MySQL

ARQUITECTURA DE DATOS

Actualmente, el sistema utiliza el atributo zip_code_prefix como un eje común entre las entidades customers, geolocation y sellers. Sin embargo, al intentar formalizar esta relación mediante la implementación de Primary Keys (PK) y Foreign Keys (FK) en MySQL, se han identificado tres obstáculos críticos que impiden la consistencia de los datos:

- Entidad customers (Integridad Referencial): La creación de la PK se ve interrumpida por la presencia de datos huérfanos. En MySQL, no es posible establecer una relación de dependencia si existen registros en la tabla hija que no tienen una correspondencia válida en la tabla padre, violando las restricciones de integridad.
- Entidad geolocation (Unicidad): El intento de definir zip_code_prefix como PK falla debido a la existencia de valores duplicados. Por definición, una Clave Primaria requiere que cada registro sea único e irrepetible, condición que actualmente no se cumple en este set de datos.
- Entidad sellers (Indexación): Se presenta una deficiencia en la estructura de índices y referencias. Para que una relación sea eficiente y válida, cada entidad debe poseer una referencia clara y un índice optimizado que permita la vinculación con el resto del modelo.

Solución Propuesta

Para resolver estas inconsistencias, se ha determinado la creación de una tabla de referencia maestra basada en la entidad geolocation. Este proceso implica un saneamiento previo de los datos (eliminación de duplicados) para consolidar a geolocation como la fuente de verdad. Una vez normalizada, permitirá la asignación exitosa de la PK y facilitará la interconexión relacional con customers y sellers, garantizando así la robustez del modelo de datos.

```
-- Creamos la tabla maestra de geolocalización

CREATE TABLE IF NOT EXISTS olistdb.geo_referencia AS

SELECT

    geolocation_zip_code_prefix,

    AVG(geolocation_lat) AS lat_promedio,

    AVG(geolocation_lng) AS lng_promedio,

    ANY_VALUE(geolocation_city) AS geolocation_city,

    ANY_VALUE(geolocation_state) AS geolocation_state

FROM olistdb.geolocation

GROUP BY geolocation_zip_code_prefix;
```

```
-- Podemos poner la Primary Key (es una sentencia aparte)
ALTER TABLE olistdb.geo_referencia
ADD PRIMARY KEY (geolocation_zip_code_prefix);
-- Creamos índices para que MySQL permita la relación
CREATE INDEX idx_seller_zip ON olistdb.sellers (seller_zip_code_prefix);
CREATE INDEX idx_customer_zip ON olistdb.customers (customer_zip_code_prefix);
SET FOREIGN_KEY_CHECKS = 0;
-- Conectamos Sellers
ALTER TABLE olistdb.sellers
ADD CONSTRAINT fk_sellers_geo_ref
FOREIGN KEY (seller_zip_code_prefix) REFERENCES olistdb.geo_referencia(geolocation_zip_code_prefix);
-- Conectamos Customers
ALTER TABLE olistdb.customers
ADD CONSTRAINT fk_customers_geo_ref
FOREIGN KEY (customer_zip_code_prefix) REFERENCES olistdb.geo_referencia(geolocation_zip_code_prefix);
SET FOREIGN_KEY_CHECKS = 1;
```

2. Automatización y Limpieza Avanzada (Python / Jupyter)

Con la base de datos estructurada, se implementó un flujo de trabajo en Python para manejar el volumen masivo de datos (más de 112k filas) que Excel no procesa con la misma eficiencia.

- **Proceso:** Se utilizó un código de conexión directa a SQL para cargar los datos. Esto permite que, si los archivos origen (CSV) se actualizan, el sistema se refresque automáticamente.
- **Tratamiento:** Se realizó una limpieza automatizada, aislando *outliers* y segmentando categorías.
- **Resultado de Elasticidad: -19.57**. Este valor representa la sensibilidad estadística "pura" de la serie histórica tras un filtrado técnico riguroso.

3. Auditoría y Validación de Datos (Excel)

Se utilizó Excel como una capa de validación rápida para entender la distribución del volumen de ventas.

- **Análisis:** Al aplicar filtros manuales (ventas superiores a 277.40), se identificaron las 8,427 filas más críticas para el negocio.
- **Resultado de Elasticidad: -28.71**. Al no contar con la limpieza profunda de Python y estar sujeto a picos extremos de los datos crudos, este valor sirve para identificar la volatilidad máxima, aunque no es el número final para la toma de decisiones.

4. Dashboard e Insights de Negocio (Power BI)

La etapa final aterrizó la estadística en una herramienta de visualización estratégica para la Gerencia.

- **Modelado DAX:** Se creó una medida de "Elasticidad Real" que filtra el ruido y se enfoca en el comportamiento de categorías específicas.
- **Resultado de Elasticidad: -3.93**. Este es el número más cercano a la realidad operativa; es una medida refinada que permite proyecciones financieras realistas.

Conclusión del Análisis de Elasticidad

Tras auditar el comportamiento desde tres ángulos distintos, la conclusión para la toma de decisiones es contundente:

1. **Convergencia de Resultados:** Aunque los valores numéricos varían (**-3**, **-19**, **-28**) debido al nivel de limpieza de cada herramienta, todos coinciden en un punto crítico: **el resultado es siempre negativo y mayor a 1**.
2. **Diagnóstico:** La categoría es **altamente elástica**. Esto significa que la demanda es extremadamente sensible a las variaciones de precio.
3. **Recomendación Estratégica:** No se recomienda realizar aumentos de precio de lista de manera aislada. Debido a la alta volatilidad detectada, cualquier incremento debe ser compensado con estrategias promocionales o de valor agregado para evitar una caída drástica en el volumen de ventas.