## 📝 Informe Final SynthData

El objetivo de este informe es resumir los pasos realizados en el análisis de los datasets de ventas, clientes y marketing, desde la carga inicial hasta la integración de datos para la obtención de resultados.

### 🧩 Etapa 1: Recopilación y Preparación de Datos

**1. Carga de datos:** Se cargaron los datasets ventas.csv, clientes.csv, y marketing.csv en DataFrames de pandas. Se verificaron las dimensiones iniciales de cada DataFrame:

* ventas: (3035, 6)
* clientes: (567, 5)
* marketing: (90, 6)

**2. Análisis exploratorio inicial:** Se realizó un análisis exploratorio inicial utilizando la función eda para comprender la estructura, tipos de datos, y presencia de valores nulos en cada DataFrame.

* **Ventas:** Se identificaron 2 valores nulos en las columnas 'precio' y 'cantidad'. La columna 'precio' fue identificada como tipo object debido a la presencia del símbolo '$'. Se requiere convertir la columna fecha en datatime.
* **Clientes:** No se encontraron valores nulos ni duplicados.
* **Marketing:** No se encontraron valores nulos ni duplicados. Se identificó la necesidad de convertir las columnas de fecha a tipo datetime.

**3. Calidad de los datos:** Se utilizó la función calidad para verificar la presencia de valores nulos y duplicados.

* **Ventas:** Se confirmaron los 2 nulos en 'precio' y 'cantidad'. Se encontraron 70 filas completamente duplicadas y 70 duplicados en la clave 'id\_venta'.
* **Clientes:** No se encontraron nulos ni duplicados.
* **Marketing:** No se encontraron nulos ni duplicados.

### 🧹 Etapa 2: Preprocesamiento y Limpieza de Datos

**4. Limpieza de datos:** Se crearon copias de los DataFrames originales (ventas\_clean, clientes\_clean, marketing\_clean).

* Se eliminaron las 35 filas completamente duplicadas del DataFrame ventas\_clean.
* Se normalizaron las columnas de texto en los tres DataFrames (ventas\_clean, clientes\_clean, marketing\_clean) convirtiéndolas a mayúsculas y eliminando espacios y caracteres no deseados.
* Se convirtieron las columnas de fecha en ventas\_clean y marketing\_clean a tipo datetime.
* Se normalizó la columna 'precio' en ventas\_clean, eliminando el símbolo '$' y convirtiéndola a tipo float.
* Se convirtió la columna 'cantidad' en ventas\_clean a tipo entero (Int64).
* Se guardaron los DataFrames limpios como archivos CSV con el sufijo "\_NM".

**Reporte Global luego de la limpieza de datos:** Se generó un reporte comparativo mostrando la situación de los DataFrames antes y después de la limpieza:

| **Dataset** | **Filas** | **Columnas** | **Nulos totales** | **Duplicados** |
| --- | --- | --- | --- | --- |
| VENTAS Original | 3035 | 6 | 4 | 70 |
| CLIENTES Original | 567 | 5 | 0 | 0 |
| MARKETING Original | 90 | 6 | 0 | 0 |
| VENTAS Copia | 3000 | 6 | 4 | 0 |
| CLIENTES Copia | 567 | 5 | 0 | 0 |
| MARKETING Copia | 90 | 6 | 0 | 0 |

Se observa que la limpieza eliminó los duplicados en el DataFrame de ventas.

### 5. Transformación de datos: Productos de Alto Rendimiento

* Se detectó la columna de producto (producto) en el DataFrame ventas\_clean.
* Se calculó el ingreso por registro multiplicando 'precio' por 'cantidad'.
* Se agregaron las métricas por producto (ingreso\_total, unidades, precio\_promedio, registros) utilizando groupby y agg.
* Se calculó el percentil 80 del ingreso\_total (52.518,85).
* Se filtraron los productos con un ingreso\_total mayor o igual al percentil 80 para identificar los productos de alto rendimiento.

Se identificaron los productos de alto rendimiento:

| **producto** | **ingreso\_total** | **unidades** | **precio\_promedio** | **registros** |
| --- | --- | --- | --- | --- |
| LÁMPARA DE MESA | 82276.38 | 1112 | 72.72 | 176 |
| AURICULARES | 74175.58 | 958 | 76.30 | 143 |
| MICROONDAS | 72562.89 | 912 | 79.18 | 135 |
| CAFETERA | 59607.31 | 765 | 79.05 | 117 |
| CUADRO DECORATIVO | 54297.6 | 726 | 74.58 | 100 |
| SMARTPHONE | 54132.44 | 665 | 81.40 | 101 |

### 

### 6. Agregación por Categoría

* Se detectó la columna de categoría (categoria) en el DataFrame ventas\_clean.
* Se aseguró la existencia de la columna ingreso.
* Se agregaron las métricas por categoría (ingreso\_total, unidades, ventas, precio\_promedio) utilizando groupby y agg.
* Se calculó el ticket\_promedio\_por\_venta para cada categoría.

Se obtiene el resumen por categoría, ordenado por ingreso\_total:

| **categoria** | **ingreso\_total** | **unidades** | **ventas** | **precio\_promedio** | **ticket\_promedio\_por\_venta** |
| --- | --- | --- | --- | --- | --- |
| ELECTRODOMÉSTICOS | 505299.63 | 6592 | 1000 | 76.52 | 505.3 |
| ELECTRÓNICA | 482577.8 | 6413 | 999 | 75.25 | 483.06 |
| DECORACIÓN | 479216.09 | 6490 | 1001 | 74.10 | 478.74 |

### 

### 7. Integración de datos (Opcional)

* Se intentó combinar los DataFrames ventas\_clean y marketing\_clean.
* Se detectó la clave común: producto.
* Se identificó una cardinalidad de "m:m" (muchos a muchos) entre los DataFrames en la clave producto.
* Se realizó un LEFT JOIN utilizando la clave producto, resultando en un DataFrame ventas\_marketing con 9000 filas. Todas las filas en el DataFrame unificado se originaron de la combinación de ambos DataFrames.
* Se realizó un resumen del ingreso total por campaña (id\_campanha) y por canal (canal).
* Se observó que el ingreso total es el mismo para todos los canales y campañas, lo cual puede indicar que la unión por producto no refleja la contribución individual de cada campaña o canal a las ventas específicas.

Este informe proporciona una visión general de los pasos de preprocesamiento, limpieza y análisis inicial realizados en los datasets, así como los hallazgos clave sobre el rendimiento de productos y categorías.