Contenido

[ETL 1](#_Toc207122662)

[Pandas 1](#_Toc207122663)

[Matplotlib/Seaborn 2](#_Toc207122664)

[Flujo de trabajo y preprocesamiento 2](#_Toc207122665)

[Lógica y estadística básica 3](#_Toc207122666)

[Buenas prácticas y razonamiento 3](#_Toc207122667)

# ETL

**Explica qué hace el paso de "Transform" en un proceso ETL.**

✅ **Respuesta:**  
En un proceso **ETL (Extract, Transform, Load)**, el paso **Transform (Transformar)** consiste en **convertir, limpiar y dar formato a los datos extraídos** para que sean consistentes y útiles antes de cargarlos en el destino final (por ejemplo, un data warehouse).

Algunas tareas comunes en esta etapa incluyen:

* Limpiar datos (eliminar duplicados, manejar valores nulos, corregir errores).
* Cambiar formatos (por ejemplo, fechas o unidades de medida).
* Integrar información de múltiples fuentes.
* Aplicar reglas de negocio (cálculos, clasificaciones, agregaciones).
* Normalizar o desnormalizar datos según sea necesario.

👉 En pocas palabras, es el paso donde los datos se preparan y se adaptan para poder ser utilizados de manera eficiente en el análisis o en las aplicaciones de negocio.

# Pandas

¿Qué es una Serie y qué es un DataFrame?

¿Cómo seleccionas filas por condición?

¿Cómo manejas valores nulos?

¿Qué hace groupby() y cuándo se usa?

¿Cómo haces un merge entre dos DataFrames?

¿Cuál es la diferencia entre df.loc[] y df.iloc[]?

¿Cómo manejarías duplicados en un DataFrame y cómo decides si eliminarlos?

¿Qué hace el método .groupby() y en qué situaciones es útil?

¿Cómo puedes combinar o unir dos DataFrames? Menciona diferencias entre merge(), join() y concat().

¿Cómo filtrarías las filas de un DataFrame con múltiples condiciones lógicas?

¿Qué hace el método .apply() y cuándo lo usarías en lugar de un bucle for?

¿Cómo convertirías una columna con fechas en un índice temporal y qué beneficios tiene hacerlo?

# Matplotlib/Seaborn

* ¿Diferencias entre Matplotlib y Seaborn?
* ¿Qué gráfico usarías para:  
  a) distribución,  
  b) correlación,  
  c) comparación de categorías?

 ¿Cómo cambiarías el tamaño de una figura en Matplotlib?

 ¿Qué función se usa para agregar etiquetas a los ejes y un título a la gráfica?

 ¿Cómo graficarías múltiples líneas en un solo gráfico y cómo distinguirlas visualmente?

 ¿Qué ventajas ofrece usar subplot() y cuándo lo usarías?

 ¿Cómo cambiarías el estilo de línea y color en una gráfica de líneas?

 ¿Cuál es la diferencia entre plt.plot(), plt.bar(), y plt.hist()?

 ¿Qué técnicas usarías para hacer una visualización más clara si tienes muchos datos en un mismo gráfico?

# Flujo de trabajo y preprocesamiento

* ¿Cómo cargas un CSV y ves las primeras filas?
* ¿Cómo identificas y tratas valores faltantes?
* ¿Qué es codificación one-hot? ¿Cuándo la usas?
* ¿Qué diferencia hay entre normalización y estandarización?
* ¿Cómo filtras, ordenas y renombrar columnas con pandas?

 ¿Qué pasos considerarías antes de entrenar un modelo con tus datos?

 ¿Cómo manejarías valores faltantes? ¿Qué criterios usarías para imputarlos o eliminarlos?

 ¿Por qué es importante normalizar o escalar los datos antes de usar ciertos modelos?

 ¿Cuál es la diferencia entre codificación ordinal y codificación one-hot? ¿Cuándo usar cada una?

 ¿Cómo detectarías y manejarías outliers?

 ¿Qué es el leakage de datos y cómo lo evitarías en el preprocesamiento?

 ¿Cuál es la diferencia entre transformar y estandarizar datos?

# Lógica y estadística básica

* ¿Cómo calculas media, mediana y std en pandas?
* ¿Qué es la correlación y cómo la obtienes?
* ¿Qué es un outlier y cómo lo detectarías?
* ¿Qué diferencia hay entre apply() y map()?
* ¿Qué hace value\_counts() y para qué sirve?

# Buenas prácticas y razonamiento

* ¿Qué es código reproducible?
* ¿Por qué es mejor vectorizar que usar bucles?
* ¿Qué significa documentar bien tu código?
* ¿Cómo dividirías un proyecto de análisis en etapas claras?
* ¿Qué es un pipeline y cómo ayuda en ciencia de datos?

 ¿Por qué es importante documentar tu código y cómo lo haces normalmente?

 ¿Cuál es el propósito de usar control de versiones como Git en proyectos de ciencia de datos?

 ¿Qué es un pipeline de datos y por qué es útil implementarlo?

 ¿Qué consideraciones tienes al trabajar con datos sensibles o privados?

 ¿Por qué es importante la reproducibilidad en ciencia de datos y cómo la garantizas?

 ¿Qué errores comunes has encontrado al trabajar con notebooks de Jupyter y cómo los evitas?

 ¿Por qué deberías evitar modificar un DataFrame directamente sin hacer una copia explícita?