INTELIGENCIA ARTIFICIAL Y MACHINE LEARNING

DATOS VACIOS

ANDRES ANTONIO CERVANTES CONSUEGRA
LUIS FERNANDO PICON CARRILLO
CAMILA ANDREA ROSALES MERCADO

JORGE RUDAS

TECNOLOGÍA EN GESTIÓN DE SISTEMAS INFORMÁTICOS

INSTITUCIÓN DE EDUCACIÓN SUPERIOR HUMBERTO VELASQUE GARCIA (INFOTEP)

CIENAGA - MAGDALENA

2024

introducción: verificamos y analizamos los datos null(vacios) para definir la calidad de información si no hay datos suministrados para hacer una estrategia para definir si dichos datos pueden ser suministrados o eliminados por un algoritmo.

Objetivos:

identificar si el dato es (null) vacío se elimina

Analizar si la cantidad de datos suministrada es deficiente

detectar y procesar los datos de manera efectiva para garantizar que el conjunto de datos, sea utilizado de manera unifomer.

Defina una métrica de calidad de datos vacíos

Es la medida utilizada para evaluar el impacto de los valores faltantes en un conjunto de datos. Ayuda a determinar la calidad de los datos en un análisis y en tomas de decisiones lo podemos encontrar en frecuencias de valores faltantes por columnas, distribucion de valores faltantes, impacto en la completitud del dataset y en la relacion con otras variables.

Utilizar visualizaciones para identificar columnas con valores faltante, incluyen valores NaN (no un número) o NULOS.

Determine el impacto de los valores que faltan en su análisis o modelo. Tener en cuenta el porcentaje de valores que faltan en cada columna y su importancia para el conjunto de datos global.

Para las funciones numéricas, puede imputar los valores que faltan usando técnicas como la media, la mediana o el método de imputación de modo (fillna() en pandas). Para las características categóricas, puede imputar con la categoría más frecuente.

Defina un algoritmo para evaluar la calidad de los datos

1. para la revision de los valores vacíos lo que podemos hacer agregar la funcion isnull para decir que hacen falta datos en la columna

   # Cuenta valores faltantes por columna

   print(df.isnull())

2. Visualizar los datos faltantes

La forma más básica de visualizar los datos faltantes es usar la función `isnull()` para identificar los valores nulos y luego aplicar `sum()` para contar cuántos valores faltan por columna.

 # Ver la cantidad de valores faltantes por columna

missing_data = df.isnull().sum()

3. Evaluar la completitud del dataset

Proporción de valores faltantes por columna

Esto te dará un porcentaje de datos faltantes en cada columna. Si la proporción de datos faltantes es alta en alguna columna, es posible que esa columna no sea útil para el análisis o que deba tomar medidas, es decir que debamos eliminar esa columna

missing_proportion = d_medellin.isnull().mean()

Completitud por fila (Evaluar filas con datos faltantes)

Se puede ver cuántas filas tienen datos faltantes y calcular la proporción de filas que están completas

 # Calcular el porcentaje de filas completas

complete_rows = df.dropna()

complete_percentage = len(complete_rows) / len(df) * 100

 # Mostrar el porcentaje de filas completas

print(f"Porcentaje de filas completas: {complete_percentage}%")

 # Filas con valores faltantes

incomplete_rows = df[df.isnull().any(axis=1)]

4. Detectar patrones en los datos vacíos

Identificar si los datos faltantes son aleatorios o tienen patrones

 # Visualizar el patrón de los valores faltantes con un gráfico tipo matrix

msno.matrix(df)

 # Mostrar la visualización

plt.show()

conclusión: según los datos nulos (null) en SQL son importantes en el proceso de análisis de datos, debido a saber las métricas requeridas para saber si el archivo es de alta prioridad o visualizar el problema que está afectando la base y que decisión tomar y como solucionalo.