**Proyecto 1 – Análisis de Ventas: Superstore**

**Objetivo:**

Mi objetivo con este proyecto es analizar los datos históricos de ventas de una empresa de comercio minorista para identificar cuáles son los productos más vendidos, evaluar qué regiones generan mayor rentabilidad y entender mejor el comportamiento de los clientes. También quiero examinar cómo influyen las promociones y descuentos en las ganancias finales.

Para lograrlo, decidí trabajar por etapas y utilizar distintas herramientas según la necesidad de cada fase del análisis:

* **Primero usé Excel** para hacer una limpieza inicial, corrigiendo errores, estandarizando las fechas y dejando el archivo listo para trabajar.
* Luego, **cargué los datos en SQL Server**, donde validé la estructura, corregí tipos de datos y realicé consultas para obtener métricas generales como ventas totales, ganancias, top de productos, etc.
* En la siguiente etapa, **estoy utilizando Python** para aplicar análisis más profundos, como la segmentación de clientes, clasificación de productos según su rentabilidad y análisis del impacto de los descuentos.
* Finalmente, **usaré Power BI** para crear dashboards interactivos que resuman los principales hallazgos y permitan tomar decisiones más estratégicas en las áreas de ventas, marketing y logística.

Mi enfoque ha sido trabajar cada parte con un propósito claro, integrando las herramientas de forma ordenada para obtener resultados sólidos que puedan aplicarse en el mundo real.

**Sprint 1 – Preparación y limpieza de datos**

**Duración:** 4 días

**Objetivo:**

Preparar y limpiar la base de datos de ventas para garantizar su calidad y estructura, dejándola lista para su integración en SQL Server. Esta etapa incluyó la identificación y corrección de errores, normalización de campos clave y estandarización de formatos necesarios para el análisis posterior.

**Historias de usuario:**

* Como analista de datos, quiero obtener y revisar los datos para iniciar el análisis con insumos confiables.
* Como analista de datos, quiero limpiar y normalizar la información para asegurar su consistencia antes de cargarla en SQL.

**Tareas:**

1. **Descarga y renombrado del archivo:**

* Se descargó el dataset **Sample - Superstore.csv** desde Kaggle.

1. **Limpieza estructural en Excel:**

* Eliminación de filas con valores vacíos en campos críticos como **Sales**, **Quantity**, **Discount** y **Profit**.
* Revisión de duplicados: se permitió la repetición de **Order ID** siempre que estuviera asociada a distintos **Product ID**.
* Corrección de errores en fechas, originados por la coexistencia de formatos mixtos (texto y fecha).

1. **Normalización de fechas mediante fórmula avanzada en Excel:**

* Se utilizó la siguiente fórmula para convertir fechas en distintos formatos a un formato uniforme **dd/mm/yyyy**:

=SI(NO(ESNUMERO(C2));

SI.ERROR(

SI(

VALOR(EXTRAE(C2;ENCONTRAR("/";C2)+1;ENCONTRAR("/";C2;ENCONTRAR("/";C2)+1)-ENCONTRAR("/";C2)-1))>12;

TEXTO(FECHA(

VALOR(DERECHA(C2;4));

VALOR(IZQUIERDA(C2;ENCONTRAR("/";C2)-1));

VALOR(EXTRAE(C2;ENCONTRAR("/";C2)+1;ENCONTRAR("/";C2;ENCONTRAR("/";C2)+1)-ENCONTRAR("/";C2)-1))

);"dd/mm/yyyy");

TEXTO(FECHA(

VALOR(DERECHA(C2;4));

VALOR(EXTRAE(C2;ENCONTRAR("/";C2)+1;ENCONTRAR("/";C2;ENCONTRAR("/";C2)+1)-ENCONTRAR("/";C2)-1));

VALOR(IZQUIERDA(C2;ENCONTRAR("/";C2)-1))

);"dd/mm/yyyy")

);

""

);

TEXTO(C2;"dd/mm/yyyy")

)

1. **Estandarización de campos de fecha:**

* Se aplicó el formato **dd/mm/yyyy** a **Order Date** y **Ship Date** para evitar errores regionales al importarlos a SQL Server.

1. **Corrección de códigos postales:**

* Se normalizaron los valores de **Postal Code**, aplicando el formato personalizado **00000** para mantener los ceros a la izquierda.

1. **Formato monetario uniforme:**

* Se ajustó la visualización de las columnas **Sales** y **Profit** para que muestren solo dos decimales.

**Conclusión del Sprint:**

En este sprint trabajé en limpiar y preparar el archivo original Sample - Superstore.csv. Eliminé errores, normalicé fechas, corregí formatos y verifiqué campos clave. Al final, obtuve un archivo limpio y estructurado: **Ventas\_Separadas\_v1.csv**, ubicado en la ruta **\Datos\**, listo para ser cargado en SQL Server en el siguiente sprint.

**Sprint 2 – Carga y Exploración de Datos**

**Duración:** 3 días

**Objetivo:**

Cargar el archivo limpio **Ventas\_Separadas\_v1.csv** en SQL Server y realizar una primera revisión de la información, asegurando que los datos estén correctamente estructurados y sin inconsistencias. Esta etapa sienta las bases para análisis posteriores.

**Historias de usuario:**

* Como analista de datos*,* necesito cargar el dataset limpio a SQL Server para establecer la base del análisis.
* Como analista de datos*,* necesito validar la estructura y consistencia de los datos importados para asegurar su calidad.
* Como analista de datos*,* deseo explorar métricas generales para entender el comportamiento inicial de las ventas.

**Tareas:**

1. **Creación de la base de datos VentasSuperstore:**

Se creó una base de datos dedicada para almacenar y trabajar exclusivamente con el dataset limpio.

1. **Importación del archivo Ventas\_Separadas\_v1.csv:**

El dataset fue importado utilizando el asistente de importación de SQL Server.  
*Problema detectado:* El campo Product Name fue truncado inicialmente a 50 caracteres por configuración del asistente.  
*Solución:* Se editó manualmente el tipo de dato en el editor avanzado a VARCHAR(1024) para evitar pérdida de información.

1. **Validación de longitudes de texto:**

Se utilizaron funciones LEN() para identificar la longitud máxima de campos tipo texto. Esto permitió ajustar correctamente los tamaños con ALTER COLUMN para cada campo tipo VARCHAR.

1. **Ajuste de tipos de datos:**

Con base en las longitudes identificadas, se modificaron los tipos de datos de columnas texto, optimizando el almacenamiento y previniendo futuros errores de carga.

1. **Verificación del total de registros:**

Se confirmó que se cargaron correctamente 9986 registros, lo cual valida que el proceso fue exitoso.

1. **Validación de valores nulos:**

Se comprobó que no existen campos nulos en las columnas clave, lo cual asegura una buena calidad de los datos.

1. **Validación de duplicados:**

Se revisó si existían combinaciones repetidas de Order ID y Product ID.  
Resultado: 0 duplicados detectados.

1. **Prueba de conversión de tipos:**

Se verificó que todos los campos numéricos y de fecha se puedan convertir correctamente a sus tipos esperados (FLOAT, INT, DATE).  
Resultado: sin errores.

1. **Revisión del rango de fechas:**

Se identificó que el dataset abarca pedidos desde el 02/01/2014 hasta el 30/12/2017, útil para definir el marco temporal del análisis.

1. **Análisis exploratorio inicial (métricas generales):**

Se realizaron consultas para conocer el comportamiento básico de las ventas:

* Total de ventas: S/ 2,297,201.07
* Total de ganancias: S/ 286,397.79
* Top 10 productos más vendidos por monto total
* Ventas por categoría de producto
* Rentabilidad por región
* Comportamiento mensual de ventas, útil para detectar estacionalidades y tendencias

**Conclusión del Sprint:**

En este sprint cargué el archivo limpio **Ventas\_Separadas\_v1.csv**, ubicado en la ruta **\Datos\**, a la base de datos VentasSuperstore en SQL Server. Ajusté los tamaños de texto, corregí un error de truncamiento en Product Name y validé que no haya nulos, duplicados ni errores de tipo. También revisé el rango de fechas y realicé un análisis general de ventas y ganancias. Todo el proceso quedó documentado en el archivo **VentasSuperstore.sql**, ubicado en **\SQL\**, dejando listo el avance hacia Power BI en el siguiente sprint.

**Sprint 3 – Carga y Exploración de Datos**

**Duración:** 3 días

**Objetivo:**

Aplicar técnicas analíticas avanzadas en Python sobre la base de datos VentasSuperstore, con el fin de explorar patrones de comportamiento de los clientes, identificar productos rentables, evaluar la rentabilidad regional y analizar el impacto de los descuentos en las ganancias del negocio.

**Historias de usuario:**

* Como analista de datos, quiero identificar los clientes más valiosos según sus compras y rentabilidad.
* Como analista de datos, quiero clasificar los productos por su rendimiento económico para enfocar las estrategias de venta.
* Como analista de datos, deseo evaluar si los descuentos aplicados realmente mejoran las ventas o reducen la rentabilidad.

**Tareas:**

1. **Carga de datos desde SQL Server**

* Se utilizó pyodbc para conectarse al servidor y extraer la tabla **Ventas\_Separadas\_v1**.
* Se cargó la tabla en un DataFrame de pandas para su análisis posterior.
* Este paso permitió trabajar con datos reales y estructurados desde un entorno OLTP.

1. **Segmentación de clientes – RFM simplificado**

* Se agruparon las ventas por Customer ID y se calcularon tres métricas clave:
  + Total comprado (Sales)
  + Frecuencia de compra (número de pedidos distintos)
  + Última fecha de compra (recencia)
* Se creó el RFM Score combinando estas tres dimensiones.
* Se visualizó la cantidad de clientes por grupo de valor.
* Esto permitió distinguir clientes prioritarios y posibles inactivos.

1. **Clasificación de productos por rentabilidad**

* Se calculó el margen de ganancia (%) a partir de Profit / Sales.
* Se clasificaron los productos en 3 grupos:
  + **Estrella**: alto margen y alta venta
  + **Trampa**: alta venta, pero margen bajo o negativo
  + **Ocultos**: baja venta, pero muy rentables
* Se usaron gráficos de dispersión (scatterplot) para visualizar estas categorías.
* Esto permitió identificar productos estratégicos y oportunidades de mejora.

1. **Regiones más rentables**

* Se agruparon las ventas por Region y se calcularon:
  + Ganancia total
  + Margen de ganancia (%)
* Se visualizaron las regiones por rentabilidad total y por eficiencia en el margen.
* La región West fue la más rentable, y Central la menos eficiente.
* El análisis sirve para priorizar esfuerzos de expansión o ajustes en distribución.

1. **Impacto de los descuentos en la rentabilidad**

* Se analizó cómo varía la ganancia total y el margen (%) según el nivel de descuento.
* Se agruparon los datos por rangos de descuento (Discount) y se calculó la ganancia promedio.
* Se encontró que los descuentos altos afectan negativamente la rentabilidad, con márgenes incluso negativos en algunos casos.
* Los descuentos cercanos a 0.0 o 0.2 son más sostenibles.

**Conclusión del Sprint:**

Este sprint permitió profundizar en el análisis de datos con Python y generar hallazgos estratégicos:

* Se identificaron los clientes más valiosos mediante el modelo RFM.
* Se clasificaron los productos más rentables y riesgosos.
* Se detectaron las regiones con mayor retorno económico.
* Se evaluó de forma clara el impacto de los descuentos en la utilidad.

Todo el análisis quedó registrado en el cuaderno **Sprint3\_AnalisisAvanzado.ipynb**, disponible en la ruta: **\Python\Sprint3\_AnalisisAvanzado.ipynb**

**Sprint 4 – Visualización Interactiva y Presentación**

**Duración:** 3 días

**Objetivo:**

Transformar los análisis obtenidos en los sprints anteriores en visualizaciones interactivas utilizando Power BI, facilitando la interpretación y toma de decisiones por parte de los usuarios del negocio.

**Historias de usuario:**

* Como gerente comercial, quiero visualizar los productos más vendidos y rentables para priorizar mis esfuerzos de venta.
* Como analista, quiero identificar visualmente las regiones más rentables para planificar la distribución.
* Como equipo de marketing, deseo comprender el impacto de los descuentos sobre las utilidades para ajustar promociones futuras.
* Como usuario estratégico, necesito tableros que me permitan explorar los datos y generar insights en tiempo real.

**Tareas:**

1. **Preparación del dataset final para Power BI**

* Exportación desde Python de los archivos trabajados (clientes\_rfm.csv, productos\_rentabilidad.csv, regiones\_rentabilidad.csv, analisis\_descuentos.csv).
* Revisión de consistencia en nombres de columnas, tipos de datos y formato de fechas.

1. **Diseño del modelo de datos en Power BI**

* Conexión de Power BI a los archivos .csv.
* Organización del modelo tabular (sin relaciones complejas, ya que cada tabla responde a un análisis específico).

1. **Creación de dashboards visuales**

Se diseñaron visualizaciones interactivas que incluyen:

* Segmentación de clientes por monto total y perfil RFM.
* Relación entre descuentos y rentabilidad (ganancia y margen).
* Rentabilidad y volumen de ventas por región.
* Visualización del segmento RFM de cada cliente.
* Dispersión de productos por tipo: estrella, trampa, oculto, otros.

1. **Publicación y entrega**

* Archivo final guardado como: **\PowerBI\Proyecto\_Sprint4\_Analisis\_Ventas\_Superstore.pbix**
* Incluye diseño, colores, títulos explicativos y segmentaciones por cada tipo de análisis.

**Conclusión del Sprint:**

Este sprint consolidó todo el análisis previo en un conjunto de dashboards interactivos que permiten al equipo de negocio:

* Identificar los productos y clientes que generan mayor valor.
* Detectar regiones estratégicamente más rentables.
* Evaluar qué niveles de descuento son sostenibles.

Con ello, se cierra la fase de desarrollo analítico del proyecto y se deja preparado el entregable visual para la toma de decisiones.