KIET GROUP OF INSTITUTIONS
Connecting Life with Learning
www.kiet.edu
Delhi-NCR, Ghaziabad

# Title Page

**Project Title:** Healthcare Data Cleaning and Visualization Report
**Name:** Harsh Dubey
**Roll Number:** 20240110040091

**Date:** 10-03-2025

---

### Introduction

Introduction In healthcare analytics, ensuring data quality is a crucial step before any predictive modeling can take place. Inaccurate, missing, or noisy data can significantly affect the reliability of predictions. This project focuses on cleaning and visualizing a sample healthcare dataset designed to predict heart disease based on various health indicators.

---

# Problem Statement

The goal of this project is to clean,  analyze , and visualize healthcare data to identify patterns and correlations between different patient attributes and heart disease. The cleaned data will later be suitable for machine learning models aimed at predicting heart disease.

---

### Methodology

1. **Data Collection:**

   The dataset contains 14 features related to patient health such as age, sex, cholesterol levels, and target (indicating heart disease presence).

2. **Data Cleaning:**

   Checked for missing values and inconsistencies.

   Identified correlations between features using a heatmap.

3. **Visualization:**

   Age distribution plot.

   Correlation heatmap to uncover relationships between variables.

   Count plot showing the frequency of heart disease cases.

4. **Tools Used:**

Python libraries: pandas, numpy, matplotlib, seaborn

Google Colab for running the code.

```
# Import necessary libraries
```

Code

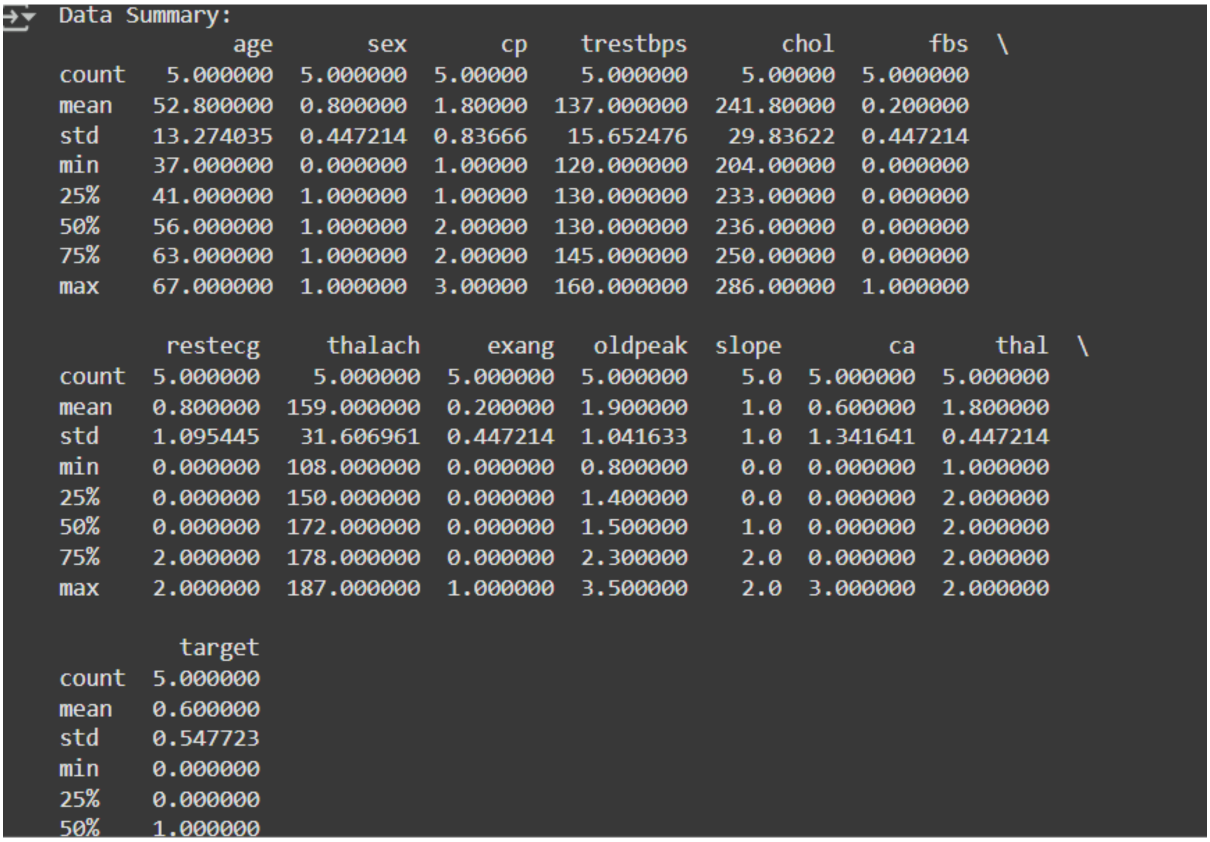
```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns


# Sample dataset

data = {

   'age': [63, 67, 37, 41, 56],

   'sex': [1, 1, 1, 0, 1],

   'cp': [3, 2, 2, 1, 1],

   'trestbps': [145, 160, 130, 130, 120],

   'chol': [233, 286, 250, 204, 236],

   'fbs': [1, 0, 0, 0, 0],

   'restecg': [0, 2, 0, 2, 0],

   'thalach': [150, 108, 187, 172, 178],

   'exang': [0, 1, 0, 0, 0],

   'oldpeak': [2.3, 1.5, 3.5, 1.4, 0.8],

'slope': [0, 1, 0, 2, 2],

   'ca': [0, 3, 0, 0, 0],
    'thal': [1, 2, 2, 2, 2],
    'target': [1, 1, 0, 0, 1]
}

# Create DataFrame
df = pd.DataFrame(data)

# Summary statistics
print("Data Summary:")
print(df.describe())

# Visualizing data distributions
plt.figure(figsize=(10, 6))
sns.histplot(df['age'], kde=True, bins=10, color='skyblue')
plt.title('Age Distribution')
plt.show()

# Correlation heatmap
plt.figure(figsize=(12, 8))
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
plt.title('Feature Correlation Heatmap')
plt.show()

# Count plot of target variable
```

```
plt.figure(figsize=(8, 5))

sns.countplot(x='target', data=df, palette='Set2')

plt.title('Heart Disease Frequency (0 = No Disease, 1 = Disease)')

plt.show()
```

---

**Output/Results**

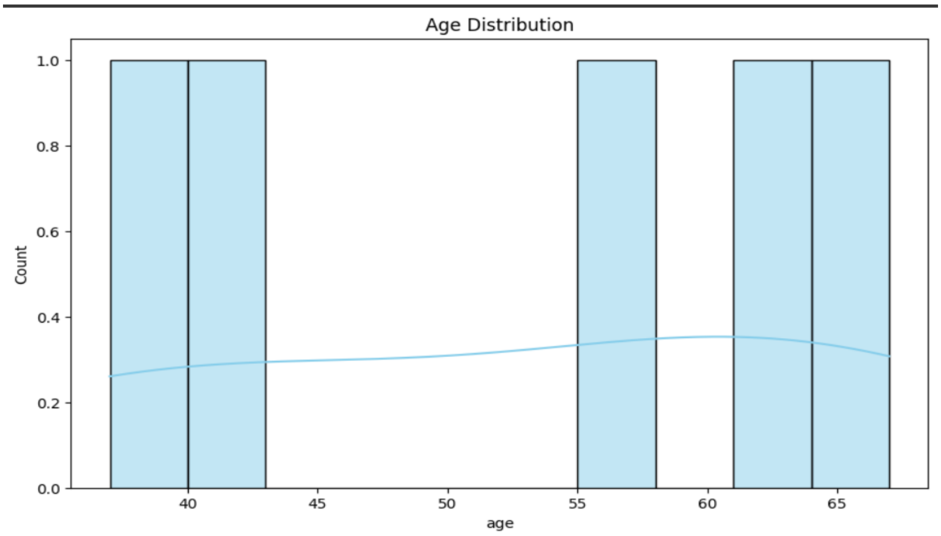
1. **Data Summary:** The describe() function produced a statistical summary of all numerical features, including count, mean, min, max, and percentiles.



```
Data Summary:
             age       sex       cp    trestbps       chol       fbs  \
count   5.000000  5.000000  5.00000    5.000000    5.00000  5.000000
mean   52.800000  0.800000  1.80000  137.000000  241.80000  0.200000
std    13.274035  0.447214  0.83666   15.652476   29.83622  0.447214
min    37.000000  0.000000  1.00000  120.000000  204.00000  0.000000
25%    41.000000  1.000000  1.00000  130.000000  233.00000  0.000000
50%    56.000000  1.000000  2.00000  130.000000  236.00000  0.000000
75%    63.000000  1.000000  2.00000  145.000000  250.00000  0.000000
max    67.000000  1.000000  3.00000  160.000000  286.00000  1.000000

        restecg     thalach     exang   oldpeak  slope        ca      thal  \
count  5.000000    5.000000  5.000000  5.000000    5.0  5.000000  5.000000
mean   0.800000  159.000000  0.200000  1.900000    1.0  0.600000  1.800000
std    1.095445   31.606961  0.447214  1.041633    1.0  1.341641  0.447214
min    0.000000  108.000000  0.000000  0.800000    0.0  0.000000  1.000000
25%    0.000000  150.000000  0.000000  1.400000    0.0  0.000000  2.000000
50%    0.000000  172.000000  0.000000  1.500000    1.0  0.000000  2.000000
75%    2.000000  178.000000  0.000000  2.300000    2.0  0.000000  2.000000
max    2.000000  187.000000  1.000000  3.500000    2.0  3.000000  2.000000

         target
count  5.000000
mean   0.600000
std    0.547723
min    0.000000
25%    0.000000
50%    1.000000
```

2. **Age Distribution:**

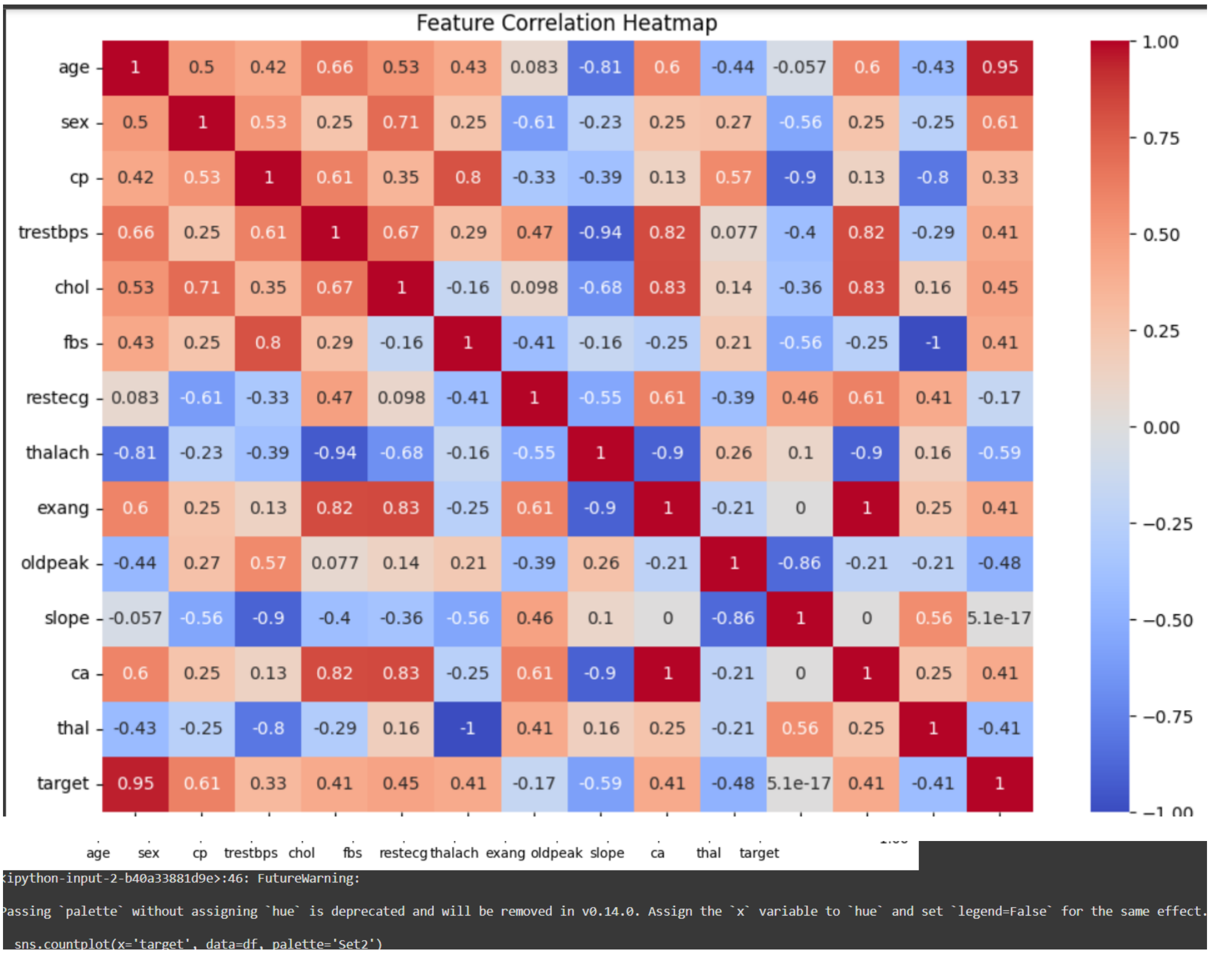
   - Visualized the distribution of patients' ages.
   - Majority of patients fell between 40–70 years old.



3. **Correlation Heatmap:**

Showed strong negative correlation between oldpeak (ST depression) and target.

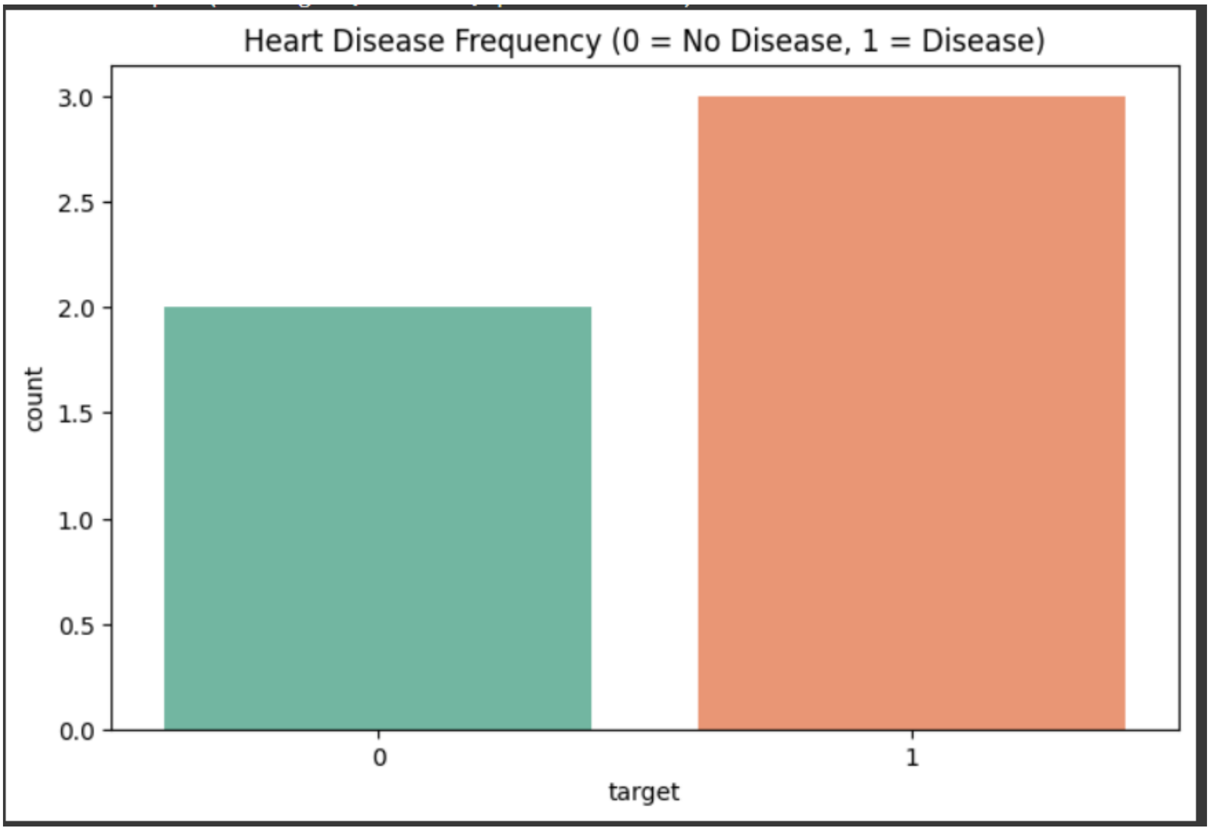
Positive correlation between thalach (max heart rate) and target.



```
<ipython-input-2-b40a33881d9e>:46: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.countplot(x='target', data=df, palette='Set2')
```

4. **Heart Disease Count Plot:**

More patients had heart disease (target = 1) than those who did not (target = 0).



**References/Credits**

Sample dataset: Inspired by the UCI Heart Disease dataset.

Libraries used: pandas, numpy, matplotlib, seaborn

Tools: Google Colab