

Prueba de Homologación Python: Análisis Estratégico de Datos para "Global-Tech Retail"

Contexto del Problema:

Eres el nuevo Analista de Datos Senior de "Global-Tech Retail", una empresa multinacional que vende productos tecnológicos a través de una plataforma online. La dirección ha notado un crecimiento en las ventas, pero también un aumento en los costos operativos y en las devoluciones de productos. No tienen claro qué productos son los más rentables, qué campañas de marketing son efectivas, ni en qué regiones deberían enfocar sus esfuerzos.

Tu misión es realizar un análisis de datos exhaustivo para responder a preguntas de negocio clave y presentar tus hallazgos y recomendaciones estratégicas a la junta directiva (en este caso, tu mentor técnico).

Objetivo General:

Realizar un análisis de datos completo, desde la ingesta y limpieza hasta la visualización y la generación de insights, para proporcionar a "Global-Tech Retail" recomendaciones estratégicas basadas en evidencia que permitan optimizar su catálogo de productos, mejorar la rentabilidad y enfocar sus esfuerzos de marketing para el próximo año fiscal.

Fase 1: Configuración del Entorno y Carga de Datos

1. **Entorno de Trabajo:** Configura un notebook. En la primera celda, utiliza comentarios para documentar el propósito del proyecto, tu nombre y la fecha.
2. **Carga de Datos:** La empresa te proporciona tres fuentes de datos desestructuradas:
 - **ventas.csv:** Un archivo con todas las transacciones. Contiene ID_Venta, ID_Producto, ID_Cliente, Fecha_Venta, Cantidad, Precio_Unitario.
 - **productos.csv:** Un archivo CSV con el catálogo de productos. Incluye ID_Producto, Nombre_Producto, Categoria, Costo_Unitario.
 - **clientes.db:** Una pequeña base de datos SQLite que contiene la información demográfica de los clientes (ID_Cliente, Ciudad, Pais, Fuente_Marketing). Deberás usar pandasql o la librería sqlite3 para leer estos datos.
3. **Exploración Inicial:** Realiza una inspección básica de cada DataFrame cargado. Obtén un primer vistazo de la estructura, los tipos de datos y las estadísticas descriptivas. Documenta tus observaciones iniciales.

Fase 2: Limpieza y Transformación de Datos

1. Manejo de Datos Faltantes y Duplicados:

- Inspecciona cada DataFrame en busca de valores nulos y datos duplicados.
- Define y aplica una estrategia para manejarlos (ej. eliminar filas, imputar valores con la media/mediana/moda). Justifica cada decisión en los comentarios de tu código.

2. Corrección de Tipos de Datos: Asegúrate de que cada columna tenga el tipo de dato correcto. La columna Fecha_Venta es crucial y debe ser de tipo datetime.

3. Ingeniería de Características (Creación de Nuevas Columnas):

- Crea una columna Ingreso_Total ($\text{Cantidad} * \text{Precio_Unitario}$).
- **Unión de Datos:** Combina los tres DataFrames en un único DataFrame maestro que contenga toda la información relevante por transacción. Utiliza las funciones merge o join de Pandas.
- Una vez unidos, crea una columna Ganancia ($\text{Ingreso_Total} - (\text{Cantidad} * \text{Costo_Unitario})$).
- Extrae el Mes y el Año de la Fecha_Venta en nuevas columnas para facilitar el análisis temporal.

4. Función Personalizada: Define una función en Python que clasifique a los clientes en categorías (Nuevo, Recurrente, Leal) basándose en su número de compras. Aplica esta función al DataFrame maestro usando .apply().

Fase 3: Análisis Exploratorio y Visualización

Utiliza el DataFrame maestro y limpio para responder a las siguientes preguntas de negocio. **Cada respuesta debe estar soportada por al menos una visualización clara y bien etiquetada.**

1. Análisis de Productos:

- ¿Cuáles son los 10 productos más vendidos en términos de cantidad?
- ¿Cuáles son los 10 productos que generan mayor ingreso? ¿Y mayor ganancia? ¿Son los mismos?
- ¿Qué categorías de productos son las más populares?

2. Análisis de Ventas y Tiempo:

- ¿Cómo ha sido la evolución de los ingresos totales mes a mes? ¿Hay alguna estacionalidad evidente? (Usa un gráfico de líneas).
- ¿Qué día de la semana se registran más ventas?

3. Análisis Geográfico y de Clientes:

- ¿Cuáles son las 5 ciudades con el mayor volumen de ventas? (Usa un gráfico de barras).

- ¿Cómo se distribuye la ganancia por país?
- 4. Análisis de Marketing:**
- ¿Qué Fuente_Marketing (ej. 'Redes Sociales', 'Búsqueda Orgánica', 'Email') atrae a los clientes que más gastan?

Fase 4: Implementación Avanzada

1. **Programación Orientada a Objetos (POO):** Refactoriza una parte de tu análisis. Crea una clase de Python llamada AnalizadorVentas.
 - El constructor (`__init__`) debe recibir el DataFrame maestro.
 - Crea métodos dentro de la clase para realizar análisis específicos, por ejemplo:
 - `get_top_productos(metrica='Ganancia', n=10)`
 - `plot_ventas_mensuales()`
 - `generar_reporte_por_pais(pais='todos')`
 - Instancia tu clase y utiliza sus métodos para volver a generar al menos dos de los análisis de la Fase 3.
2. **Manejo de Excepciones:** Envuelve el código de carga de archivos (Fase 1) en bloques try-except.

Entregables y Sustentación:

1. **Notebook de Google Colab (.ipynb):**
 - El notebook debe estar perfectamente organizado, con títulos en celdas de texto para cada fase y sub-sección.
 - El código debe estar limpio, ser eficiente y, sobre todo, **extensamente comentado**. Los comentarios deben explicar la lógica detrás de las decisiones de limpieza y los pasos del análisis.
 - Todas las celdas deben haber sido ejecutadas y mostrar sus salidas (tablas, gráficos, etc.).
2. **Presentación Ejecutiva (Sustentación Oral):**
 - Prepara una presentación de 10-15 minutos dirigida a la "junta directiva".
 - **No te centres en el código.** Enfócate en los hallazgos.
 - Presenta tus visualizaciones más impactantes.
 - Concluye con **3 a 5 recomendaciones estratégicas claras y accionables** basadas en tu análisis.