


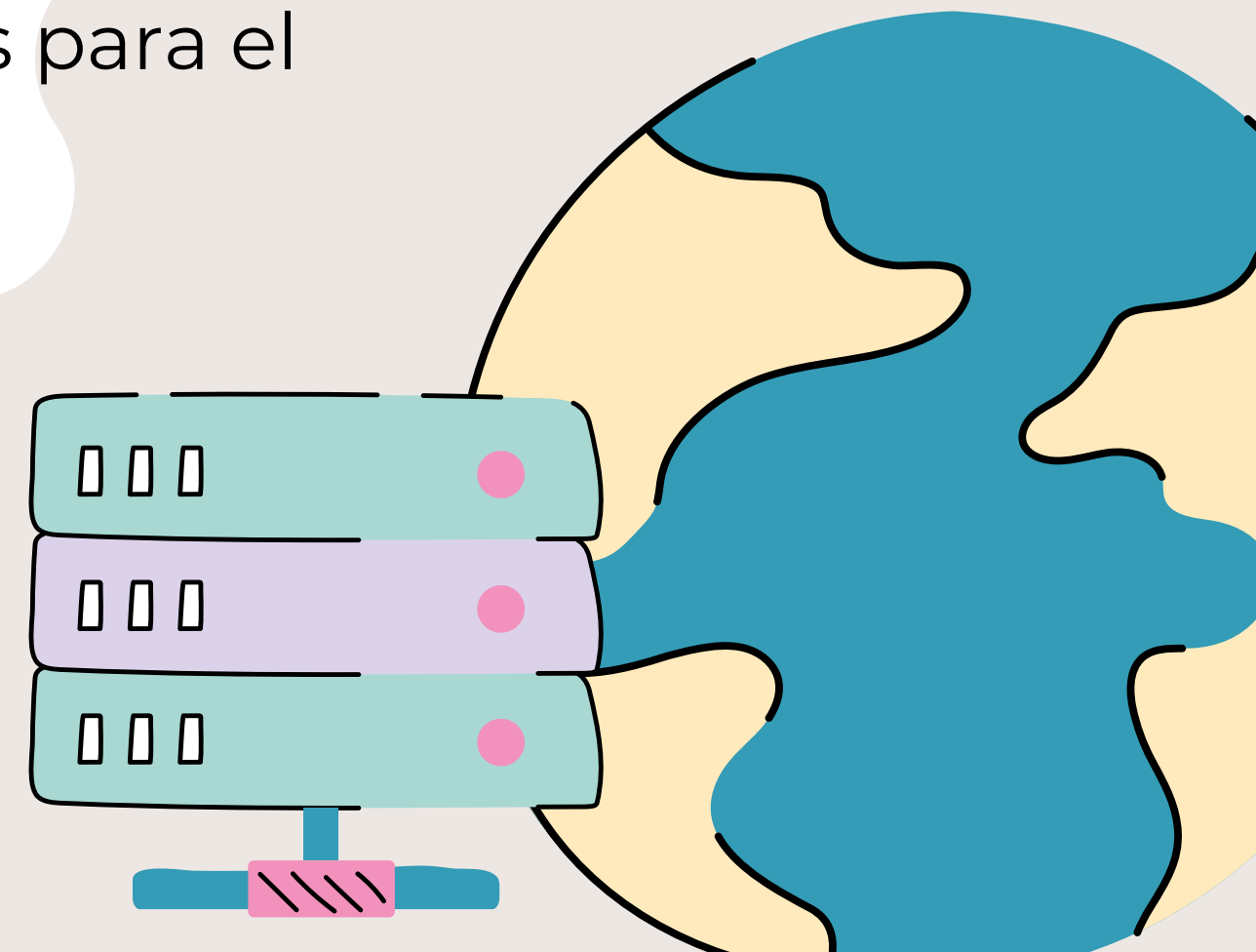


Limpieza y preprocesamiento de datos en Python

Domina las técnicas y prepara los datos para el análisis y modelos predictivos



Por Juan Duran

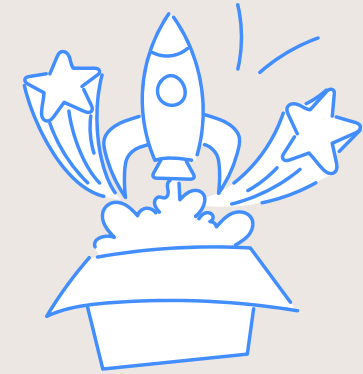


Introducción

La **limpieza** y el **preprocesamiento** de datos son pasos fundamentales en cualquier **análisis** de **datos** o proyecto de **machine learning**. Los datos crudos, tal como los obtenemos de diversas fuentes, suelen estar desordenados, incompletos o incluso erróneos. Antes de poder hacer un análisis o aplicar modelos predictivos, es necesario aplicar técnicas de **limpieza** y **transformación** para que los datos sean útiles y efectivos.

En esta presentación, exploraremos cómo usar **Python** para:

- **Manejar** datos faltantes.
- **Detectar** y tratar outliers.
- **Utilizar** librerías clave como pandas, sklearn.preprocessing y missingno para facilitar este proceso.



Puntos clave



Datos faltantes

Los **datos faltantes** son comunes y pueden **afectar negativamente** los resultados de los modelos.

Podemos **reemplazarlos** con valores como la **media**, la **mediana**, o usar técnicas más complejas como la **imputación** múltiple.



Detección outliers

Los **outliers** son **valores extremos** que pueden **distorsionar** los **resultados**. Identificarlos y tratarlos es esencial para mejorar la precisión del modelo. Técnicas como el método de **z-score** y la visualización con **boxplots** son útiles para su detección.



Transformación datos

La **normalización** y **estandarización** son técnicas clave para que los datos se ajusten a las suposiciones de los modelos. Usar **sklearn.preprocessing** para escalar datos es fundamental para mejorar el rendimiento de modelos como SVM o redes neuronales.

Limpieza de datos faltantes

El manejo de **valores nulos** o faltantes es uno de los aspectos más comunes en el **preprocesamiento** de **datos**. Existen diversas formas de tratar estos valores según el contexto y el tipo de análisis que se quiera realizar.

- **Eliminación** de filas/columnas: Si el porcentaje de datos faltantes es bajo, podemos eliminar las filas o columnas que contienen estos valores.
- **Imputación**: Podemos reemplazar los valores faltantes con estadísticas como la media, mediana o moda. También podemos utilizar técnicas más avanzadas como la imputación por el método KNN o la regresión.
- **Interpolación**: En datos temporales, la interpolación es útil para estimar valores faltantes en función de los valores circundantes.

Eliminación

Visualización

Imputación

Interpolación



Manejo de Outliers

Los **outliers** son **valores** que se alejan significativamente del resto de los datos y pueden **distorsionar** las **estadísticas** y **modelos**.

Detectarlos y manejarlos correctamente es esencial para obtener modelos más precisos.

- Método **Z-score**: Calcula el número de desviaciones estándar que un dato se aleja de la media. Si el Z-score es mayor que un umbral determinado (generalmente 3), el valor es considerado un outlier.
- **Boxplot**: Utiliza la representación gráfica de los cuartiles para identificar puntos que se encuentran fuera del rango intercuartil, que se consideran outliers.
- **Transformaciones**: En lugar de eliminar los outliers, podemos transformarlos, como aplicar una logaritmización, para mitigar su impacto.

Z-score



Boxplot

Transformación

Detección



Transformación de datos

La **transformación** de los datos es crucial para que nuestros modelos funcionen correctamente. Es fundamental que los datos estén **normalizados** o **estandarizados** para que los algoritmos como K-means, SVM o redes neuronales puedan procesarlos adecuadamente.

- **Normalización:** Escala los datos para que estén dentro de un rango específico, típicamente $[0, 1]$, utilizando la fórmula $(x - \min) / (\max - \min)$.
- **Estandarización:** Transforma los datos para que tengan una media de 0 y una desviación estándar de 1, lo que es útil para los modelos que asumen una distribución normal de los datos.
- **Escalado** robusto: Utiliza el método de escalado que es más resistente a los outliers, ideal cuando se espera que haya valores extremos.

Normalización



Estandarización

Escalado

MinMaxScaler





Conclusiones



Limpieza

Sin un **preprocesamiento** adecuado, incluso los modelos más sofisticados pueden fallar o producir resultados poco fiables.

Pandas y sklearn

Estas **herramientas** facilitan el manejo de datos y la implementación de técnicas de preprocesamiento.

Datos faltantes

Imputación o **eliminación**, según el contexto y el impacto en el análisis.

Transformaciones

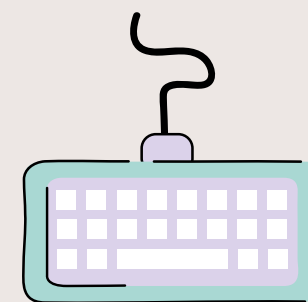
La **normalización** y **estandarización** mejoran el rendimiento de muchos algoritmos.

Outliers

Detectarlos y tratarlos es fundamental para mejorar la calidad del modelo.

Visualización

Usar **missingno** y otras herramientas visuales para entender los **patrones** en los datos faltantes y **outliers** es muy útil para tomar decisiones informadas.



Gracias



Por Juan Duran

“Coding, Gaming and Leveling Up”