Transformaciones Básicas con Pandas - Día 2

Change Status

pending 40 min

Learning Objectives

1. Entender operaciones básicas de limpieza de datos
2. Aprender manejo de valores faltantes
3. Comprender normalización de datos
4. Conocer técnicas de validación básica

Practical exercise to apply the concepts learned.

Ejercicio: Limpiar y validar un dataset de ventas

Dataset con problemas:

```
import pandas as pd
import numpy as np

# Crear datos de ejemplo con problemas
ventas = pd.DataFrame({
    'producto': ['A', 'B', None, 'A', 'C'],
    'precio': [100, None, 150, 100, 200],
    'cantidad': [1, 2, None, 1, 3],
    'fecha': ['2024-01-01', None, '2024-01-03', '2024-01-01', 'invalid']
})

print("Datos originales:")
print(ventas)
print(f"Valores faltantes por columna:\n{ventas.isnull().sum()}")
```

Limpiar datos:

```
def limpiar_datos_ventas(df):
    df_limpio = df.copy()

    # 1. Eliminar duplicados
    df_limpio = df_limpio.drop_duplicates()

    # 2. Imputar valores faltantes
    df_limpio['precio'] = df_limpio['precio'].fillna(df_limpio['precio'].median())
    df_limpio['cantidad'] = df_limpio['cantidad'].fillna(1)  # Asumir cantidad mínima

    # 3. Eliminar filas con producto faltante
    df_limpio = df_limpio.dropna(subset=['producto'])

    # 4. Corregir fechas inválidas
    df_limpio['fecha'] = pd.to_datetime(df_limpio['fecha'], errors='coerce')
    df_limpio = df_limpio.dropna(subset=['fecha'])

    # 5. Calcular total
    df_limpio['total'] = df_limpio['precio'] * df_limpio['cantidad']

    return df_limpio

ventas_limpias = limpiar_datos_ventas(ventas)
print("\nDatos limpios:")
print(ventas_limpias)
print(f"\nRegistros finales: {len(ventas_limpias)}")
```

Validar datos limpios:

```
def validar_ventas_limpias(df):
    validaciones = {
        'sin_faltantes': df.isnull().sum().sum() == 0,
        'precios_positivos': (df['precio'] > 0).all(),
        'cantidades_positivas': (df['cantidad'] > 0).all(),
        'fechas_validas': pd.api.types.is_datetime64_any_dtype(df['fecha']),
        'total_correcto': np.allclose(df['total'], df['precio'] * df['cantidad'])
    }
```

```
    print("Validaciones:")
    for check, passed in validaciones.items():
        status = "✅" if passed else "❌"
        print(f"  {status} {check}")

    return all(validaciones.values())

es_valido = validar_ventas_limpias(ventas_limpias)
print(f"\nDataset válido: {es_valido}")
```

Verificación: ¿Cuándo deberías eliminar datos faltantes vs imputarlos? ¿Qué tipos de validaciones son más importantes para diferentes tipos de datos?

Requerimientos:

Pandas instalado
Conocimiento básico de manipulación de datos
Comprensión de calidad de datos