Informe preprocesamiento de datos

Autor: Kristhian Santiago Palomino Fajardo
Fecha: 18 de marzo de 2025

Introducción

Este informe describe detalladamente el proceso seguido para la limpieza y transformación del conjunto de datos de ventas. Se explican los métodos empleados para manejar valores nulos, datos faltantes y duplicados, así como el proceso de normalización de las variables numéricas y la codificación de las variables categóricas. El objetivo es preparar los datos para el desarrollo del modelo predictivo, garantizando consistencia y calidad en el proceso de transformación.

Manejo de Valores Nulos, Datos Faltantes y Duplicados

Valores Nulos y Faltantes

- Verificación: Al cargar el dataset, se realizó un análisis de valores nulos utilizando la función df.isnull().sum().
- Resultado: No se encontraron valores nulos en ninguna de las columnas del dataset.
- Justificación: La ausencia de valores nulos indica que los datos ya estaban completos en el nivel de registro, lo que elimina la necesidad de imputación o eliminación. En casos donde se detecten valores faltantes, se podría aplicar imputación utilizando la media o la mediana para variables numéricas y la moda para variables categóricas, o eliminar filas si la proporción de datos faltantes es muy baja.

Duplicados

- Verificación: Se revisó la existencia de registros duplicados utilizando df.duplicated().sum().
- Resultado: No se detectaron registros duplicados.
- Justificación: La presencia de duplicados puede sesgar el análisis y el modelado; en este caso, al no existir duplicados, no fue necesario aplicar técnicas de eliminación. Si se hubieran encontrado, se habría optado por eliminarlos, ya que estos podrían representar entradas redundantes o errores en la captura de datos.

Transformación y Feature Engineering

Conversión de Fechas y Extracción de Características Temporales

- Conversión de Fecha: La columna Date se convierte a formato datetime utilizando pd.to_datetime(), lo cual es crucial para poder extraer información temporal y facilitar análisis de series de tiempo.
- Extracción de Variables:
  Se derivaron las siguientes características:
    - Mes: Permite identificar patrones estacionales mensuales.
    - Día_de_la_semana: Proporciona información sobre la variación de las ventas según el día (por ejemplo, mayor actividad en ciertos días).
    - Trimestre: Facilita el análisis de tendencias a nivel trimestral.
    - Day_of_Month: Puede capturar efectos asociados al final o inicio de mes.
    - Es_fin_de_semana: Indicador binario (1 si es sábado o domingo, 0 en caso contrario) para diferenciar días laborables de fines de semana.
- Justificación: Estas nuevas variables permiten capturar patrones estacionales y de comportamiento que pueden influir en las ventas, facilitando posteriormente la selección de características relevantes para el modelado.

Normalización y Estandarización de Variables Numéricas

- Proceso: Se aplicó la estandarización (utilizando StandardScaler) a las variables numéricas, en este caso, Unit_Price y Day_of_Month.
- Justificación: La estandarización transforma las variables para que tengan una media de 0 y una desviación estándar de 1, lo cual es fundamental para algoritmos sensibles a la escala (como regresión lineal, Lasso, Ridge y otros modelos basados en distancia).

Codificación de Variables Categóricas

- Técnica Utilizada: Se aplicó One-Hot Encoding a las variables categóricas, es decir, a Store, Category, Día_de_la_semana y Es_fin_de_semana.
- Justificación:
    - One-Hot Encoding:
      Esta técnica transforma cada categoría en una columna binaria. Se eligió esta técnica porque:
        - Permite que los algoritmos de aprendizaje automático trabajen con datos categóricos sin asumir un orden inherente.

        - Facilita la interpretación de los modelos lineales y no lineales, ya que cada dummy representa una categoría específica.
        - Dado que el número de categorías es manejable (por ejemplo, el día de la semana tiene 7 posibles valores), el aumento en la dimensionalidad es aceptable.
    - Manejo de Categorías Desconocidas: Se configuró el One-Hot Encoder con el parámetro handle_unknown='ignore' para evitar errores en el conjunto de test si se presentan categorías no vistas durante el ajuste.

Guardado del Preprocesador y Datos Preprocesados

- Preprocesador: El objeto preprocesador (un ColumnTransformer que integra la estandarización y el One-Hot Encoding) se guarda en la carpeta models/preprocessor en el archivo preprocessor.pkl utilizando joblib.dump().
- Dataset Preprocesado: Se guardaron los datos transformados en dos archivos CSV (uno para entrenamiento y otro para prueba) en la carpeta data/data_processed. Esto permite reutilizar el mismo conjunto de datos preprocesados para el modelado sin tener que repetir el proceso de transformación.