Healthcare Data Cleaning Report

Name- Vidisha Goel

Branch- CSE (AI)

Section- D

University Roll No. - 202401100300278

Introduction

Data cleaning is a crucial step in the data preprocessing pipeline, ensuring data quality, consistency, and accuracy. This report documents the process of cleaning a healthcare dataset to remove missing values, handle duplicates, standardize text data, and detect outliers. The cleaned dataset enhances the reliability of analysis and machine learning models.

Methodology

The following steps were undertaken to clean the healthcare dataset:

- The provided dataset was uploaded using Google Colab's **files.upload()** method.
- **Pandas** was also used to load and process the dataset.
- Displayed the first five rows of the dataset.
- Checked dataset information using **df.info()**
- Identified missing values using **df.isnull().sum()**
- Numerical columns were filled with their respective median values.
- Categorical columns were filled with their most frequent values (mode).
- Duplicate rows were removed using **df.drop_duplicates(inplace=True)**
- Columns that were having 'date' in their name were converted to date time format using **pd.to_datetime()**
- Categorical text data was converted to lowercase and stripped of leading/trailing spaces.
- The IQR (Inter-Quartile Range) method was used to detect potential outliers in numerical columns.
- Outliers were identified using **Q1 - 1.5 * IQR and Q3 + 1.5 * IQR.**
- The final cleaned dataset was saved as **healthcare_data_cleaned.csv**.

Code

```
from google.colab import files

import pandas as pd


# Upload file

uploaded = files.upload()


# Load the dataset (assuming only one file is uploaded)

file_name = list(uploaded.keys())[0]  # Get the uploaded file name

df = pd.read_csv(file_name)


# Display basic information about the dataset

print("Initial Dataset Info:")

df.info()

print("\nFirst 5 Rows:")

display(df.head())


# Handling missing values

print("\nChecking missing values:")

print(df.isnull().sum())


# Fill missing values
```

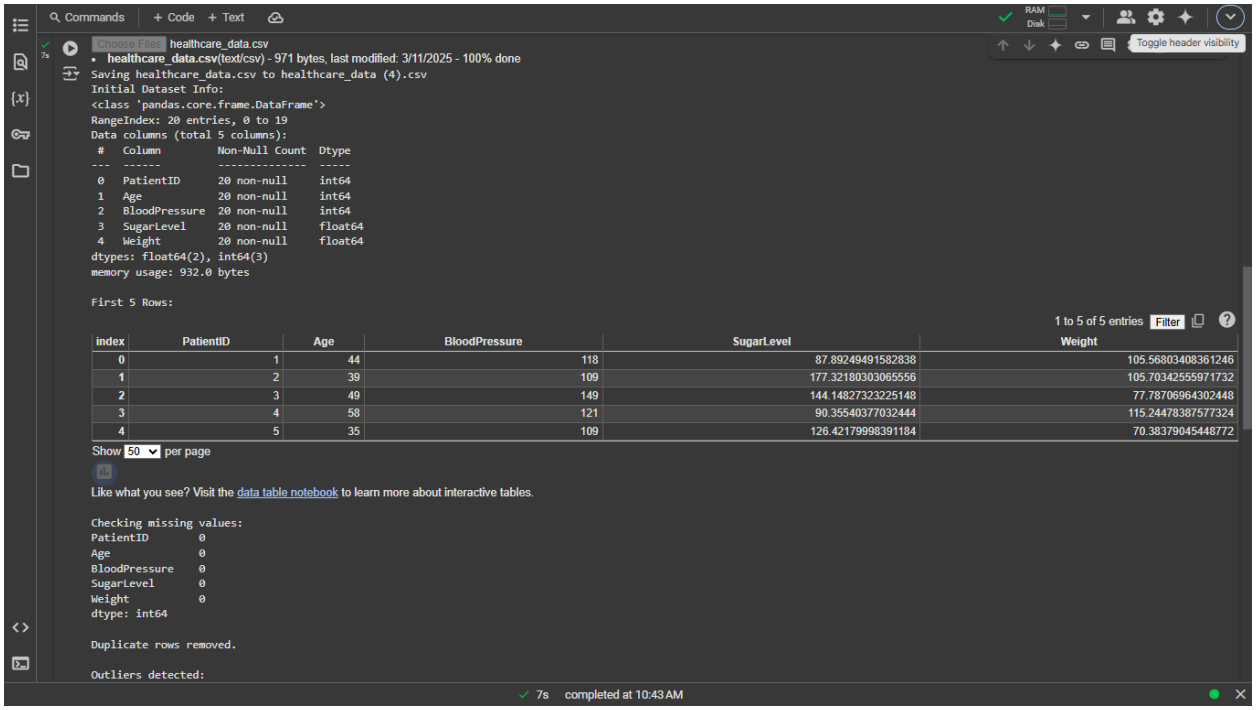
```
numeric_cols = df.select_dtypes(include=['number']).columns

df[numeric_cols] = df[numeric_cols].fillna(df[numeric_cols].median())

categorical_cols = df.select_dtypes(include=['object']).columns

if not categorical_cols.empty:

    df[categorical_cols] = df[categorical_cols].fillna(df[categorical_cols].mode().iloc[0])


# Remove duplicate rows

df.drop_duplicates(inplace=True)

print("\nDuplicate rows removed.")


# Convert date columns

to datetime format

for col in df.columns:

    if 'date' in col.lower():

        df[col] = pd.to_datetime(df[col], errors='coerce')


# Standardizing text data

for col in categorical_cols:

    df[col] = df[col].str.lower().str.strip()


# Detecting outliers

Q1 = df[numeric_cols].quantile(0.25)

Q3 = df[numeric_cols].quantile(0.75)

IQR = Q3 - Q1
```

```
outliers = ((df[numeric_cols] < (Q1 - 1.5 * IQR)) | (df[numeric_cols] > (Q3 + 1.5 * IQR))).sum()

print("\nOutliers detected:")

print(outliers)


# Save the cleaned dataset

cleaned_file_path = "healthcare_data_cleaned.csv"

df.to_csv(cleaned_file_path, index=False)

print(f"\nData cleaning complete. Cleaned file saved as '{cleaned_file_path}'.")
```

Output/Result

The cleaned dataset is saved as `healthcare_data_cleaned.csv`. Key results include:

- Missing values were successfully handled.
- Duplicate records were removed.
- Date columns were standardized.
- Text data was cleaned and standardized.
- Outliers were identified for further review.



```
healthcare_data.csv
healthcare_data.csv(text/csv) - 971 bytes, last modified: 3/11/2025 - 100% done
Saving healthcare_data.csv to healthcare_data (4).csv
Initial Dataset Info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 5 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   PatientID      20 non-null     int64
 1   Age            20 non-null     int64
 2   BloodPressure  20 non-null     int64
 3   SugarLevel     20 non-null     float64
 4   Weight         20 non-null     float64
dtypes: float64(2), int64(3)
memory usage: 932.0 bytes

First 5 Rows:
```

1 to 5 of 5 entries

index	PatientID	Age	BloodPressure	SugarLevel	Weight
0	1	44	118	87.89249491582838	105.56803408361246
1	2	39	109	177.32180303065556	105.70342555971732
2	3	49	149	144.14827323225148	77.78706964302448
3	4	58	121	90.35540377032444	115.24478387577324
4	5	35	109	126.42179998391184	70.38379045448772

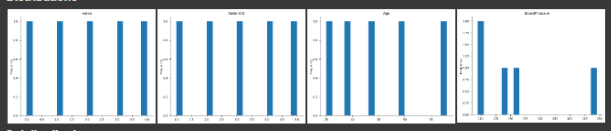
```
Like what you see? Visit the data table notebook to learn more about interactive tables.

Checking missing values:
PatientID        0
Age              0
BloodPressure    0
SugarLevel       0
Weight           0
dtype: int64

Duplicate rows removed.

Outliers detected:
```

7s completed at 10:43 AM

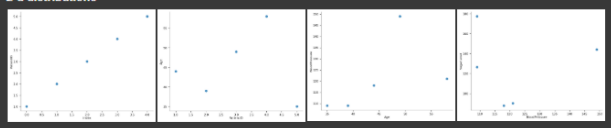
Commands + Code + Text

```
Outliers detected:
PatientID        0
Age              0
BloodPressure    0
SugarLevel       0
Weight           0
dtype: int64

Data cleaning complete. Cleaned file saved as 'healthcare_data_cleaned.csv'.
```

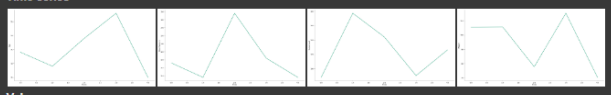
Distributions



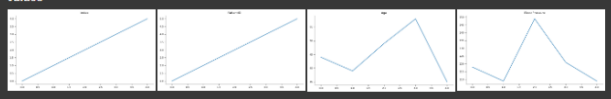
2-d distributions



Time series



Values



7s completed at 10:43 AM

References/Credits

- **Pandas Documentation**: https://pandas.pydata.org/docs/
- **Google Colab Documentation**: https://colab.research.google.com/notebooks/
- **Interquartile Range (IQR) Outlier Detection**: https://en.wikipedia.org/wiki/Interquartile_range
- **Chatgpt**