Health Monitoring Analysis Project

Name: Mudumala Kalpana

USN: 1BO23MC025

Department: MCA(VTU)

Semester: 3rd Sem

➢ Objective :

The objective of a **Health Monitoring Analysis Project** is typically to collect, analyze, and interpret health-related data to gain insights into an individual's or population's overall health status. Here are some key goals that such a project might aim to achieve:

1. Early Detection of Health Issues

- Use data such as heart rate, blood pressure, sleep patterns, or physical activity to identify early signs of health issues or medical conditions like cardiovascular disease, diabetes, or hypertension.

2. Tracking and Improving Health Metrics

- Provide real-time tracking of vital health parameters, such as blood glucose levels, body temperature, or oxygen saturation, and offer recommendations for improvement based on the data.

3. Predictive Health Insights

- Develop predictive models to anticipate future health outcomes, helping to prevent diseases before they occur or worsen. For example, using historical health data to predict potential risks for chronic illnesses.

➢ Problem Statement :

The increasing prevalence of chronic diseases, coupled with the growing availability of health-related data from wearable devices and mobile health apps, has created an opportunity for more proactive and personalized healthcare. However, individuals and healthcare professionals are often overwhelmed by the vast amount of data generated daily, leading to missed early signs of health deterioration, inefficient healthcare interventions, and delayed medical response. Additionally, many health systems lack real-time integration and actionable insights, preventing timely decision-making and hindering preventive measures.

This project seeks to address the following key challenges:

1. **Early Detection of Health Issues:** Identifying potential health risks or abnormalities in real-time before they become critical, based on continuous health data from wearables, sensors, and user inputs.
2. **Data Overload and Complexity:** Developing a framework to manage and interpret large volumes of health data effectively, providing users and healthcare professionals with actionable insights and not just raw information.

➢ Solution :

The solution is to design and develop an **intelligent health monitoring system** that combines data collection, real-time monitoring, predictive analytics, and personalized recommendations to empower individuals and healthcare professionals. The system will use data from wearables, health sensors, mobile applications, and medical records to provide continuous health insights, early detection of health issues, and tailored wellness plans.

1. Data Collection and Integration

2. Data Processing and Real-Time Monitoring

3. Predictive Analytics and Early Detection

4. Personalized Health Insights and Recommendations

➢ Implementation :

```
import pandas as pd
health_data = pd.read_csv('healthmonitoring.csv')
print(health_data.head())
health_data.isnull().sum()
median_body_temp = health_data['BodyTemperature'].median()
median_oxygen_sat = health_data['OxygenSaturation'].median()

# fill missing values
health_data['BodyTemperature'].fillna(median_body_temp, inplace=True)
health_data['OxygenSaturation'].fillna(median_oxygen_sat, inplace=True)
import matplotlib.pyplot as plt
import seaborn as sns
sns.set(style="whitegrid")

# summary statistics
summary_stats = health_data.describe()

# plotting distributions of numerical features
fig, axes = plt.subplots(3, 2, figsize=(14, 18))
sns.histplot(health_data['Age'], bins=20, kde=True, ax=axes[0, 0])
axes[0, 0].set_title('Age Distribution')

sns.histplot(health_data['HeartRate'], bins=20, kde=True, ax=axes[0, 1])
axes[0, 1].set_title('Heart Rate Distribution')

sns.histplot(health_data['RespiratoryRate'], bins=20, kde=True, ax=axes[1, 0])
axes[1, 0].set_title('Respiratory Rate Distribution')

sns.histplot(health_data['BodyTemperature'], bins=20, kde=True, ax=axes[1, 1])
axes[1, 1].set_title('Body Temperature Distribution')

sns.histplot(health_data['OxygenSaturation'], bins=10, kde=True, ax=axes[2, 0])
```

```
axes[2, 0].set_title('Oxygen Saturation Distribution')

fig.delaxes(axes[2,1])  # remove unused subplot

plt.tight_layout()
plt.show()
print(summary_stats)
gender_counts = health_data['Gender'].value_counts()
correlation_matrix = health_data[['Age', 'HeartRate', 'RespiratoryR
fig, axes = plt.subplots(1, 2, figsize=(12, 6))
gender_counts.plot(kind='pie', ax=axes[0], autopct='%1.1f%%', startangle=90,
colors=['#ff9999','#66b3ff'])
axes[0].set_ylabel('')
axes[0].set_title('Gender Distribution')
sns.heatmap(correlation_matrix, annot=True, fmt=".2f", cmap='coolwarm',
ax=axes[1])
axes[1].set_title('Correlation Matrix')

plt.tight_layout()
plt.show()
plt.figure(figsize=(10, 6))
sns.boxplot(x='ActivityLevel', y='HeartRate', data=health_data)
plt.title('Heart Rate by Activity Level')
plt.ylabel('Heart Rate (beats per minute)')
plt.xlabel('Activity Level')
plt.show()
health_data[['SystolicBP', 'DiastolicBP']] = health_data['BloodPressure'].str.split('/',
expand=True).astype(int)

# blood pressure distribution
plt.figure(figsize=(12, 6))
sns.histplot(health_data['SystolicBP'], color="skyblue", label="Systolic", kde=True)
sns.histplot(health_data['DiastolicBP'], color="red", label="Diastolic", kde=True)
plt.title('Blood Pressure Distribution')
```

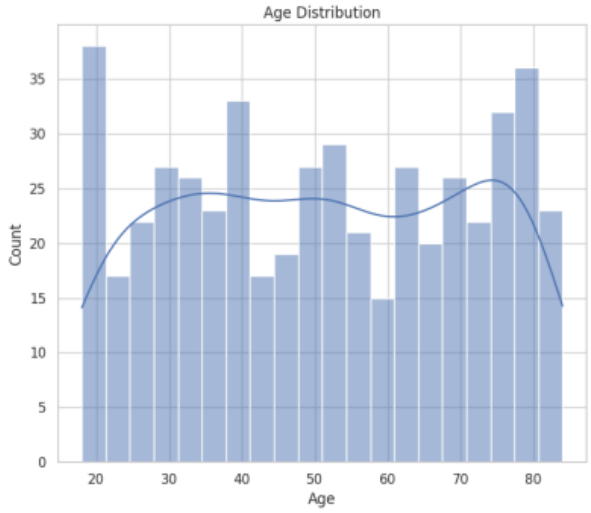
```
plt.xlabel('Blood Pressure (mmHg)')
plt.legend()
plt.show()

# health metrics by gender
fig, axes = plt.subplots(1, 2, figsize=(14, 6))
sns.boxplot(x='Gender', y='HeartRate', data=health_data, ax=axes[0])
axes[0].set_title('Heart Rate by Gender')
axes[0].set_xlabel('Gender')
axes[0].set_ylabel('Heart Rate (beats per minute)')

sns.boxplot(x='Gender', y='OxygenSaturation', data=health_data, ax=axes[1])
axes[1].set_title('Oxygen Saturation by Gender')
axes[1].set_xlabel('Gender')
axes[1].set_ylabel('Oxygen Saturation (%)')

plt.tight_layout()
plt.show()
sleep_quality_order = ['excellent', 'good', 'fair', 'poor']
stress_level_order = ['low', 'moderate', 'high']

# creating plots to examine relationships
fig, axes = plt.subplots(2, 2, figsize=(16, 12))

# heart rate by sleep quality
sns.boxplot(x='SleepQuality', y='HeartRate', data=health_data,
order=sleep_quality_order, ax=axes[0, 0])
axes[0, 0].set_title('Heart Rate by Sleep Quality')
axes[0, 0].set_xlabel('Sleep Quality')
axes[0, 0].set_ylabel('Heart Rate (beats per minute)')

# heart rate by stress level
sns.boxplot(x='StressLevel', y='HeartRate', data=health_data,
order=stress_level_order, ax=axes[0, 1])
```

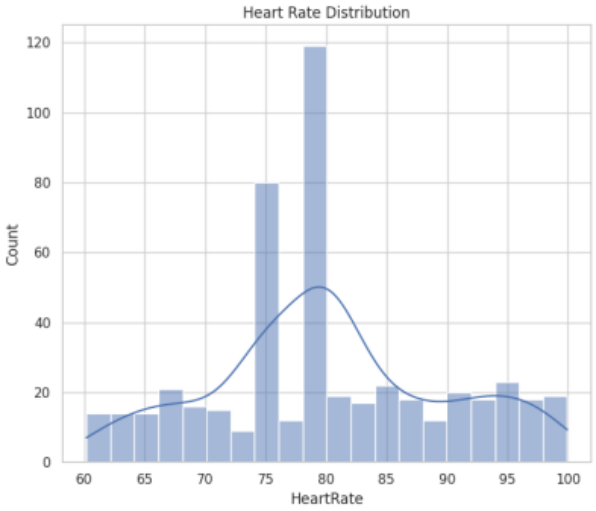
```
axes[0, 1].set_title('Heart Rate by Stress Level')
axes[0, 1].set_xlabel('Stress Level')
axes[0, 1].set_ylabel('Heart Rate (beats per minute)')

# oxygen saturation by sleep quality
sns.boxplot(x='SleepQuality', y='OxygenSaturation', data=health_data,
order=sleep_quality_order, ax=axes[1, 0])
axes[1, 0].set_title('Oxygen Saturation by Sleep Quality')
axes[1, 0].set_xlabel('Sleep Quality')
axes[1, 0].set_ylabel('Oxygen Saturation (%)')

# oxygen saturation by stress level
sns.boxplot(x='StressLevel', y='OxygenSaturation', data=health_data,
order=stress_level_order, ax=axes[1, 1])
axes[1, 1].set_title('Oxygen Saturation by Stress Level')
axes[1, 1].set_xlabel('Stress Level')
axes[1, 1].set_ylabel('Oxygen Saturation (%)')

plt.tight_layout()
plt.show()
fig, axes = plt.subplots(1, 2, figsize=(16, 6))

# respiratory rate by activity level
sns.boxplot(x='ActivityLevel', y='RespiratoryRate', data=health_data, ax=axes[0])
axes[0].set_title('Respiratory Rate by Activity Level')
axes[0].set_xlabel('Activity Level')
axes[0].set_ylabel('Respiratory Rate (breaths per minute)')

# body temperature by activity level
sns.boxplot(x='ActivityLevel', y='BodyTemperature', data=health_data, ax=axes[1])
axes[1].set_title('Body Temperature by Activity Level')
axes[1].set_xlabel('Activity Level')
axes[1].set_ylabel('Body Temperature (°F)')
```

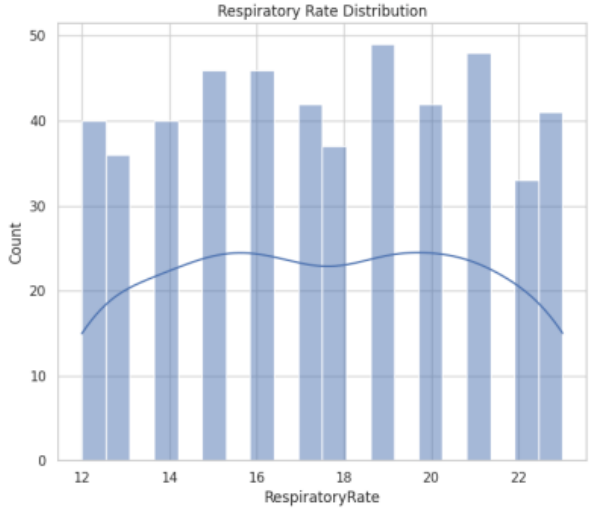
```
plt.tight_layout()
plt.show()
def age_group(age):
    if age <= 35:
        return 'Young'
    elif age <= 55:
        return 'Middle-aged'
    else:
        return 'Senior'

# function to categorize Blood Pressure
def bp_category(systolic, diastolic):
    if systolic < 120 and diastolic < 80:
        return 'Normal'
    elif 120 <= systolic < 140 or 80 <= diastolic < 90:
        return 'Elevated'
    elif 140 <= systolic < 160 or 90 <= diastolic < 100:
        return 'Hypertension Stage 1'
    else:
        return 'Hypertension Stage 2'

# function to categorize Heart Rate
def hr_category(hr):
    if hr < 60:
        return 'Low'
    elif hr <= 100:
        return 'Normal'
    else:
        return 'High'

# function to categorize Oxygen Saturation
def oxy_category(oxy):
    if oxy < 94:
        return 'Low'
```

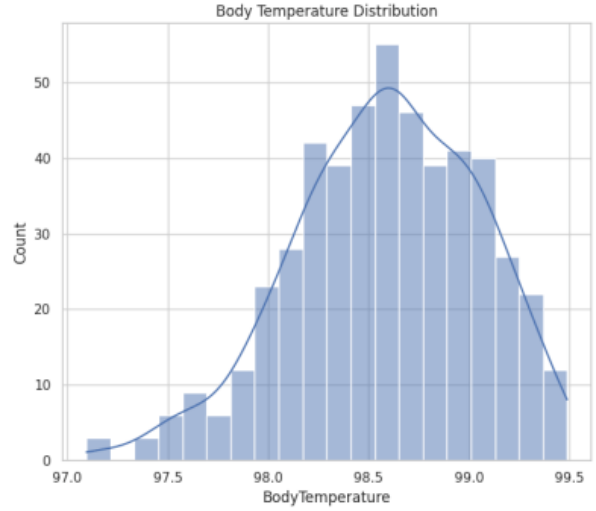
```
    else:
        return 'Normal'

# applying categorizations
health_data['AgeGroup'] = health_data['Age'].apply(age_group)
health_data['BPCategory'] = health_data.apply(lambda x:
bp_category(x['SystolicBP'], x['DiastolicBP']), axis=1)
health_data['HRCategory'] = health_data['HeartRate'].apply(hr_category)
health_data['OxyCategory'] = health_data['OxygenSaturation'].apply(oxy_category)
print(health_data[['Age', 'AgeGroup', 'SystolicBP', 'DiastolicBP', 'BPCategory',
'HeartRate', 'HRCategory', 'OxygenSaturation', 'OxyCategory']].head())
fig, axes = plt.subplots(2, 2, figsize=(16, 12))

# Age Group count plot
sns.countplot(x='AgeGroup', data=health_data, ax=axes[0, 0])
axes[0, 0].set_title('Distribution of Age Groups')

# Blood Pressure Category count plot
sns.countplot(x='BPCategory', data=health_data, ax=axes[0, 1])
axes[0, 1].set_title('Distribution of Blood Pressure Categories')

# Heart Rate Category count plot
sns.countplot(x='HRCategory', data=health_data, ax=axes[1, 0])
axes[1, 0].set_title('Distribution of Heart Rate Categories')

# Oxygen Saturation Category count plot
sns.countplot(x='OxyCategory', data=health_data, ax=axes[1, 1])
axes[1, 1].set_title('Distribution of Oxygen Saturation Categories')

# Show the plots
plt.tight_layout()
plt.show()
```

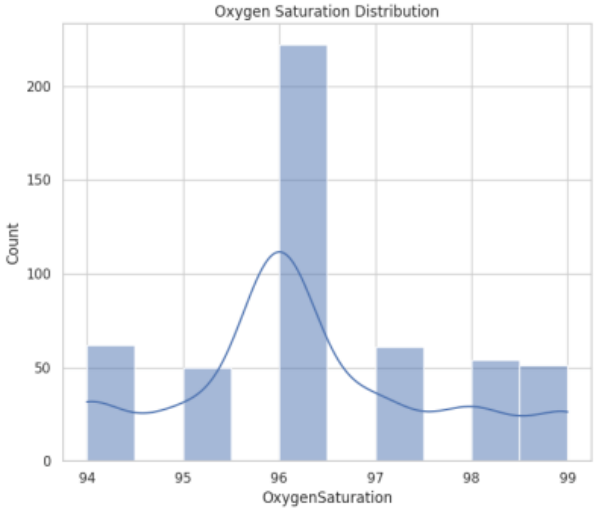
➢ Output :

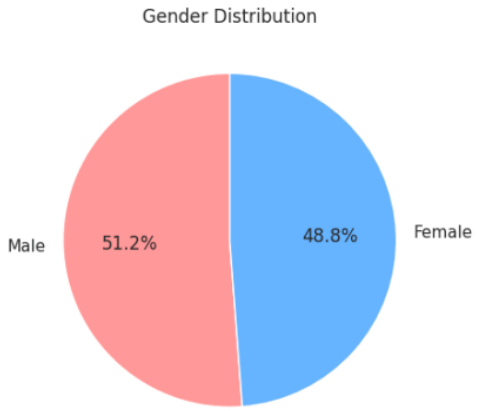

Age Distribution
Count
Age
0
5
10
15
20
25
30
35
20
30
40
50
60
70
80

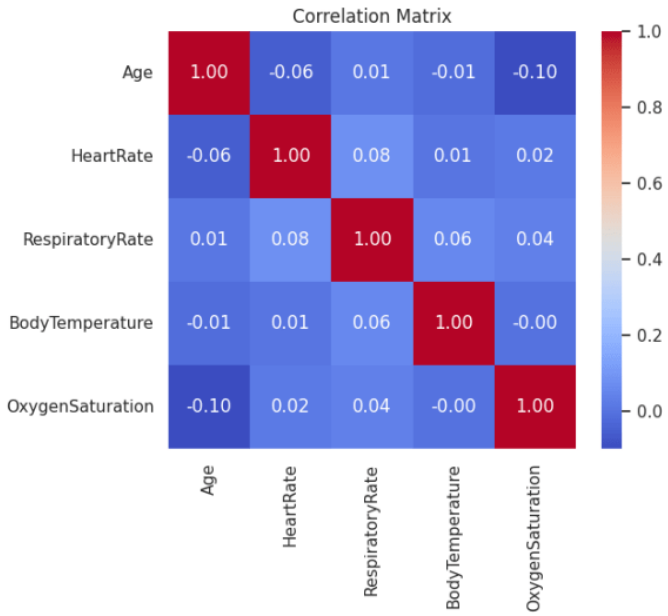

Heart Rate Distribution
Count
HeartRate
0
20
40
60
80
100
120
60
65
70
75
80
85
90
95
100

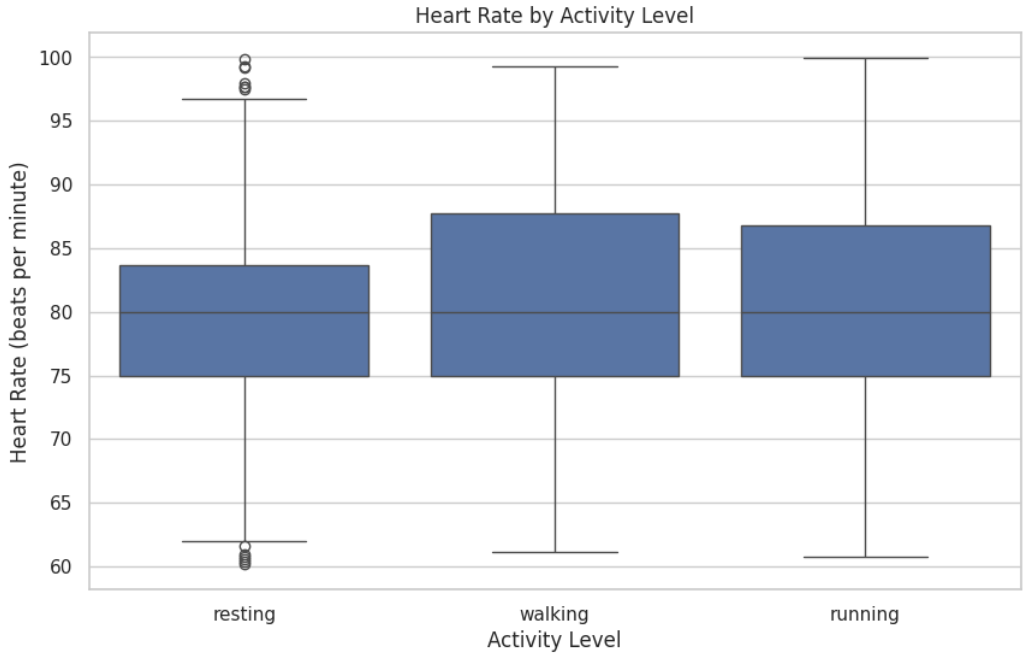

Respiratory Rate Distribution
Count
RespiratoryRate
0
10
20
30
40
50
12
14
16
18
20
22

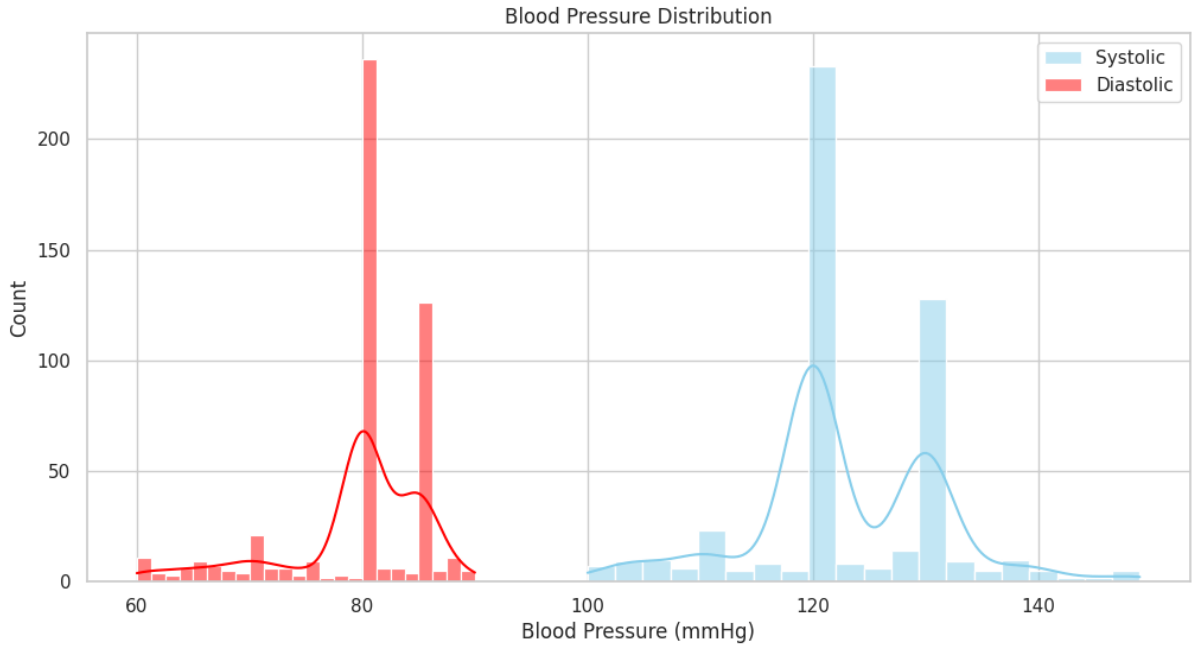

Body Temperature Distribution
Count
BodyTemperature
0
10
20
30
40
50
97.0
97.5
98.0
98.5
99.0
99.5

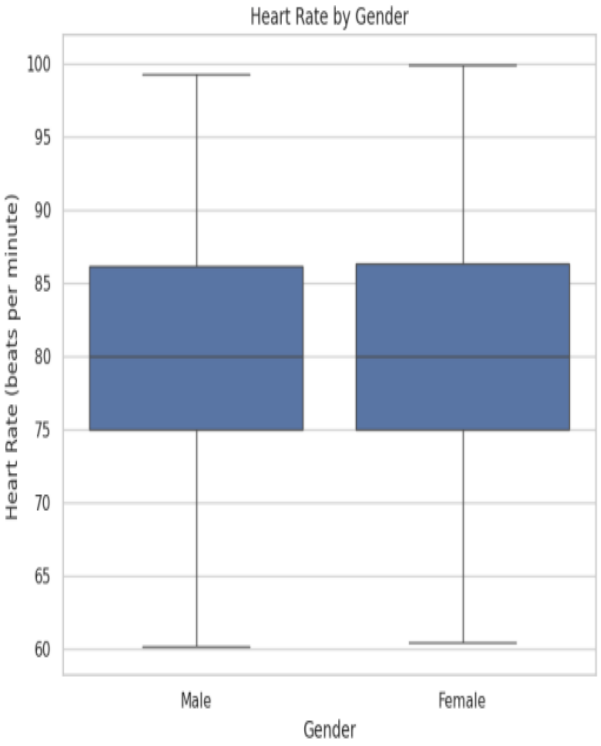

Oxygen Saturation Distribution
Count
OxygenSaturation
0
50
100
150
200
94
95
96
97
98
99

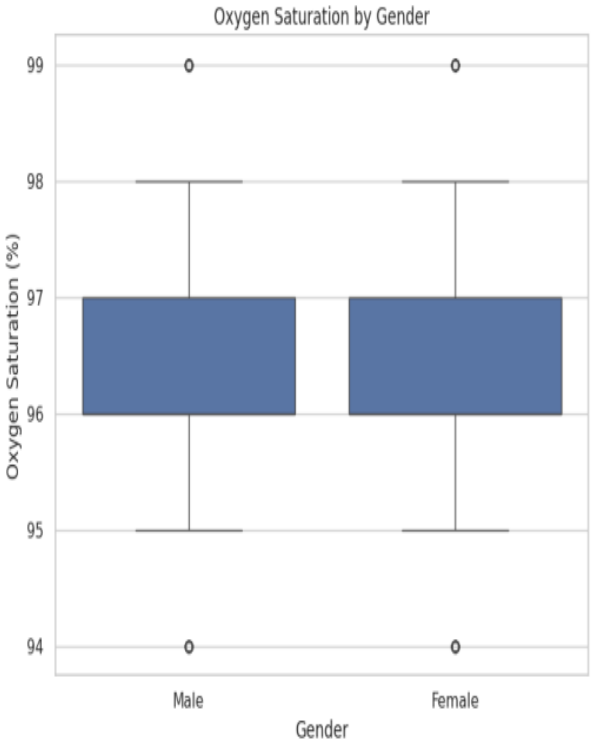

Gender Distribution
Male
51.2%
48.8%
Female

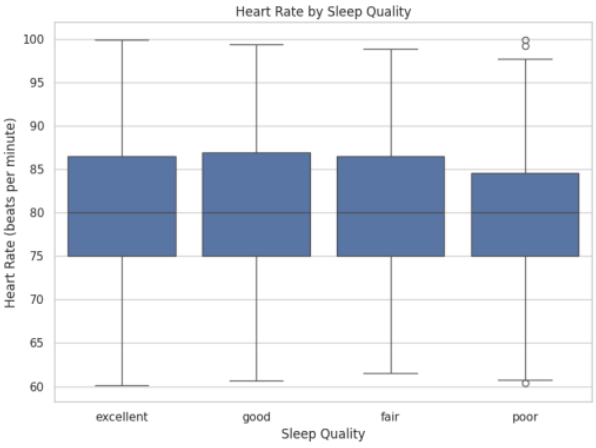

Correlation Matrix
Age
HeartRate
RespiratoryRate
BodyTemperature
OxygenSaturation
1.00 -0.06 0.01 -0.01 -0.10
-0.06 1.00 0.08 0.01 0.02
0.01 0.08 1.00 0.06 0.04
-0.01 0.01 0.06 1.00 -0.00
-0.10 0.02 0.04 -0.00 1.00
1.0
0.8
0.6
0.4
0.2
0.0

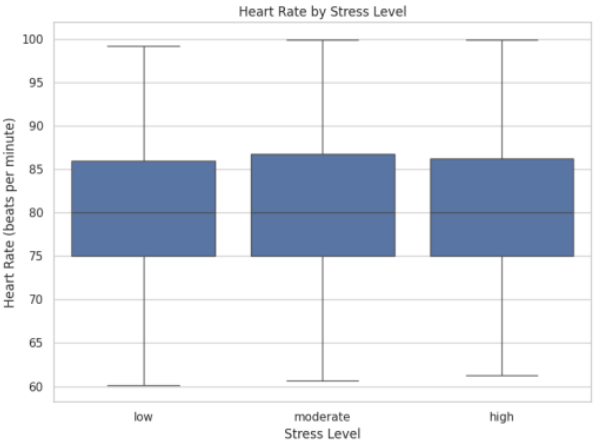

Heart Rate by Activity Level
Heart Rate (beats per minute)
100
95
90
85
80
75
70
65
60
resting
walking
running
Activity Level

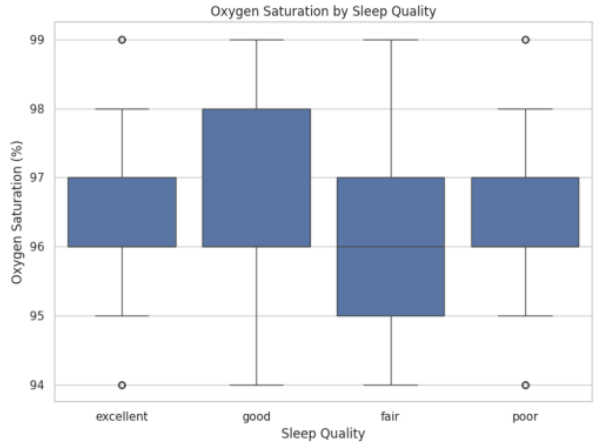

Blood Pressure Distribution
Systolic
Diastolic
Count
0
50
100
150
200
60
80
100
120
140
Blood Pressure (mmHg)

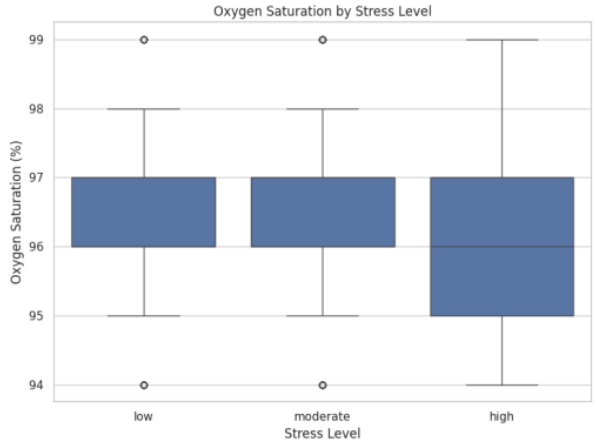

Heart Rate by Gender
Heart Rate (beats per minute)
100
95
90
85
80
75
70
65
60
Male
Female
Gender

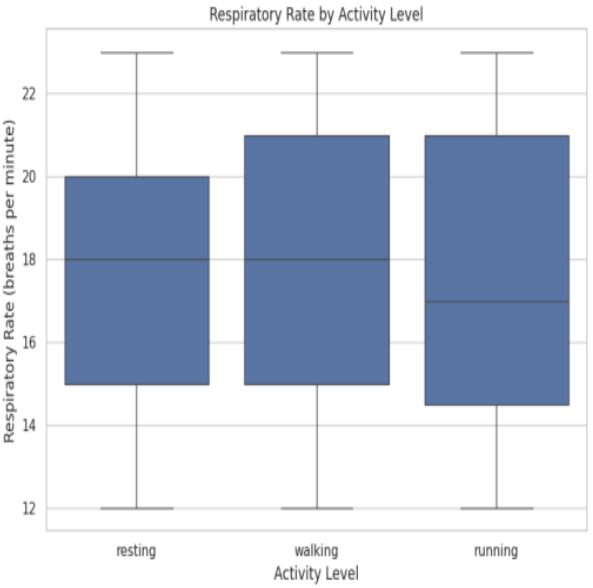

Oxygen Saturation by Gender
Oxygen Saturation (%)
99
98
97
96
95
94
Male
Female
Gender

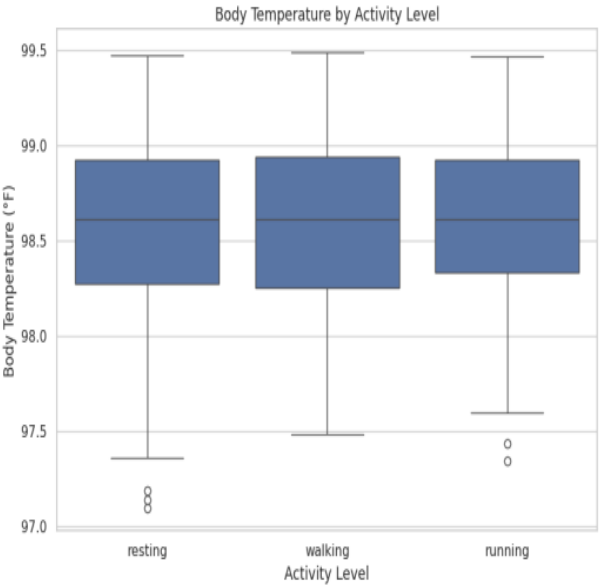

Heart Rate by Sleep Quality
Heart Rate (beats per minute)
100
95
90
85
80
75
70
65
60
excellent
good
fair
poor
Sleep Quality

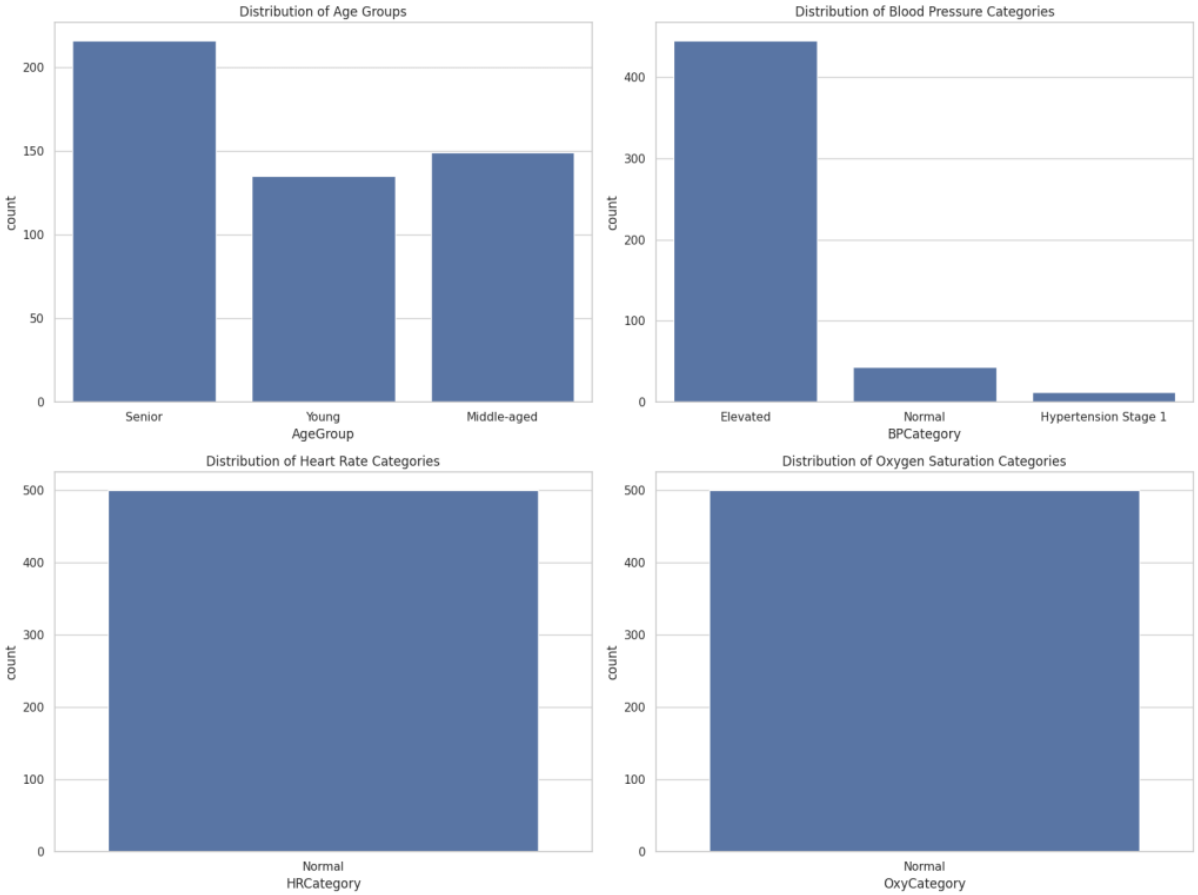

Heart Rate by Stress Level
Heart Rate (beats per minute)
100
95
90
85
80
75
70
65
60
low
moderate
high
Stress Level


Oxygen Saturation by Sleep Quality
Oxygen Saturation (%)
99
98
97
96
95
94
excellent
good
fair
poor
Sleep Quality


Oxygen Saturation by Stress Level
Oxygen Saturation (%)
99
98
97
96
95
94
low
moderate
high
Stress Level


Respiratory Rate by Activity Level
Respiratory Rate (breaths per minute)
22
20
18
16
14
12
resting
walking
running
Activity Level


Body Temperature by Activity Level
Body Temperature (°F)
99.5
99.0
98.5
98.0
97.5
97.0
resting
walking
running
Activity Level


Distribution of Age Groups
count
AgeGroup
Senior
Young
Middle-aged
0
50
100
150
200
Distribution of Blood Pressure Categories
count
BPCategory
Elevated
Normal
Hypertension Stage 1
0
100
200
300
400
Distribution of Heart Rate Categories
count
HRCategory
Normal
0
100
200
300
400
500
Distribution of Oxygen Saturation Categories
count
OxyCategory
Normal
0
100
200
300
400
500