RESUMEN

Flujo de Trabajo

Este documento resume el flujo de trabajo del proyecto "Preparación de Datos con Python", enfocado en la obtención, limpieza y estructuración de información utilizando herramientas avanzadas de programación.

1

Justificación de las herramientas utilizadas

El proyecto se fundamenta en el uso exclusivo de NumPy y Pandas por las siguientes razones técnicas:

- **NumPy**: es el pilar de la computación científica en Python, permitiendo el manejo eficiente de arreglos multidimensionales y la ejecución de operaciones vectorizadas. Esto elimina la necesidad de bucles manuales, mejorando el rendimiento computacional y la legibilidad del código.
- **Pandas**: Es la biblioteca estándar para el análisis de datos estructurados. Ofrece estructuras como el DataFrame que facilitan la carga, limpieza, filtrado y análisis estadístico de datos provenientes de múltiples formatos como CSV y Excel.

2

Descripción del Dataset y fuentes integradas

El conjunto de datos final se construyó a partir de tres orígenes distintos:

a) Dataset generado con NumPy: Se crearon datos ficticios para 50 clientes, incluyendo IDs, edades (rango 18-75 años), montos de compras y número de transacciones. Adicionalmente, para el ejercicio, se crearon 4 filas duplicadas en base a la información de este.

b)Fuente externa de Excel: Se integró el archivo "Detalle de Datos de Clientes vf.xlsx", que aportó información complementaria como el Nombre, Comuna y Sector de los usuarios.

c) Fuente externa de tabla de la Web: Se extrajeron datos adicionales (como códigos de comuna, provincias y regiones) desde tablas web para enriquecer la localización geográfica de los clientes.

3

Técnicas aplicadas para la limpieza y transformación

El proceso de Data Wrangling incluyó las siguientes técnicas fundamentales:

- **Exploración**: Uso de métodos como head(), info() y describe() para verificar la carga y estructura inicial.
- **Limpieza de registros**: Identificación y eliminación de 4 registros duplicados.
- **Tratamiento de nulos**: Se identificaron valores faltantes y se reemplazaron con la etiqueta "Desconocido/a" o el valor 0 según correspondiera a columnas categóricas o numéricas.
- **Detección de outliers**: Se aplicaron métodos como el Rango Intercuartil (IQR) y Z-score, además de visualizaciones con Boxplots, determinando que no existían valores atípicos que distorsionaran el análisis en este conjunto específico.
- **Estructuración**: Se realizaron uniones de tablas mediante merge() y reestructuraciones con melt() y pivot_table() para generar reportes dinámicos.

4

Justificación de las herramientas utilizadas

a) **Formatos**: Se debió configurar correctamente el separador (,) en archivos CSV para garantizar la integridad de los datos durante la carga.

b) **Extracción Web**: El mayor desafío fue la lectura de datos desde la web; el método simple read_html() falló inicialmente, por lo que se decidió implementar una solución más robusta utilizando requests y StringIO para asegurar la captura de la información.

c) **Consolidación**: Al unificar las fuentes, se generaron columnas redundantes o innecesarias, las cuales fueron eliminadas para mantener un dataset optimizado y limpio.

5

Resultados Obtenidos y Estado Final

Al finalizar el flujo de trabajo, se obtuvo un DataFrame consolidado, limpio y enriquecido. Los datos ahora cuentan con una estructura jerárquica clara, tipos de datos correctamente asignados y una segmentación por sectores y rangos etarios lista para ser utilizada en modelos predictivos o dashboards. El dataset final fue exportado exitosamente en formatos CSV y Excel.

Bernardita Ortega F. https://github.com/bernimof