Healthcare

Exploring Drowsiness Patterns in Wearable Device Data

Background:

You are a data scientist working for a wearable technology company that produces smartwatches with vital signs sensors. These sensors monitor heart rate and PPG (Photoplethysmography) signals, which include variations in green, red, and infrared light. One of the key features of your company's smartwatch is its ability to detect and alert users to potential drowsiness based on their physiological data.

Objective:

Your task is to perform an Exploratory Data Analysis (EDA) on a dataset collected from these smartwatches. The dataset includes various physiological parameters along with a 'drowsiness' label, which indicates the level of sleepiness based on an adapted Karolinska Sleepiness Scale (KSS).

Dataset Details:

- Columns:
  - heartRate: Heart rate readings from the smartwatch sensors.
  - ppgGreen, ppgRed, ppgIR: PPG (Photoplethysmography) sensor readings in green, red, and infrared wavelengths respectively.
  - drowsiness: Label indicating the level of drowsiness based on an adapted Karolinska Sleepiness Scale (KSS). Values range from 0.0 to 2.0, where 0.0 represents alertness and 2.0 represents significant drowsiness.

Dataset Access:

- *Link: Drowsiness Dataset on Kaggle*
- *Download Instructions:*
  1. *Visit the provided Kaggle dataset link.*
  2. *Sign in to your Kaggle account (or create one if you don't have it).*
  3. *Click on the "Download" button to download the dataset (drowsiness_dataset.csv).*

Imagine you've just been assigned this project. Your first step is to familiarize yourself with the dataset. You start by loading and examining the data to understand its structure and contents. You notice columns such as heart rate, PPG signals (green, red, and infrared), and the drowsiness level. Each row represents a time-stamped measurement from a smartwatch wearer.

As you dive deeper into the dataset, you begin plotting histograms and distributions to visualize the distribution of each variable. You observe how heart rate and PPG signals vary across different levels of drowsiness. You also calculate summary statistics to understand the central tendencies and variations in these physiological measures.

Next, you explore potential correlations between variables. Are there any patterns indicating that certain PPG signals correlate with higher or lower drowsiness levels? You use scatter plots and correlation matrices to investigate these relationships.

Throughout your analysis, you ensure to handle missing data and outliers appropriately, using techniques such as imputation and robust statistical methods. Finally, you summarize your findings in a comprehensive report, highlighting key insights and recommendations for further research or product development.

Conclusion:

By the end of your project, you've not only gained valuable insights into the drowsiness patterns captured by the smartwatches but also provided actionable recommendations based on your EDA. Your findings could potentially influence future iterations of the smartwatch's drowsiness detection algorithm, making it more effective in real-world scenarios.

How to Submit Your Project

1. **Prepare Your Report**: Your final report should be in a well-documented Jupyter notebook and detailed PDF report. Ensure that all code and visualizations are included, along with explanations and insights.

2. **Submit Your Work**: Email your completed project to LMS

3. **Deadline**: Ensure that you submit your project by the specified deadline provided in your course or project guidelines.

For your reference:

```
# Define the number of periods
num_periods = 4

# Calculate the size of each period
period_size = len(data) // num_periods

# Create a period column
period_labels = ['Morning', 'Afternoon', 'Evening', 'Night']
data['period'] = pd.cut(data.index, bins=num_periods, labels=period_labels)

# Check the distribution of periods
print(data['period'].value_counts())

# Segment the data by period
morning_data = data[data['period'] == 'Morning']
afternoon_data = data[data['period'] == 'Afternoon']
evening_data = data[data['period'] == 'Evening']
night_data = data[data['period'] == 'Night']
```

Analyzing and Visualizing Correlations for Each Period: Calculate and visualize the correlation between drowsiness levels and other variables (heart rate and PPG readings) for each period separately.

```
import matplotlib.pyplot as plt

def calculate_and_plot_correlations(data, period_name):
    correlation_heart_rate = data['drowsiness'].corr(data['heartRate'])
    correlation_ppg_green = data['drowsiness'].corr(data['ppgGreen'])
    correlation_ppg_red = data['drowsiness'].corr(data['ppgRed'])
    correlation_ppg_ir = data['drowsiness'].corr(data['ppgIR'])

    print(f'Correlation between drowsiness and heart rate ({period_name}): {correlation_heart_rate}')
    print(f'Correlation between drowsiness and PPG Green ({period_name}): {correlation_ppg_green}')
    print(f'Correlation between drowsiness and PPG Red ({period_name}): {correlation_ppg_red}')
    print(f'Correlation between drowsiness and PPG IR ({period_name}): {correlation_ppg_ir}')

    plt.scatter(data['heartRate'], data['drowsiness'], alpha=0.5, label='Heart Rate')
    plt.scatter(data['ppgGreen'], data['drowsiness'], alpha=0.5, label='PPG Green', color='green')
    plt.scatter(data['ppgRed'], data['drowsiness'], alpha=0.5, label='PPG Red', color='red')
    plt.scatter(data['ppgIR'], data['drowsiness'], alpha=0.5, label='PPG IR', color='purple')
    plt.title(f'Drowsiness Levels vs. Heart Rate and PPG Readings ({period_name})')
    plt.xlabel('Activity Level')
    plt.ylabel('Drowsiness Level')
    plt.legend()
    plt.show()
```

```
# Calculate and plot correlations for each period
calculate_and_plot_correlations(morning_data, 'Morning')
calculate_and_plot_correlations(afternoon_data, 'Afternoon')
calculate_and_plot_correlations(evening_data, 'Evening')
calculate_and_plot_correlations(night_data, 'Night')
```