Ficha técnica: Proyecto 5 de Análisis de Datos

Título del proyecto: Proyecto Profundización: Estructura de datos

Objetivo:
A través del proceso ETL (Extract, Transform, Load), construir un sistema tabular que nos permita almacenar datos de manera eficiente y consultar estos datos más fácilmente.

Equipo:
Individual

Herramientas y Tecnologías:

- Google BigQuery
- Google Colab
- Google Looker Studio
- Google Slides
- Google Docs

Lenguajes:

- SQL
- Python

Insumos:
- Conjunto de datos en este enlace.

Diccionario de datos:

- **category**: Representan las categorías de productos vendidos en el hipermercado.
- **city**: Representa la ciudad donde se realizó el pedido.
- **country:** Representa el país en el que se encuentra el hipermercado.
- **customer_id:** Representa un identificador único para cada cliente.
- **customer_name:** Representa el nombre del cliente que realizó el pedido.
- **discount:** Representa el descuento aplicado al pedido.
- **market:** Representa el mercado o región donde opera el hipermercado.
- **unknown:** Una columna desconocida o no especificada.
- **order_date:** Representa la fecha en la que se realizó el pedido.
- **order_id:** Un identificador único para cada pedido.
- **order_priority:** Representa el nivel de prioridad del pedido.

- **product_id:** Representa un identificador único para cada producto.
- **product_name**: Representa el nombre del producto.
- **profit:** Representa el beneficio generado por el pedido.
- **quantity:** Representa la cantidad de productos pedidos.
- **region:** Representa la región donde se realizó el pedido.
- **row_id:** Representa un identificador único para cada fila del conjunto de datos.
- **sales:** Representa el monto total de venta del producto en el pedido.
- **segment:** Representa el segmento de clientes (por ejemplo, consumidores, empresas u oficinas en casa).
- **ship_date:** Representa la fecha en la que se envió el pedido.
- **ship_mode:** Representa el modo de envío utilizado para el pedido.
- **shipping_cost:** Representa el costo de envío del pedido.
- **state:** Representa el estado o región dentro del país.
- **sub_category:** Representa la subcategoría de productos dentro de la categoría principal.
- **year:** Representa el año en el que se realizó el pedido.
- **market2:** Otra columna relacionada con información de mercado.
- **weeknum:** Representa el número de la semana en la que se realizó el pedido.

Procesamiento y análisis:

2.1 Procesar y preparar la base de datos

1. **Conectar/importar datos a otras herramientas**

Se creó el proyecto5-etl y el conjunto de datos Dataset en BigQuery.

- Tablas importadas: *superstore*

2. **Identificar y manejar valores nulos**

Se identifican valores nulos a través de comandos SQL COUNTIF, IS NULL, AS.

- **superstore:** no se encontraron valores nulos.

3. **Identificar y manejar valores duplicados**

Se identifican valores duplicados a través de comandos SQL COUNT, GROUP BY, HAVING.

- **superstore:** no se encontraron valores duplicados.

4. **Identificar y manejar datos discrepantes en variables categóricas**

Se identifican datos discrepantes en variables categóricas a través de comandos SQL DISTINCT, ORDER BY.

- No se encontraron datos discrepantes en variables categóricas.

5. **Identificar y manejar datos discrepantes en variables numéricas**

Se identifican datos discrepantes en variables numéricas a través de los comandos COUNT, ARRAY_AGG, CAST.

- No se encontraron datos discrepantes en variables numéricas.

6. **Comprobar y cambiar tipo de dato**

Se identifica que las variables ship_date y order_date tienen el tipo de dato TIMESTAMP, sin embargo, los datos tienen el valor de 0:00:00 en la hora, por lo que es recomendable cambiar el tipo de dato a DATE.

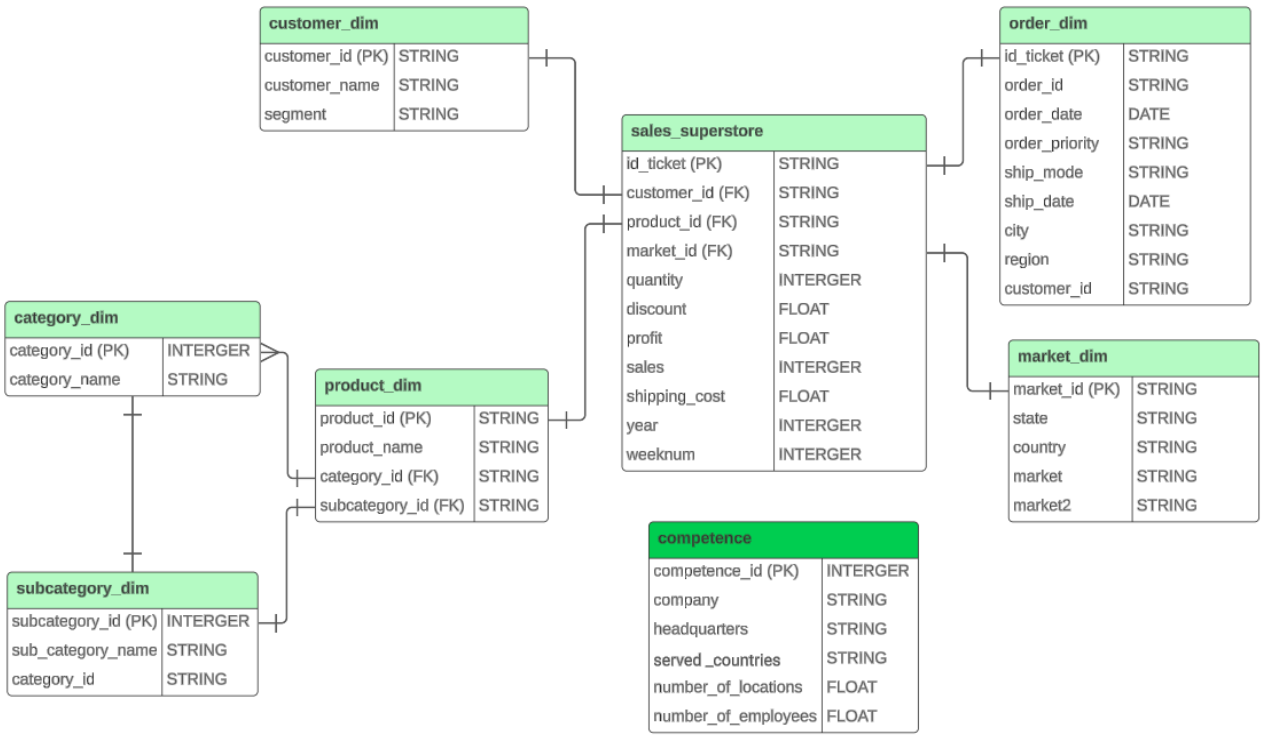
7. **Buscar datos de otras fuentes**

Se extrae información de la tabla “multinacional” de wikipidia utilizando Python en Google Colab.

- Se descargó el archivo csv y se importó a BigQuery para su procesamiento. Tabla importada: *multinational.*

8. **Diseñar estructura de la base de datos (tablas de hechos y dimensiones)**

Modelo de esquema Estrella



Categorizar variables como dimensiones o hechos:

- **Dimensiones**: category, city, country, customer_id, customer_name, market, order_date, order_priority, product_id, product_name, region, segment, ship_date, ship_mode, state, sub_category, year, weeknum.
- **Hechos**: discount, profit, quantity, sales, shipping_cost.
- **Columnas Desconocidas/No especificadas**: unknown, row_id, market2.

Tablas de Dimensiones

customer_dim

- **customer_id (PK):** Representa un identificador único para cada cliente.
- **customer_name:** Representa el nombre del cliente que realizó el pedido.
- **segment:** Representa el segmento de clientes (por ejemplo, consumidores, empresas u oficinas en casa).

category_dim

- **category_id (PK):** identificador único para cada categoría.
- **category_name:** Nombre de las categorías de productos vendidos en el hipermercado.

sub_category

- **subcategory_id:** identificador único para cada subcategoría.
- **sub_category_name:** Nombre de la subcategoría de productos dentro de la categoría principal.
- **category_id: i**dentificador único para cada categoría.

product_dim

- **product_id (PK):** Representa un identificador único para cada producto.
- **product_name:** Representa el nombre del producto.
- **category_id (FK)**
- **subcategory_id (FK)**

order_dim

- **Id_ticket:** identificador único por transacción.
- **order_id (PK):** Un identificador único para cada pedido.
- **order_priority:** Representa el nivel de prioridad del pedido.
- **ship_mode:** Representa el modo de envío utilizado para el pedido.
- **order_date:** Representa la fecha en la que se realizó el pedido.
- **ship_date:** Representa la fecha en la que se envió el pedido.
- **city:** Representa la ciudad donde se realizó el pedido.

- **region:** Representa la región donde se realizó el pedido.

market_dim

- **market_id (PK):** identificador único para cada ubicación.
- **state:** Representa el estado o región dentro del país.
- **country:** Representa el país en el que se encuentra el hipermercado.
- **market:** Representa el mercado o región donde opera el hipermercado.
- **market2:** Otra columna relacionada con información de mercado.

Tabla de Hechos

sales_superstore

- **Id_ticket:** identificador único por transacción.
- **order_id (FK):** Un identificador único para cada pedido.
- **customer_id (FK):** Representa un identificador único para cada cliente.
- **product_id (FK):** Representa un identificador único para cada producto.
- **market_id (FK):**
- **quantity:** Representa la cantidad de productos pedidos.
- **discount:** Representa el descuento aplicado al pedido.
- **profit:** Representa el beneficio generado por el pedido.
- **sales:** Representa el monto total de venta del producto en el pedido.
- **shipping_cost:** Representa el costo de envío del pedido.
- **year:** Representa el año en el que se realizó el pedido.
- **weeknum:** Representa el número de la semana en la que se realizó el pedido.

9. **Crear estructura de la base de datos (tablas de hechos y dimensiones)**

Con los comandos CREATE TABLE, SELECT, DISTINCT, JOIN se crearon y llenaron las tablas en BigQuery.

- **Tabla customer_dim**: contiene 4,873 datos de clientes.
- **Tabla category_dim:** contiene 3 categorías de productos. El category_id se creó colocando el prefijo “CAT” concatenado con un número incremental de al menos dos dígitos.
- **Tabla subcategory_dim:** contiene 17 sub categorías. El subcategory_id se creó colocando el prefijo “SUB” concatenando con un número incremental de al menos dos dígitos.
- **Tabla product_dim**: contiene 10,768 productos.

- **Tabla order_dim:** contiene 25,754 órdenes. Se cambió el tipo de dato de las variables order_date y ship_date de TIMESTAMP a DATE. Se creó un id_ticket único concatenando el order_id y customer_id.
- **Tabla market_dim:** contiene 11,126 estados, se genera un market_id, con el comando CONCAT utilizando las variables state y country.
- **Tabla sales_superstore:** contiene la información de 51, 290 transacciones.
- **Tabla compentence_multinational:** contiene la información de 325 compañías, se creó un competence_id, asignando un número a cada company, se quitaron de la información 49 filas que no tenían en información en las columnas served_countries, number_of_locations, number_of_employees.

10. Programar actualizaciones de tabla

Para diseñar un pipeline de actualización se consideraron las relaciones entre las tablas de hechos y dimensiones. El objetivo es actualizar los datos de manera eficiente y mantener la integridad referencial de las tablas.

1. Obtener Nuevos Datos de la Fuente

- Primero, se obtienen los nuevos datos de pedidos. Estos datos pueden provenir de archivos CSV, JSON, o de una API. Antes de cargarlos en el sistema, se deben preprocesar.

2. Preprocesamiento de Datos

- **Limpieza de datos:** Eliminar duplicados, manejar valores nulos y transformar los datos al formato correcto.
- **Validación de datos:** Asegurarse de que todos los campos cumplan con las restricciones de tipo de datos y otros requisitos.

3. Actualizar Tablas de Dimensiones Primero

Secuencia de Actualización de Tablas de Dimensiones:

1. **category_dim:** Verificar si hay nuevas categorías de productos. Insertar los registros que no existan.
2. **subcategory_dim:** Actualizar con nuevas subcategorías. Asegurarse de que category_id ya esté presente en category_dim antes de insertar.
3. **product_dim:** Verificar e insertar productos nuevos. Validar que tanto category_id como subcategory_id ya existan en sus respectivas tablas de dimensiones.

4. **customer_dim:** Insertar nuevos clientes si no existen en la tabla. Verificar por customer_id.
5. **market_dim:** Insertar combinaciones nuevas de state y country para generar un nuevo market_id.
6. **competence:** Actualizar con nuevos registros de competencia, asegurándose de evitar duplicados.
7. **order_dim:** Insertar los nuevos pedidos. Asegurarse de que los campos order_id, order_date, order_priority, ship_mode, city, region, y customer_id estén correctamente referenciados.

4. Generación del id_ticket para la Tabla de Hechos

El id_ticket es la clave primaria de la tabla de hechos sales_superstore. Para generarlo:

- Combina el order_id y el customer_id con un identificador único, por ejemplo: CONCAT(order_id, '_', customer_id) para asegurar la unicidad.
- Esta clave se debe generar y validar antes de insertar registros en la tabla de hechos.

5. Actualizar la Tabla de Hechos

Una vez que todas las tablas de dimensiones y la tabla order_dim están actualizadas, se puede proceder con la actualización de la tabla de hechos:

- **Insertar Nuevos Pedidos:** Verificar que las claves foráneas (customer_id, product_id, market_id, order_id) existan en las tablas correspondientes. Luego, insertar los nuevos registros en la tabla de hechos.
- **Manejo de Actualizaciones:** Para cambios en datos existentes (e.g., order_priority, ship_mode), identificar y ejecutar actualizaciones (UPDATE) en la tabla de hechos.

6. Frecuencia de actualización de tablas

1. Tablas de Dimensiones (category_dim, subcategory_dim, product_dim, customer_dim, market_dim, competence):

- **Frecuencia de actualización: Semanal o mensual.**

Generalmente nuevas categorías, productos, o clientes no se agregan constantemente, por lo que una actualización semanal o mensual es suficiente.

2. Tabla de Dimensiones (oder_dim):

- **Frecuencia de actualización: diaria.**

La tabla order_dim, sin embargo, puede estar sujeta a cambios diarios, ya que refleja los pedidos realizados.

3. Tabla de Hechos (sales_superstore):

- **Frecuencia de actualización: Diaria.**

Los datos transaccionales (ventas, ganancias, costos, etc.) suelen cambiar constantemente. Una actualización diaria permite un análisis casi en tiempo real y asegura que las decisiones de negocio se basen en la información más reciente.

Es recomendable programar la actualización durante la noche (horas no pico) para minimizar el impacto en los sistemas y garantizar que los datos estén actualizados para las actividades del día siguiente.

4. General (para todo el pipeline):

- **Frecuencia de actualización: Mensual (para revisiones completas).**

Una revisión y actualización completa mensual puede incluir validaciones exhaustivas, asegurarse de la integridad de los datos y limpiar registros obsoletos.

7. Automatización y Mantenimiento

- **Tareas Programadas:** Configurar tareas programadas (scheduled jobs) en BigQuery que ejecuten el proceso de actualización en un horario regular (por ejemplo, cada noche).
- **Monitoreo:** Registrar todas las operaciones de inserción, actualización, y eliminación para mantener la integridad y calidad de los datos.

Flujo de Actualización de Pipeline:

Fuente de Datos → Preprocesamiento → Actualización de Tablas de Dimensiones (*category_dim, subcategory_dim, product_dim, customer_dim, market_dim, competence, order_dim*) → Actualización de la Tabla de Hechos (*sales_superstore*)

Beneficios de Este Pipeline:

- **Integridad de Datos:** Se asegura de que todas las claves foráneas en la tabla de hechos estén presentes en las tablas de dimensiones, manteniendo la consistencia.

- **Automatización:** Reduce el esfuerzo manual y la posibilidad de errores.
- **Escalabilidad:** Puede adaptarse a volúmenes crecientes de datos.

11. Unir tablas

- Con los comandos SELECT, LEFT JOIN, se creó una tabla para el análisis exploratorio a partir de la estructura de tablas de dimensiones y de hechos, seleccionando variables que contengan información relevante.

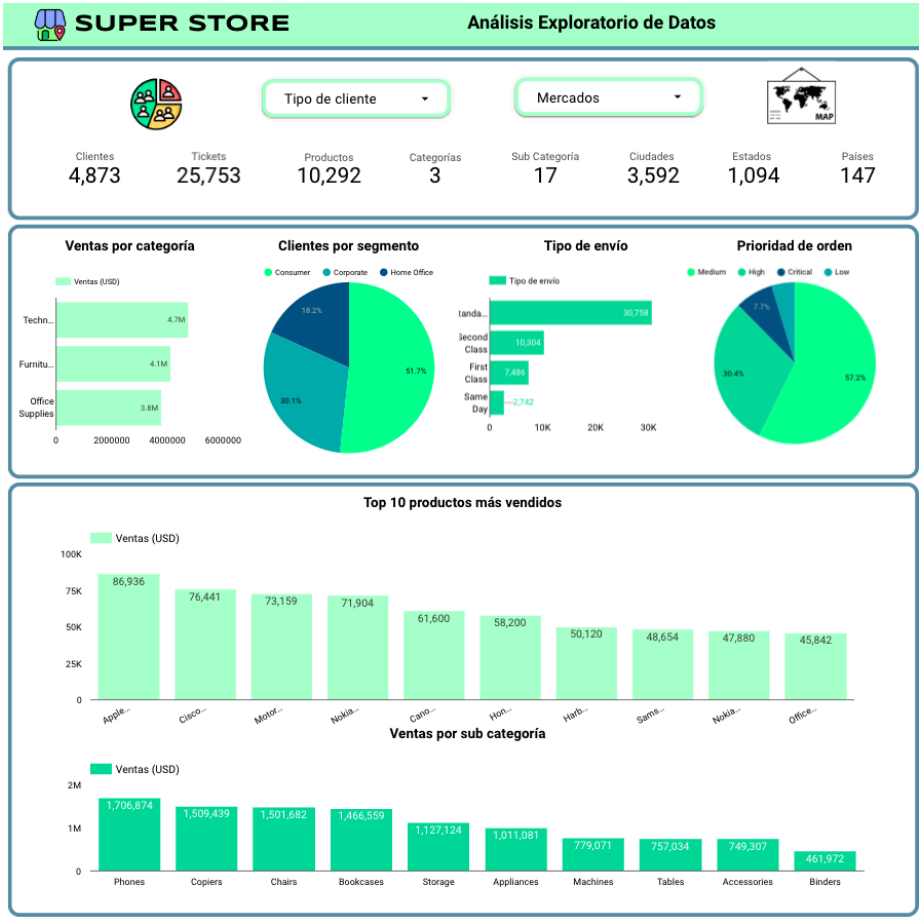
2.2 Hacer un análisis exploratorio

12. Agrupar datos según variables categóricas

- Se conectaron los datos a looker studio desde Bigquery.

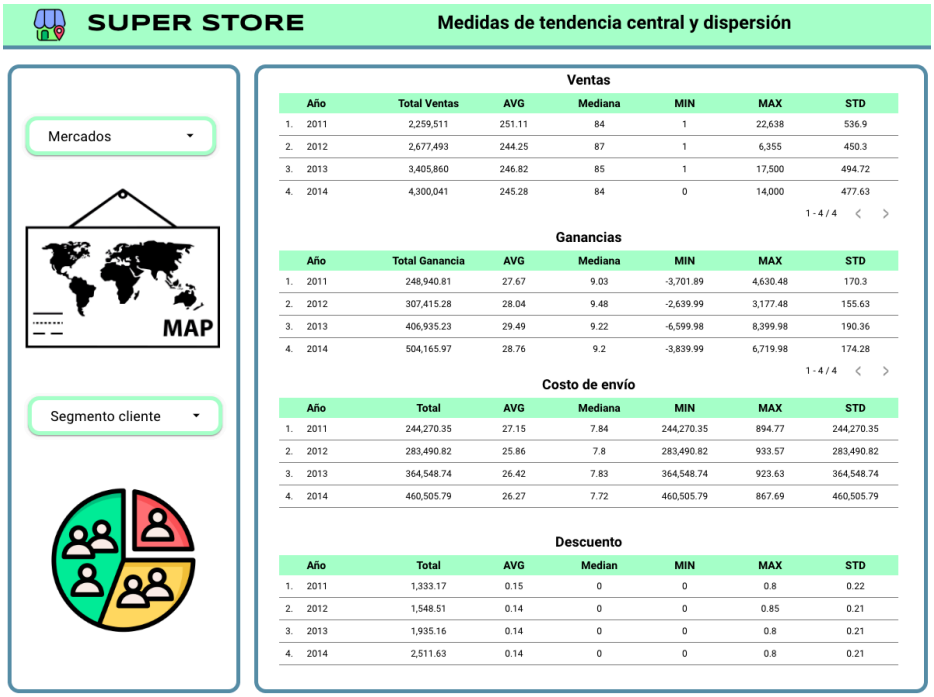
13. Visualizar las variables categóricas

- Se realizaron gráficos de barras, multivariable y de pastel para la visualización de variables y exploración de datos en looker studio.
- Se crearon drop-down list para filtrar y explorar la información y se agregaron score cards con datos relevantes.



14. Aplicar medidas de tendencia central y de dispersión

- Se crearon tablas en looker studio con las medidas de tendencia central y de dispersión (media, promedio, rango, desviación estándar) de las variables *sales, profit, shipping_cost, discount* para explorar los datos.



Ventas

	Año	Total Ventas	AVG	Mediana	MIN	MAX	STD
1.	2011	2,259,511	251.11	84	1	22,638	536.9
2.	2012	2,677,493	244.25	87	1	6,355	450.3
3.	2013	3,405,860	246.82	85	1	17,500	494.72
4.	2014	4,300,041	245.28	84	0	14,000	477.63

1 - 4 / 4

Ganancias

	Año	Total Ganancia	AVG	Mediana	MIN	MAX	STD
1.	2011	248,940.81	27.67	9.03	-3,701.89	4,630.48	170.3
2.	2012	307,415.28	28.04	9.48	-2,639.99	3,177.48	155.63
3.	2013	406,935.23	29.49	9.22	-6,599.98	8,399.98	190.36
4.	2014	504,165.97	28.76	9.2	-3,839.99	6,719.98	174.28

1 - 4 / 4

Costo de envío

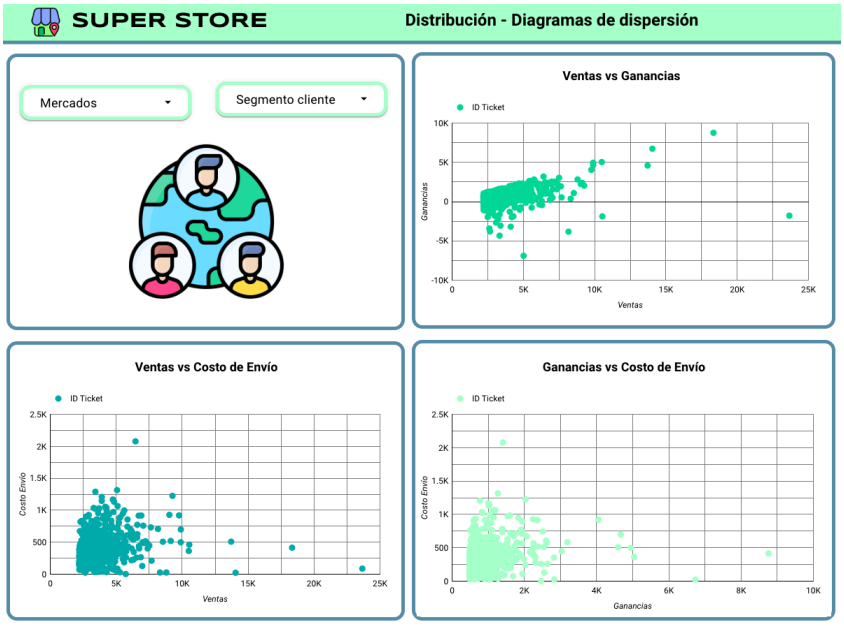
	Año	Total	AVG	Mediana	MIN	MAX	STD
1.	2011	244,270.35	27.15	7.84	244,270.35	894.77	244,270.35
2.	2012	283,490.82	25.86	7.8	283,490.82	933.57	283,490.82
3.	2013	364,548.74	26.42	7.83	364,548.74	923.63	364,548.74
4.	2014	460,505.79	26.27	7.72	460,505.79	867.69	460,505.79

Descuento

	Año	Total	AVG	Median	MIN	MAX	STD
1.	2011	1,333.17	0.15	0	0	0.8	0.22
2.	2012	1,548.51	0.14	0	0	0.85	0.21
3.	2013	1,935.16	0.14	0	0	0.8	0.21
4.	2014	2,511.63	0.14	0	0	0.8	0.21

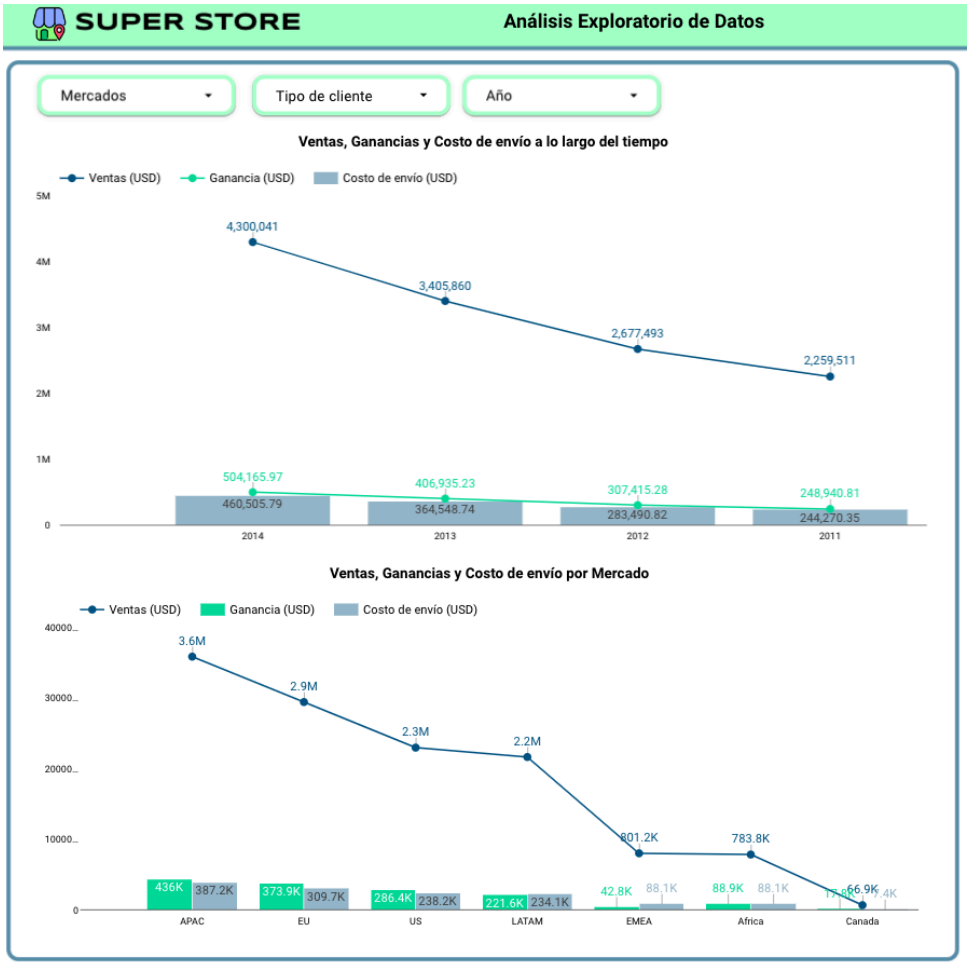
15. Visualizar distribución

- Se crearon gráficos de dispersión para observar la distribución de los datos.



16. Visualizar el comportamiento de los datos a lo largo del tiempo

- Se crearon gráficos de línea para observar el comportamiento de los datos a lo largo del tiempo.



2.3 Resultados y Conclusiones:

Diseño, creación y programación de estructura de base de datos

- La creación del esquema de datos basado en tablas de hechos y dimensiones permitirá centralizar toda la información relevante de las ventas, productos, clientes, y mercados en un solo lugar, facilitando el análisis de la información de forma precisa y rápida.
- La tabla de hechos (*sales_superstore*) almacena grandes volúmenes de datos transaccionales, mientras que las tablas de dimensiones (*customers_dim, products_dim, markets_dim, y order_dim*) contienen información descriptiva. Este diseño permite realizar consultas optimizadas y obtener resultados de manera más eficiente.
- Con el diseño de tablas de hechos y dimensiones, se podrán realizar análisis a nivel de detalle y en diferentes dimensiones, como por ejemplo, analizar ventas por producto, cliente, segmento, mercado, y período de tiempo.
- La implementación de un pipeline de actualización permitirá que las tablas se mantengan actualizadas de manera regular (por ejemplo, diariamente, semanalmente) con la última información, asegurando la disponibilidad de datos recientes para la toma de decisiones.
- El diseño de tablas de hechos y dimensiones que se implementará sigue un enfoque de esquema estrella, donde la tabla de hechos (*sales_superstore*) se conecta con múltiples tablas de dimensiones (*customers_dim, products_dim*, *orders_dim, markets_dim*). Este esquema es ideal para el análisis y la creación de dashboards, ya que permite acceder a los datos de forma simple y rápida.
- La tabla de hechos *sales_superstore* tiene como clave primaria el *id_ticket*, lo que garantiza la unicidad de las transacciones. Las llaves foráneas conectan la tabla de hechos con las tablas de dimensiones, asegurando que se pueda hacer un análisis detallado utilizando los atributos de las dimensiones.
- El proceso de ETL debe estar diseñado para extraer datos de las fuentes originales, transformarlos adecuadamente (limpieza de datos, cálculo de indicadores, generación de claves, etc.), y cargarlos en las tablas del esquema estrella. El pipeline de actualización debe manejar estos pasos automáticamente, evitando errores manuales y garantizando la integridad de los datos.
- La implementación de un pipeline de actualización permitirá actualizar las tablas de hechos y dimensiones periódicamente, asegurando que los datos reflejen la situación actual del negocio.

Análisis Exploratorio

- El análisis muestra que el segmento de clientes más grande es el consumidor individual (51.7%), seguido por los clientes corporativos (30.1%) y los de oficina en casa (18.2%). Esto indica que la mayoría de las ventas provienen de clientes particulares, lo que sugiere la necesidad de diseñar estrategias y promociones que atraigan y retengan a este tipo de clientes.
- El top 10 de productos más vendidos indica que los clientes tienen una alta demanda por tecnología (como smartphones y productos de Apple), seguido de productos de oficina. Esto muestra que las estrategias de marketing deben centrarse en destacar estos productos y lanzar ofertas específicas para fomentar la lealtad del cliente en estas categorías.
- La gráfica de "Ventas vs Ganancias por Mercado" revela que las ventas y ganancias varían considerablemente según el mercado. APAC (Asia-Pacífico) lidera en ventas, mientras que mercados como África y Canadá tienen menor participación. Esto sugiere que las estrategias de expansión deben enfocarse en fortalecer la presencia en mercados de alto rendimiento y considerar tácticas específicas para impulsar ventas en mercados con menor desempeño.
- Hay una tendencia creciente en las ventas y ganancias a lo largo de los años, alcanzando su punto máximo en 2014. Este aumento constante refleja un buen desempeño comercial durante estos años, lo cual es positivo para el negocio.

2.4 Recomendaciones:

Diseño, creación y programación de estructura de base de datos

- Es recomendable mantener el ID único (*id_ticket*), este campo se debe generar automáticamente en el proceso de ETL para garantizar que cada transacción sea única. También, se recomienda que las tablas de dimensiones mantengan sus propios identificadores (*customer_id, product_id, market_id,* etc.).
- Dado el volumen de datos y la necesidad de tomar decisiones basadas en información actual, se recomienda que el pipeline actualice las tablas diariamente para reflejar las ventas más recientes. Sin embargo, algunas dimensiones como *products_dim* o *customers_dim* pueden actualizarse semanal o mensualmente si su información no cambia con frecuencia.
- El pipeline debe extraer solo los datos nuevos o modificados desde la última actualización para minimizar el tiempo y los recursos necesarios.
- Se debe incluir un paso de validación para verificar que los datos cargados en las tablas son correctos y completos, evitando inconsistencias.

Análisis Exploratorio de Datos:

- Dado que el segmento "Consumer" representa la mayor parte de los clientes, es recomendable focalizar campañas de marketing y promociones hacia este grupo para maximizar las ventas. Sin embargo, no se debe descuidar a los segmentos "Corporate" y "Home Office", ya que también representan una parte significativa de los ingresos. Personalizar ofertas para estos segmentos podría impulsar un aumento de ventas adicionales.
- Dado que ciertos productos tienen una alta demanda, es recomendable mantener un inventario adecuado para evitar agotamientos de stock y perder ventas. También se pueden ofrecer paquetes promocionales, descuentos o ventas cruzadas con estos productos para maximizar los ingresos.
- Se debe analizar detalladamente los mercados APAC y EU para identificar factores que impactan negativamente las ganancias (por ejemplo, altos costos de envío o bajos precios de venta). Considerar ajustar precios, revisar costos logísticos y desarrollar estrategias de entrada de productos de mayor margen en estos mercados. Para los mercados de menor venta, como África y Canadá, evaluar si una estrategia de expansión con productos de alta demanda puede ser rentable.
- Invertir en estrategias de marketing y promoción enfocadas en la categoría de tecnología, ya que es la que genera más ventas. Al mismo tiempo, explorar oportunidades para impulsar ventas en las categorías menos dominantes, como "Office Supplies", mediante ofertas especiales o campañas dirigidas.
- Optimizar Costos: Reducir los costos de envío, renegociar con proveedores y optimizar la cadena de suministro.
- Mejorar la rentabilidad: Ajustar los precios y enfocarse en productos de mayor margen en mercados específicos.
- Expandir Estrategias de Venta: Crear estrategias de venta cruzada, promociones y campañas personalizadas basadas en segmentos de clientes y productos más vendidos.
- Análisis Continuo: Monitorear las tendencias de ventas y costos regularmente para ajustar las estrategias de forma dinámica y mantener un crecimiento sostenible.

La implementación de un sistema de tablas de hechos y dimensiones eficiente, junto con un pipeline de actualización automatizado, proporcionará a **Super Store** una estructura sólida para gestionar grandes volúmenes de datos y facilitar el análisis de la información de ventas, productos, clientes y mercados, impulsando así una toma de decisiones más efectiva y basada en datos.

Limitaciones/Próximos pasos:

- Implementar un sistema de tablas de hechos y dimensiones junto con un pipeline de actualización efectivo ofrecerá a Super Store una base sólida para el análisis de datos. Sin embargo, se deben abordar las limitaciones existentes, como la complejidad del proceso ETL, la gestión de grandes volúmenes de datos, y la necesidad de mantener una alta calidad de datos. Los próximos pasos se centran en optimizar el pipeline, ajustar la frecuencia de actualización, escalar la infraestructura de datos, y personalizar el dashboard para satisfacer las necesidades del negocio.

Enlaces de interés:

Dashboard

https://lookerstudio.google.com/reporting/58c871d4-115c-4fec-b35d-976aae6fcd3a