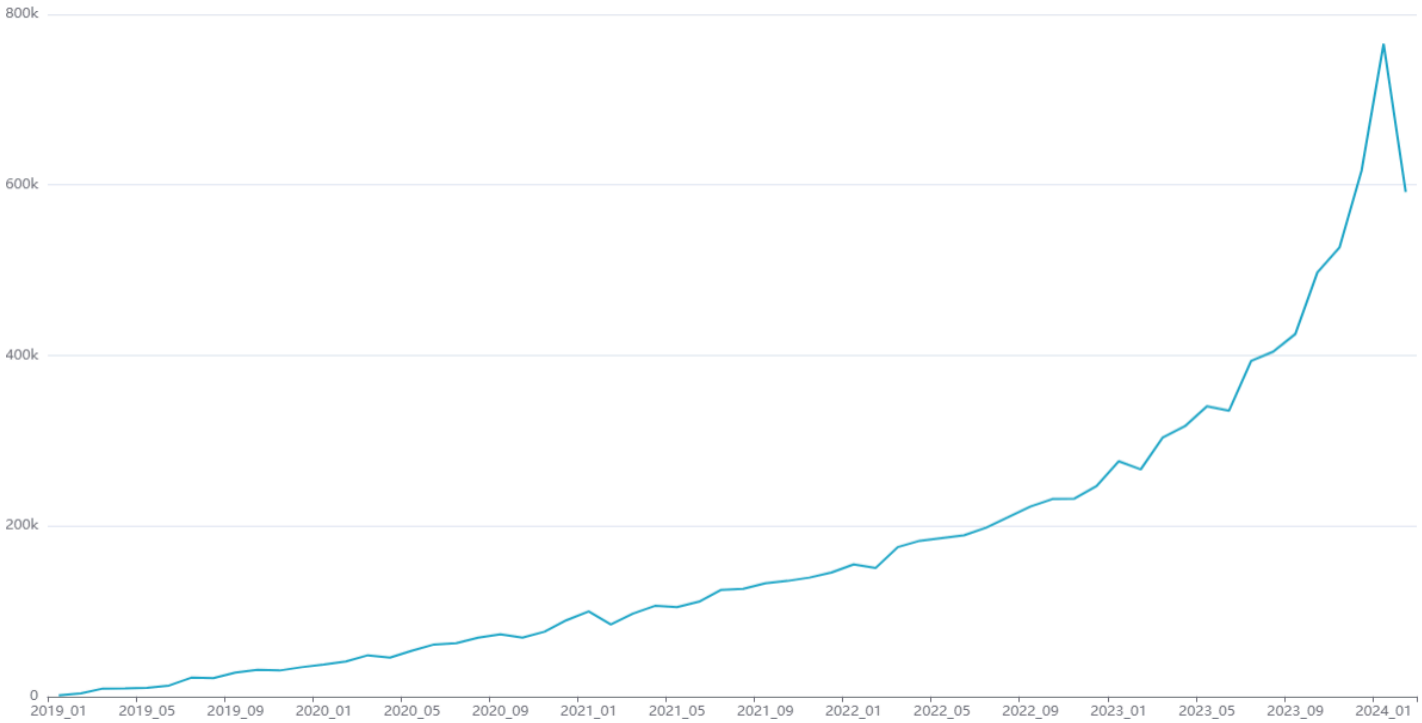
A pesar del bajón del último mes, el negocio prospera. 10.1



Ingresos totales por mes vs mes

Con preset:

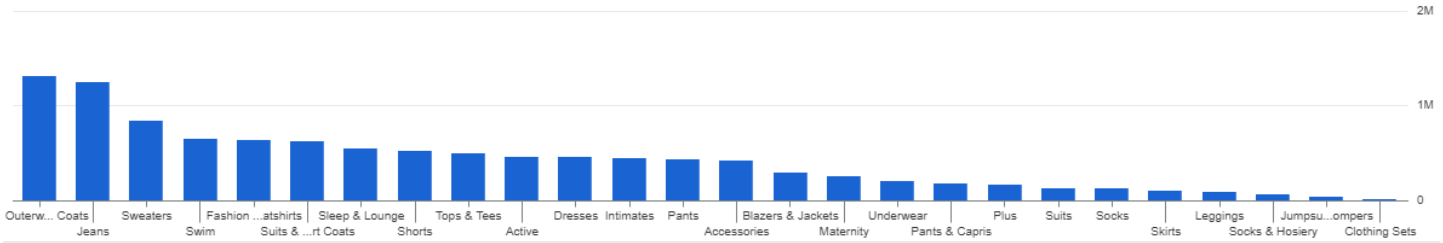




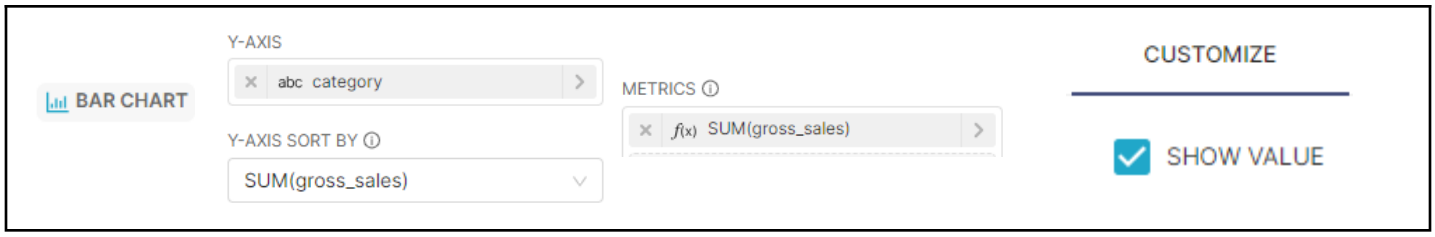
b) ¿Qué tipología de productos debemos posicionar más en la web? Mostrar los ingresos por categoría de producto, con un gráfico que permita visualizar la distribución.

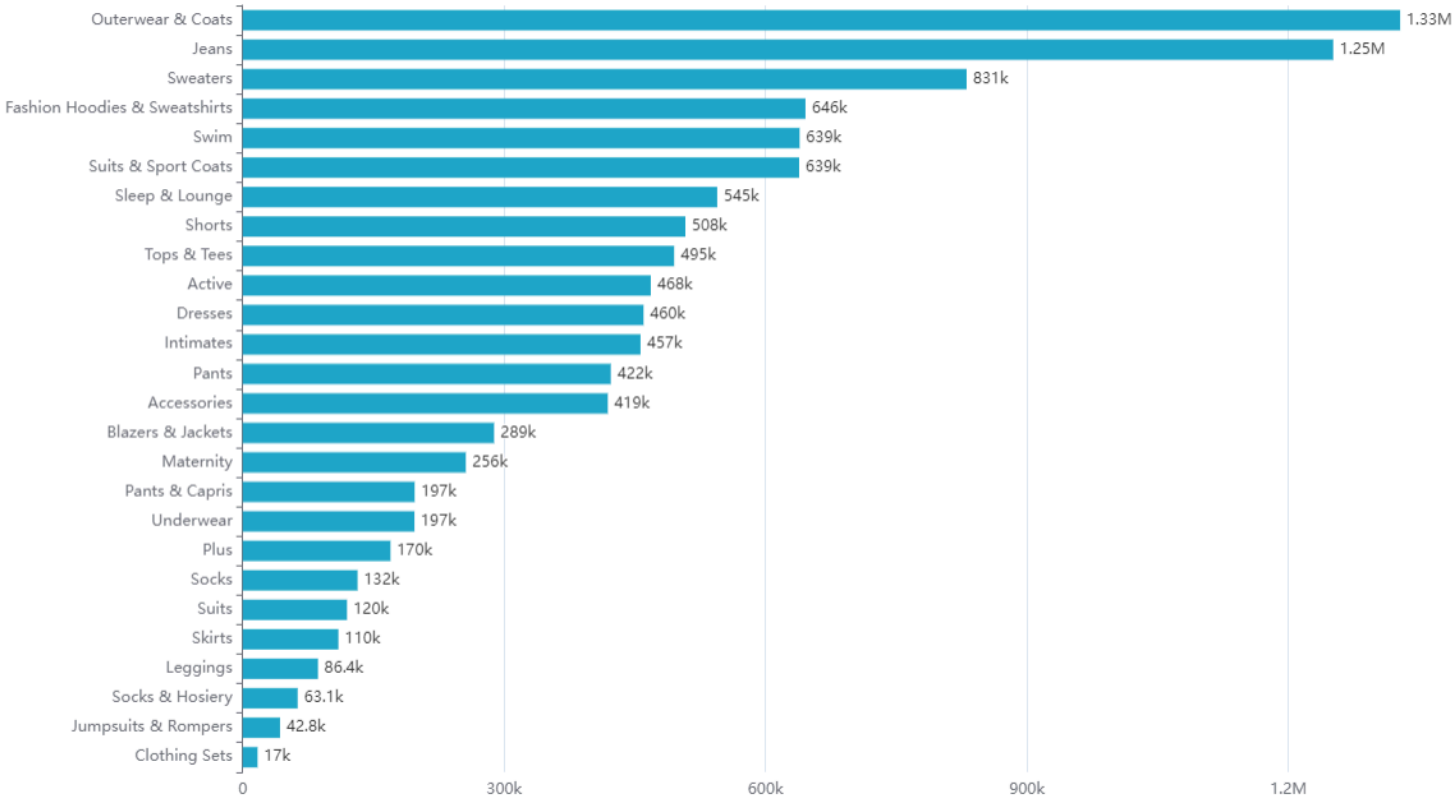
```
SELECT
  P.category,
  SUM(S.gross_sales) as total_sales --suma las ventas brutas para cada categoría de productos.
FROM
  `lasalle-albert-ripoll.caso_practico.FT_SALES` S
JOIN
  `lasalle-albert-ripoll.caso_practico.DT_PRODUCT` P 11.1 se utiliza para combinar las tablas
FT_SALES y DT_PRODUCT mediante la columna product_id
ON
  S.product_id = P.product_id
GROUP BY --agrupa los resultados por la categoría de productos.
  P.category
ORDER BY
  total_sales DESC;--ordena los resultados en orden descendente según las ventas totales.
```

Los productos de “Outwear&Coats” y “Jeans” son los productos que generan más de un millón de ingresos a la empresa.



Ingresos totales por categoría de producto

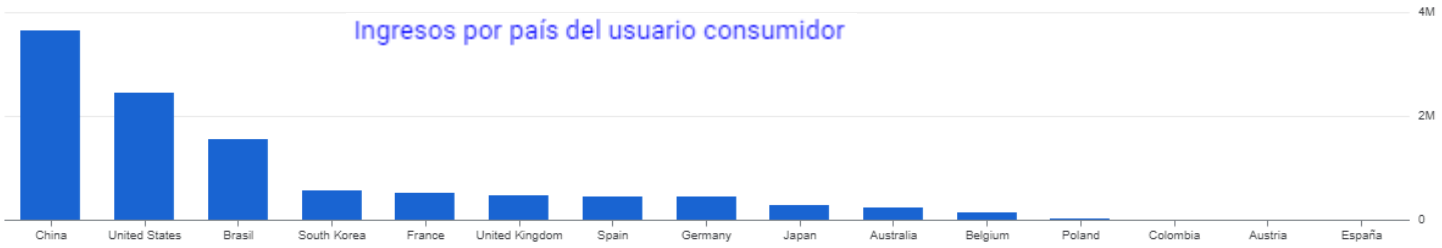


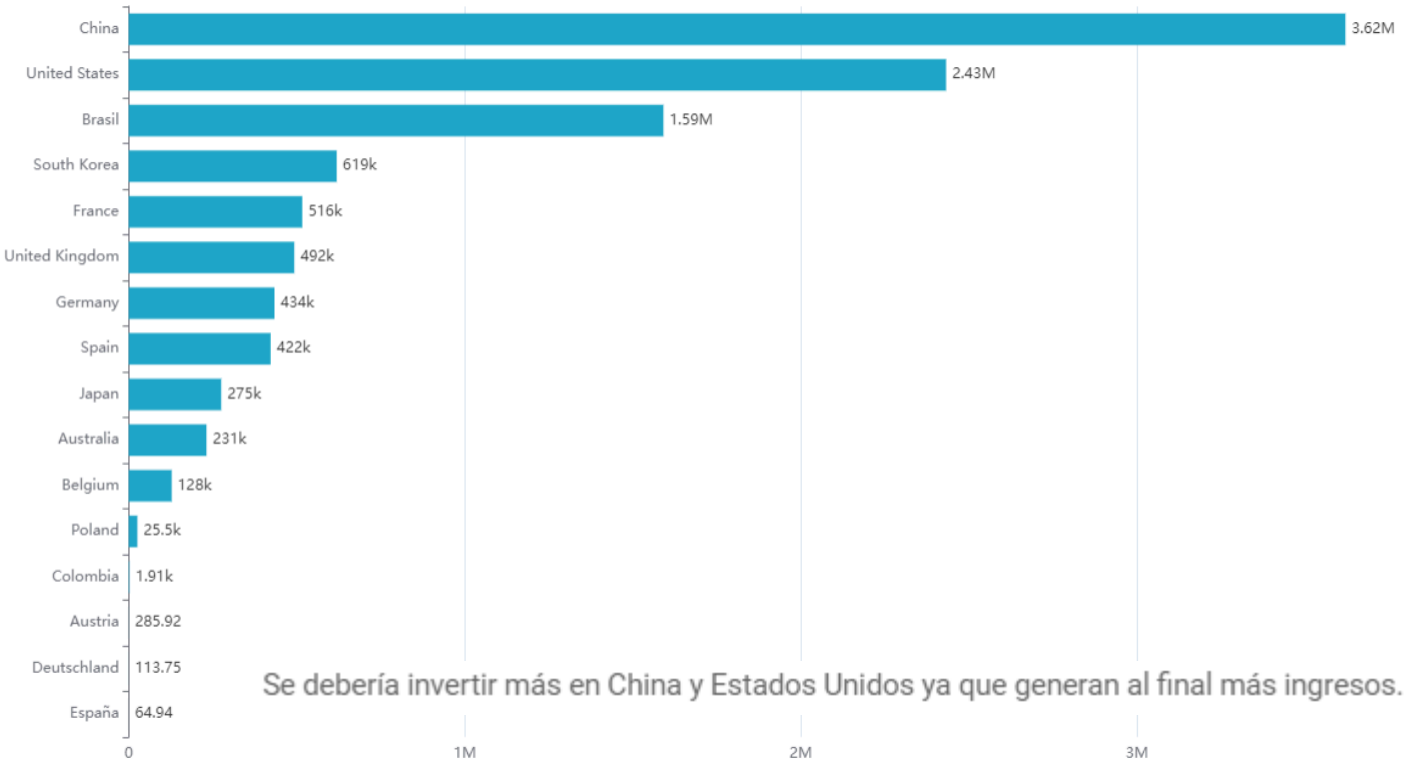


c) ¿En qué mercados (países) debemos invertir en marketing? Mostrar los ingresos a nivel de país.

```sql
SELECT
  country,
  SUM(gross_sales) as total_sales
FROM
  `lasalle-albert-ripoll.caso_practico.V_SALES_SEMANTIC_LAYER`
GROUP BY
  country
ORDER BY
  total_sales DESC;
```

13.1

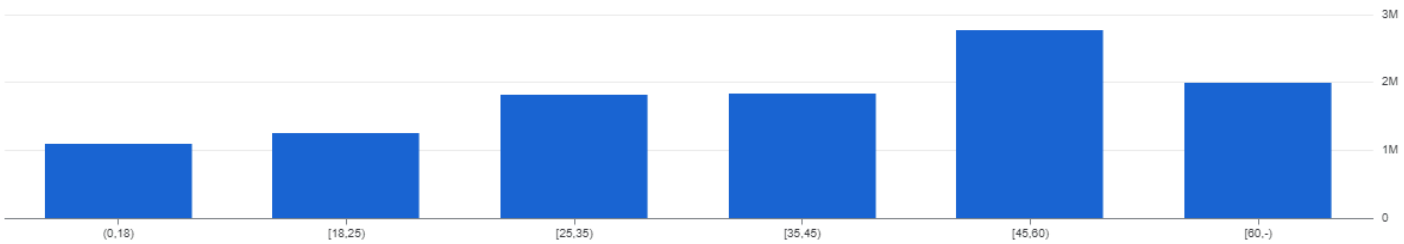




d) ¿En qué medio de marketing deberíamos invertir las campañas? Mostrar los ingresos por rango de edad para facilitar al equipo de marketing saber la edad del consumidor que más monetiza (muy jóvenes TikTok, rango medio Facebook, mayores Televisión).

```sql
SELECT
  age_range,
  SUM(gross_sales) as total_sales
FROM
  `lasalle-albert-ripoll.caso_practico.V_SALES_SEMANTIC_LAYER`
GROUP BY
  age_range
ORDER BY
  age_range;
```

14.1



Ingresos por rango de edad del consumidor

Si consideramos muy jóvenes de 0 a 25 años que usan principalmente TikTok, rango medio de 25 a 45 años que usan Facebook y mayores de 45 que consumen Televisión:

```sql
SELECT
  CASE
    WHEN age_range IN ('(0,18)', '[18,25)') THEN 'TikTok'
    WHEN age_range IN ('[25,35)', '[35,45)') THEN 'Facebook'
    WHEN age_range IN ('[45,60)', '[60,-)') THEN 'Televisión'
    ELSE 'Otro'
  END AS marketing_channel,
  SUM(gross_sales) as total_sales
FROM
  `lasalle-albert-ripoll.caso_practico.V_SALES_SEMANTIC_LAYER`
GROUP BY
  marketing_channel
ORDER BY
  total_sales DESC;
```

Ingresos por red de consumo principal de los consumidores

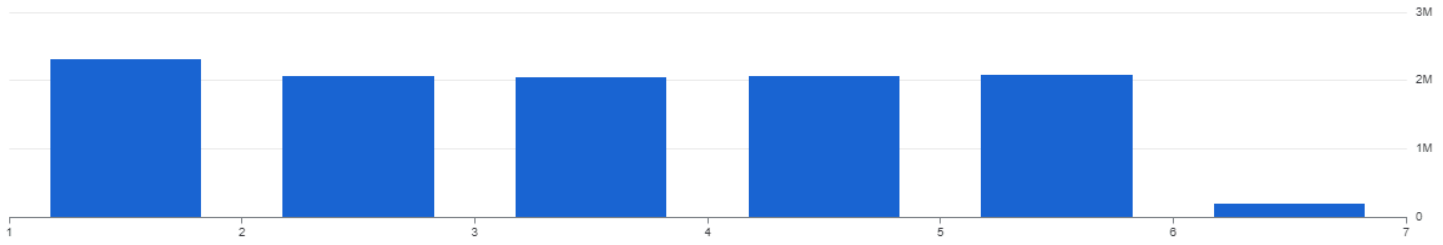
Observamos que se debería invertir más en televisión pero sin menospreciar los otros medios.

e) ¿Nuestros clientes son fieles? Se propone mostrar los ingresos por el campo "creation_lifespan_years", que mostrará los años que el cliente es fiel a la empresa.

```sql
SELECT
  creation_lifespan_years,
  SUM(gross_sales) as total_sales
FROM
  `lasalle-albert-ripoll.caso_practico.V_SALES_SEMANTIC_LAYER`
GROUP BY
  creation_lifespan_years
ORDER BY
  creation_lifespan_years;
```

15.1

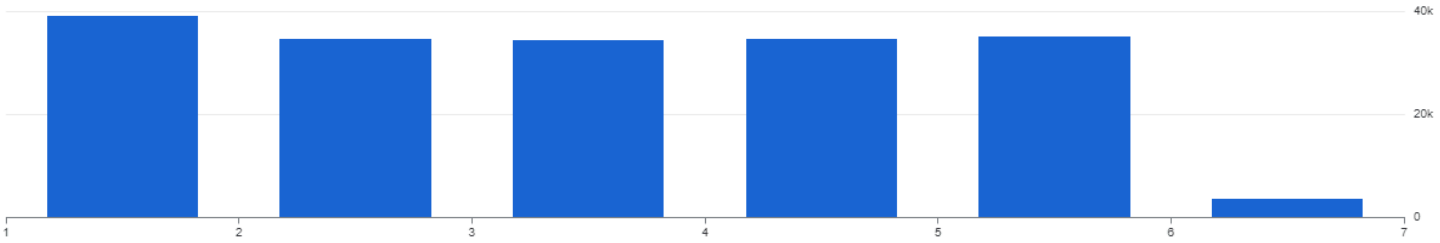
No observamos mucha diferencia de las ventas entre los clientes que llevan 1, 2, 3, 4, 5 o 6 años con la cuenta creada.



Ingresos por antigüedad de creación de la cuenta

Lo que sí vemos es que los usuarios que hace más de 6 años que tienen la cuenta creada generan muchos pocos ingresos. Pero esta razón radica seguramente en que hay pocos usuarios con cuenta de antiguedad mayor a 6 años. Como podemos comprobar:

```sql
SELECT
  CASE
    WHEN creation_lifespan_years < 1 THEN '(0,1)'
    WHEN creation_lifespan_years >= 1 AND creation_lifespan_years < 2 THEN '[1,2)'
    WHEN creation_lifespan_years >= 2 AND creation_lifespan_years < 3 THEN '[2,3)'
    WHEN creation_lifespan_years >= 3 AND creation_lifespan_years < 4 THEN '[3,4)'
    WHEN creation_lifespan_years >= 4 AND creation_lifespan_years < 5 THEN '[4,5)'
    WHEN creation_lifespan_years >= 5 AND creation_lifespan_years < 6 THEN '[5,6)'
    WHEN creation_lifespan_years >= 6 THEN '[6,-)'
    ELSE 'Otro'
  END AS lifespan_range,
  COUNT(*) as count_in_range
FROM
  `lasalle-albert-ripoll.caso_practico.V_SALES_SEMANTIC_LAYER`
GROUP BY
  lifespan_range
ORDER BY
  lifespan_range;
```



Cantidad de cuentas consumidoras por antigüedad de creación de la cuenta

16.1

Índex de comentaris

2.1	1 pto
2.2	1 pto
3.1	1 pto
3.2	1 pto
4.1	1 pto
6.1	2 ptos
7.1	1 pto
8.1	1 pto
8.2	1 pto
8.3	1 pto
9.1	1 pto
10.1	1 pto
11.1	1 pto
13.1	1 pto
14.1	1 pto
15.1	1 pto
16.1	Nota final: 17 / 17 = 10