The **WESAD** dataset (Wearable Stress and Affect Detection) is a publicly available multimodal dataset intended to help researchers develop and evaluate methods for stress and affect detection using wearable sensors. Below is an overview of what WESAD is, how it was collected, and what it contains.

**1. Background**

* **Goal**: Provide a multimodal dataset for developing algorithms and models that detect or classify human stress and affective states.
* **Publication**: Introduced in the paper by Schmidt et al. (2018) titled [*Introducing WESAD, a Multimodal Dataset for Wearable Stress and Affect Detection*](https://arxiv.org/abs/1810.00036).
* **Usage**: Widely used in research on affective computing, mental health monitoring, and stress detection systems.

**2. Participants and Protocol**

1. **Number of Participants**:
   * 15 subjects in total (12 males, 3 females), each with no known health issues that might interfere with the data collection (e.g., cardiovascular problems).
2. **Experimental Protocol**:
   * Participants underwent a lab study designed to induce multiple affective states, including stress, amusement, and baseline/relaxed states.
   * The protocol typically included:
     1. **Baseline / Neutral** period (sitting comfortably, relaxed state),
     2. **Amusement** period (subjects watched amusing video clips or were otherwise stimulated to induce an amused affective state),
     3. **Stress** period (subjects underwent a stress-inducing task, often a variant of the Trier Social Stress Test, which involves simulated public speaking and/or mental arithmetic under pressure),
     4. Sometimes an additional **Meditation** or **Recovery** period was included to return subjects to a calmer state.
   * Each phase typically lasts several minutes, giving enough time to capture physiological changes.

**3. Sensors and Measurements**

**Chest-Worn Device: RespiBAN Professional**

* **ECG (Electrocardiogram)**: Measures electrical activity of the heart (useful for Heart Rate (HR), Heart Rate Variability (HRV), etc.).
* **EDA (Electrodermal Activity)**: Indicates skin conductance level and response peaks related to sympathetic nervous system arousal.
* **EMG (Electromyogram)**: Measures muscle activity (e.g., from chest muscles); can capture respiration artifacts or muscle tension changes in stress.
* **Respiration**: Measures breathing rate and amplitude (helpful for stress detection, as breathing rate can change under stress).
* **Inertial (3-axis Acceleration)**: Captures motion or posture changes.

**Wrist-Worn Device: Empatica E4**

* **PPG (Photoplethysmography)**: Used to measure blood volume pulse, which can be analyzed for heart rate and additional hemodynamic features.
* **EDA (Electrodermal Activity)**: Another EDA channel from the wrist, complementary to the chest-worn measurement.
* **Temperature**: Skin temperature from the wrist, which can change under stress.
* **Inertial (3-axis Acceleration)**: Motion of the wrist.

**4. Dataset Contents and Structure**

1. **Raw Data Files**:
   * Each participant’s data is contained in a structured format (often .pkl files in the original release).
   * Data channels include: ECG, EDA, EMG, respiration, temperature, PPG, and accelerometer signals.
2. **Preprocessed Data**:
   * The dataset providers often supply partially preprocessed or segmented data.
   * Preprocessing typically includes filtering, removing motion artifacts, normalizing or z-scoring signals, and segmenting them by experimental condition.
3. **Labels**:
   * The dataset is labeled by segment (e.g., “baseline,” “stress,” “amusement,” “meditation”) so that researchers can directly map sensor data to affective states.
   * Labels include timestamps to identify the start and end of each experimental condition.

**5. Typical Use Cases**

* **Stress Detection and Classification**:  
  Train machine-learning or deep-learning models to classify whether a person is in a stressed versus non-stressed (baseline, amusement, meditation) state.
* **Affect Recognition**:  
  Go beyond stress vs. non-stress to differentiate among multiple emotional or affective states (e.g., “amused,” “relaxed,” “stressed”).
* **Feature Exploration**:  
  Researchers often use WESAD to explore which physiological features are most predictive of stress or affect (e.g., features from ECG, EDA peaks, heart rate variability metrics, etc.).
* **Multimodal Fusion**:  
  Combine chest-based signals (ECG, respiration, EMG) with wrist-based signals (PPG, EDA, temperature) to improve prediction accuracy or robustness.

**6. Key Advantages and Challenges**

**Advantages**

1. **Multimodality**: Offers multiple sensor channels from two different device locations (chest and wrist). This variety encourages advanced data fusion strategies.
2. **Well-Labeled Protocol**: Clear timing and labeling of stress, amusement, and baseline phases.
3. **Public and Well-Documented**: Freely available for academic research, with clear documentation on sensor specifications and experiment design.

**Challenges**

1. **Limited Number of Participants**: Only 15 subjects; some cross-validation approaches can mitigate this but generalizability can be limited.
2. **Motion Artifacts**: Movements (particularly from wrist accelerometer) can introduce noise into signals like EDA or ECG.
3. **Data Imbalance**: Some segments (e.g., stress vs. baseline) may have different durations or varying intensities among participants, requiring care during training and evaluation.

**7. Accessing WESAD**

* **Availability**: The WESAD dataset is typically hosted on various academic data repositories. It’s free for research purposes but typically requires agreement to specific terms of use.
* **Request Process**: You usually fill out a data request form or email the authors to obtain access.

**Summary**

In essence, the **WESAD dataset** is a rich, multimodal collection of physiological signals gathered from both chest and wrist wearables. It is carefully annotated to support research on stress detection and affective computing. By leveraging signals such as ECG, EDA, EMG, respiration, PPG, temperature, and accelerometer readings, WESAD makes it possible to develop and benchmark algorithms that recognize when someone is stressed or amused, thus contributing to wearable-based health monitoring and well-being applications.

**1. Loading a Subject’s Data**

A typical way to load a subject’s data in Python is:

python

CopyEdit

import pickle

subject\_id = 2 # example subject "S2"

filename = f"S{subject\_id}.pkl"

with open(filename, 'rb') as f:

data = pickle.load(f)

After this, data becomes a Python dictionary with keys such as:

* data['signal']
* data['label']
* data['subject']
* data['questionnaire'] (if available)

**2. Dictionary Structure**

**Top-Level Keys**

1. **subject**
   * The subject ID or any metadata (e.g., "2" for S2).
2. **signal**
   * A dictionary containing sensor data from:
     + Chest device (e.g., RespiBAN)
     + Wrist device (Empatica E4)
3. **label**
   * An array of numeric labels for each time sample.
   * Each integer corresponds to a particular experiment phase (e.g., 0 = baseline, 1 = stress, 2 = amusement, etc.).
4. **questionnaire** (optional)
   * Some files include self-report questionnaire data or additional metadata.

**signal Dictionary**

Inside data['signal'] there are typically two main keys:

1. **chest** – Data from the RespiBAN chest-worn device
2. **wrist** – Data from the Empatica E4 wrist-worn device

Both are dictionaries themselves, containing arrays for each sensor channel.

**2.1 Chest Signals**

python

CopyEdit

data['signal']['chest'] # Dictionary with keys like 'ACC', 'ECG', 'EMG', 'EDA', 'Temp', 'Resp'

Typical keys in data['signal']['chest'] include:

* **ACC**: 3-axis accelerometer from the chest device
* **ECG**: Electrocardiogram signal (heart electrical activity)
* **EMG**: Electromyogram (muscle activity)
* **EDA**: Electrodermal activity from the chest patch
* **Temp**: Skin temperature from the chest patch
* **Resp**: Respiration signal

All of these signals typically share a uniform (high) sampling rate around 700 Hz (except ACC which can sometimes be lower, depending on firmware). You can verify the exact sampling rate for each signal in the WESAD documentation.

For example:

python

CopyEdit

chest\_data = data['signal']['chest']

for key in chest\_data.keys():

print(key, chest\_data[key].shape)

This might print something like:

scss

CopyEdit

ACC (N, 3)

ECG (N,)

EMG (N,)

EDA (N,)

Temp (N,)

Resp (N,)

where N is the number of samples in that recording. ACC has 3 columns for x, y, z axes.

**2.2 Wrist Signals**

python

CopyEdit

data['signal']['wrist'] # Dictionary with keys like 'ACC', 'BVP', 'EDA', 'TEMP'

Typical keys in data['signal']['wrist'] include:

* **ACC**: 3-axis accelerometer from the Empatica E4 (sampled at 32 Hz)
* **BVP** (or sometimes PPG): Blood volume pulse from photoplethysmography (64 Hz)
* **EDA**: Electrodermal activity from the wrist (4 Hz)
* **TEMP**: Temperature from the wrist (4 Hz)

For example:

python

CopyEdit

wrist\_data = data['signal']['wrist']

for key in wrist\_data.keys():

print(key, wrist\_data[key].shape)

Output might look like:

scss

CopyEdit

ACC (N\_acc, 3)

BVP (N\_bvp,)

EDA (N\_eda,)

TEMP (N\_temp,)

Because each channel has its own sampling rate, the length of each array (N\_acc, N\_bvp, etc.) will differ.

**3. Labels and Annotations**

The **data['label']** array aligns with the highest-resolution device sampling (often the chest device). Labels are typically integers representing:

* 0: Baseline / Neutral
* 1: Stress
* 2: Amusement
* (Other numbers or sections might include meditation, recovery, etc.)

You’ll need to map these integer labels to time windows or to the signals in order to segment the data by condition.

For example:

python

CopyEdit

labels = data['label']

print(labels.shape) # might be (N,) aligned with the chest sampling rate

print(set(labels)) # e.g., {0, 1, 2}

**4. Sampling Rates (Typical)**

Below is a quick reference for **common** sampling rates in WESAD, though always check the dataset’s official documentation because firmware versions can vary:

* **Chest (RespiBAN)**:
  + ECG, EMG, EDA, Resp: 700 Hz
  + ACC: 700 Hz (in many releases; sometimes 128 Hz or 64 Hz in other datasets)
  + Temp: 700 Hz (often derived from the same internal ADC but sometimes recorded at lower effective rate)
* **Wrist (Empatica E4)**:
  + ACC: 32 Hz
  + BVP: 64 Hz
  + EDA: 4 Hz
  + TEMP: 4 Hz

**5. Putting It All Together (Example Code)**

Here is a minimal code snippet that loads a subject, inspects data shapes, and prints out some basic info:

python

CopyEdit

import pickle

subject\_id = 2

filename = f"S{subject\_id}.pkl"

with open(filename, 'rb') as f:

data = pickle.load(f)

print("Subject:", data['subject'])

print("Available top-level keys:", data.keys())

print("\nChest signals:", data['signal']['chest'].keys())

print("Wrist signals:", data['signal']['wrist'].keys())

# Print shapes of chest signals

print("\nChest data shapes:")

for key, val in data['signal']['chest'].items():

print(f" {key}: {val.shape}")

# Print shapes of wrist signals

print("\nWrist data shapes:")

for key, val in data['signal']['wrist'].items():

print(f" {key}: {val.shape}")

# Labels

labels = data['label']

unique\_labels = set(labels)

print("\nLabel array shape:", labels.shape)

print("Unique labels:", unique\_labels)

Example output might look like:

less

CopyEdit

Subject: 2

Available top-level keys: dict\_keys(['subject', 'signal', 'label', 'questionnaire'])

Chest signals: dict\_keys(['ACC', 'ECG', 'EMG', 'EDA', 'Temp', 'Resp'])

Wrist signals: dict\_keys(['ACC', 'BVP', 'EDA', 'TEMP'])

Chest data shapes:

ACC: (123456, 3)

ECG: (123456,)

EMG: (123456,)

EDA: (123456,)

Temp: (123456,)

Resp: (123456,)

Wrist data shapes:

ACC: (7890, 3)

BVP: (15800,)

EDA: (1972,)

TEMP: (1972,)

Label array shape: (123456,)

Unique labels: {0, 1, 2}

**6. Key Points to Remember**

1. **Multirate Data**: The chest device is usually sampled at a much higher rate than the wrist device. You’ll need to handle resampling or synchronization if you want to combine signals.
2. **Segmentation by Condition**: The label array is crucial for segmenting data into baseline, stress, or amusement segments.
3. **Possible Variation**: Different subjects may have slightly different recording lengths; data collection might start or end at slightly different times.
4. **Preprocessing**: Many researchers filter, resample, and normalize these signals before building stress-detection or affect-recognition models.

**In summary**, the WESAD dataset in Python is structured as a nested dictionary within a .pkl file. The signal entry houses sensor data from chest (chest) and wrist (wrist) devices, each containing multiple channels with different shapes and sampling rates. Labels are provided in a separate array that typically aligns with the highest-frequency chest device.

o1