Quantifying Peak States Using Physiological Signals: Breathing Rate and Sound Amplitude

Joel Dietz

July 3, 2024

Abstract

This study investigates the use of physiological signals, specifically breathing rate and sound amplitude, to quantify the intensity and duration of peak states. By establishing thresholds for these parameters, we tracked the intensity over time, providing insights into the fluctuations and characteristics of these states, including calculating the area under the curve once a threshold is reached to assess the overall impact of an orgasm and improvement in orgasms over time.

1 Introduction

Peak states, such as orgasms, are characterized by distinct physiological responses. Measuring these responses can offer valuable insights into the intensity and duration of such states. This study employs breathing rate and sound amplitude as proxies to identify and quantify these peak experiences.

2 Methodology

2.1 Data Collection

We used sample data to simulate minute-level observations of breathing rate and sound amplitude over seven days. The thresholds for identifying peak states were set at 22 breaths per minute (bpm) and 73 decibels (dB) for breathing rate and sound amplitude, respectively.

2.2 Data Analysis

The intensity of the peak state was calculated using a normalized sum of breathing rate and sound amplitude values exceeding their respective thresholds. The duration of peak states was determined by counting the minutes during which both parameters remained above the thresholds.

3 Results

Figure 2 illustrates the intensity of peak states over time, based on the combined measure of breathing rate and sound amplitude.

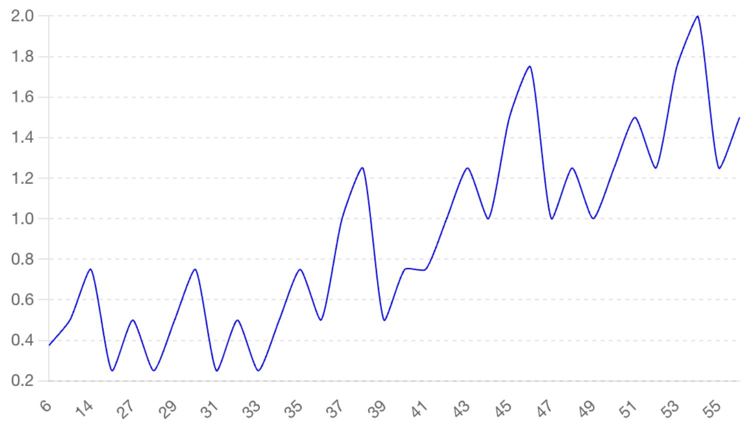

Figure 1: Intensity of Peak States Based on Breathing Rate and Sound Amplitude

Table 1 presents the sample data used in this study.

Table 1: Sample Data: Breathing Rate and Sound Amplitude

Day	Minute	Breathing Rate (bpm)	Sound Amplitude (dB)
1	1	18	68
1	2	20	70
1	3	21	71
1	4	20	72
1	5	22	74
1	6	23	75
...	...	...	...
7	5	29	80
7	6	30	81

4 Discussion

The intensity curve demonstrates the fluctuations in peak state intensity over time. This approach allows for a nuanced understanding of how physiological responses correlate with subjective experiences of peak states.

5 Python Script for AUC Calculation and Plotting

The following Python script calculates the area under the curve (AUC) for each peak state based on breathing rate and sound amplitude data, and plots the AUC values over time.

```python
import numpy as np
import matplotlib.pyplot as plt

# Sample minute-level data
minute_level_data = [
    {'Breathing Rate (bpm)': [18, 20, 21, 20, 22, 23, 19, 20], 'Sound Amplitude (dB)': [68, 70, 71, 72, 74, 75, 70, 72]},  # Day 1
    {'Breathing Rate (bpm)': [21, 22, 23, 22, 24, 25, 21, 22], 'Sound Amplitude (dB)': [71, 72, 73, 74, 75, 76, 73, 74]},  # Day 2
    {'Breathing Rate (bpm)': [20, 21, 22, 23, 21, 22, 20, 21], 'Sound Amplitude (dB)': [70, 71, 72, 73, 72, 74, 71, 72]},  # Day 3
    {'Breathing Rate (bpm)': [22, 23, 24, 23, 24, 25, 23, 24], 'Sound Amplitude (dB)': [73, 74, 75, 74, 75, 76, 74, 75]},  # Day 4
    {'Breathing Rate (bpm)': [23, 24, 25, 24, 26, 27, 24, 25], 'Sound Amplitude (dB)': [74, 75, 76, 75, 77, 78, 75, 76]},  # Day 5
    {'Breathing Rate (bpm)': [25, 26, 27, 26, 28, 29, 26, 27], 'Sound Amplitude (dB)': [76, 77, 78, 77, 79, 80, 77, 78]},  # Day 6
    {'Breathing Rate (bpm)': [26, 27, 28, 27, 29, 30, 27, 28], 'Sound Amplitude (dB)': [77, 78, 79, 78, 80, 81, 78, 79]},  # Day 7
]

# Define thresholds for peak state
breathing_threshold = 22
sound_threshold = 73

# Function to calculate intensity
def calculate_intensity(br, sa):
    normalized_br = (br - breathing_threshold) / (30 - breathing_threshold)
    normalized_sa = (sa - sound_threshold) / (81 - sound_threshold)
    intensity = normalized_br + normalized_sa
    return intensity

# Initialize lists to store peak state AUCs and their start times
peak_aucs = []
peak_start_times = []

for day_index, day_data in enumerate(minute_level_data):
    breathing_rate = day_data['Breathing Rate (bpm)']
    sound_amplitude = day_data['Sound Amplitude (dB)']

    current_peak_intensities = []
```

```python
    current_start_time = None

    for minute_index, (br, sa) in enumerate(zip(breathing_rate,
    sound_amplitude)):
        time_index = day_index * len(breathing_rate) + minute_index
     + 1
        if br > breathing_threshold and sa > sound_threshold:
            intensity = calculate_intensity(br, sa)
            if current_start_time is None:
                current_start_time = time_index
            current_peak_intensities.append(intensity)
        else:
            if current_peak_intensities:
                auc = np.trapz(current_peak_intensities)
                peak_aucs.append(auc)
                peak_start_times.append(current_start_time)
                current_peak_intensities = []
                current_start_time = None

    # Capture any remaining peak at the end of the day
    if current_peak_intensities:
        auc = np.trapz(current_peak_intensities)
        peak_aucs.append(auc)
        peak_start_times.append(current_start_time)

# Plot the AUCs over time
plt.figure(figsize=(12, 6))
plt.plot(peak_start_times, peak_aucs, marker='o', linestyle='-',
    color='b')
plt.title('Area Under the Curve (AUC) for Each Peak State')
plt.xlabel('Time (minutes)')
plt.ylabel('AUC')
plt.grid(True)
plt.show()
```

Listing 1: Python script for AUC calculation and plotting

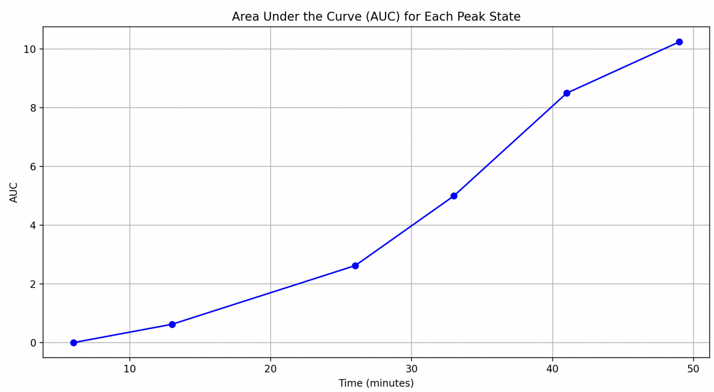

Figure 2: Area under the curve (AUC) for peak states

6 Conclusion

By leveraging physiological signals such as breathing rate and sound amplitude, this study provides a framework for quantifying the intensity and duration of peak states. Future research can expand on these methods to include additional parameters and larger datasets for more comprehensive analysis.

References

[1] J. Chiaverini, B. R. Blakestad, J. W. Britton, J. D. Jost, C. Langer, D. G. Leibfried, R. Ozeri, and D. J. Wineland, "Surface-electrode architecture for ion-trap quantum information processing," *Quantum Information and Computation*, vol. 5, no. Quantum Information and Computation, 2005.

[2] D. LEIBFRIED, R. OZERI, and D. WINELAND, "Surface-electrode architecture for ion-trap quantum information processing," *Quantum Information and Computation*, vol. 5, no. 6, pp. 419–439, 2005.

[3] S. Seidelin, J. Chiaverini, R. Reichle, J. J. Bollinger, D. Leibfried, J. Britton, J. Wesenberg, R. Blakestad, R. Epstein, D. Hume, *et al.*, "Microfabricated surface-electrode ion trap for scalable quantum information processing," *Physical review letters*, vol. 96, no. 25, p. 253003, 2006.

[4] N. Daniilidis, S. Narayanan, S. A. Möller, R. Clark, T. E. Lee, P. J. Leek, A. Wallraff, S. Schulz, F. Schmidt-Kaler, and H. Häffner, "Fabrication and heating rate study of microscopic surface electrode ion traps," *New Journal of Physics*, vol. 13, no. 1, p. 013032, 2011.