Healthcare Data Cleaning Report

NAME-PRIYESH KUMAR

ROLL NO-202401100300186

Introduction

This report summarizes the data cleaning steps performed on a healthcare dataset using Python's `pandas` library.

Initial Data Overview

- **Total Entries:** 20
- **Columns:** 5 (`PatientID`, `Age`, `BloodPressure`, `SugarLevel`, `Weight`)
- **Data Types:** Integer (3), Float (2)
- **Memory Usage:** 932 bytes

Sample Rows (Before Cleaning)

PatientID	Age	BloodPressure	SugarLevel	Weight
1	44	118	87.892495	105.568034
2	39	109	177.321803	105.703426
3	49	149	144.148273	77.787070
4	58	121	90.355404	115.244784
5	35	109	126.421800	70.383790

Cleaning Steps

```
import pandas as pd

# Load the dataset
file_path = '/content/healthcare_data.csv'
df = pd.read_csv(file_path)

# Display basic information about the dataset
print("Initial Data Overview:")
print(df.info())
print("\nFirst 5 rows:")
print(df.head())

# Data Cleaning Steps
# 1. Handling Missing Values
df.ffill(inplace=True)  # Forward fill for continuous
data

# 2. Removing Duplicates
df.drop_duplicates(inplace=True)

# 3. Standardizing Column Names
df.columns =
df.columns.str.strip().str.lower().str.replace(' ',
'_')

# 4. Converting Data Types (example for date columns)
if 'date_of_birth' in df.columns:
    df['date_of_birth'] =
pd.to_datetime(df['date_of_birth'], errors='coerce')

# 5. Removing Outliers (example for age column)
if 'age' in df.columns:
    df = df[(df['age'] >= 0) & (df['age'] <= 120)]

# 6. Encoding Categorical Variables (example for
gender)
if 'gender' in df.columns:
```

```
    df['gender'] = df['gender'].map({'Male': 1,
'Female': 0}).fillna(-1)

# Display cleaned data overview
print("\nCleaned Data Overview:")
print(df.info())
print("\nFirst 5 cleaned rows:")
print(df.head())

# Save the cleaned dataset
cleaned_file_path = '/content/healthcare_data.csv'
df.to_csv(cleaned_file_path, index=False)
print(f"Cleaned data saved to {cleaned_file_path}")
```

Cleaned Data Overview

- **Total Entries:** 20
- **Columns:** 5 (`patientid`, `age`, `bloodpressure`, `sugarlevel`, `weight`)
- **Data Types:** Integer (3), Float (2)
- **Memory Usage:** 932 bytes

Sample Rows (After Cleaning)

patientid	age	bloodpressure	sugarlevel	weight
1	44	118	87.892495	105.568034
2	39	109	177.321803	105.703426
3	49	149	144.148273	77.787070
4	58	121	90.355404	115.244784
5	35	109	126.421800	70.383790

Notes

- A **FutureWarning** was raised for `fillna(method='ffill')`. For compatibility with future versions of `pandas`, `.ffill()` was used instead.

Conclusion

The cleaned dataset is saved as I've added a pie chart visualization for the age distribution to the repor`/content/healthcare_data.csv`. The dataset is now standardized, free of duplicates, and has improved data quality for analysis.