Acoustic Data-based Characterization of Incipient Boiling for Space Applications

Hailey Ernest
Statistics Dept.
Cal Poly, SLO
hernest@calpoly.edu

Elliot Kunz
Mathematics Dept.
Cal Poly, SLO
edkunz@calpoly.edu

Cameron Hafer
Statistics Dept.
Cal Poly, SLO
clhafer@calpoly.edu

Colin Hassett
Statistics Dept.
Cal Poly, SLO
cmhasset@calpoly.edu

***Abstract* — Localized heat leaks can lead to a multitude of issues, including loss of fuel or explosion of the fuel tank. Detecting incipient boiling caused by localized heat leaks is important for space-based cryogenic fuel systems in order to manage heat leaks. This project explores whether acoustic accelerometer data could be used to distinguish meaningful boiling behaviors through unsupervised machine learning. Building on prior work, our team reviewed earlier models, engineered new signal features, tested new dimensionality reduction and clustering methods, and improved visualization of cluster behavior and separation. Our goal was to produce results that were scientifically useful while staying interpretable to non-technical stakeholders. Our analysis showed clearer separation between boiling behaviors, specifically between rhythmic and non-rhythmic patterns. Clustering analyses demonstrated approximately 99% clustering stability across parameter changes. Our pipeline identifies 6 regimes by filtering out non-active boilings and includes the clusters: chaotic, sporadic, rhythmic, dense rhythmic, double rhythmic, and single rhythmic with high stability. Our analysis also revealed substructure within the rhythmic regime, suggesting additional physically meaningful boiling subtypes. The Gaussian mixture model assigns boiling runs to clusters with high accuracy. These results show that acoustic data can help identify meaningful boiling regimes and support future interpretable machine learning tools for boiling diagnostics.**

I. Introduction

Detecting early-stage boiling is important for maintaining stability in cryogenic fuel systems, especially in space environments where heat transfer behaves differently than on Earth. Localized heat leaks caused by thermal contraction and thermal bridging can lead to issues on a mission. Some of these issues include boil-off and propellant loss, tank over-pressurization, thermal stratification, pump cavitation, structural degradation, and long-duration mission failure.

This project focuses on building a clustering framework to group similar boiling signals and better understand differences between boiling behaviors. By grouping similar signals together, we begin to understand what features separate different types of boiling behavior and interpret what each cluster represents physically. This allows us to move from raw signal data toward meaningful physical insight into what is happening inside a fuel tank.

Our goal for this project was to produce meaningful clusters of acoustic boiling regimes so we could find interpretable feature differences between groups. The previous research in *Interpretable Machine Learning for Acoustic Classification of Incipient Boiling Regimes* produced clusters and visualizations using Principal Component Analysis, or PCA, for dimensionality reduction and K-means clustering to identify boiling regimes. However, these approaches assume a simple structure and may not capture more complex or irregular relationships in the data. In this project, we explored different dimensionality reduction and clustering algorithms to determine whether clearer separation between boiling behaviors could be achieved.

Another goal of our project is to produce an app designed to present our findings and interpret new data. The app is designed to help non-technical stakeholders visualize the separation between clusters, understand which features help distinguish the regimes, and compare new boiling runs to the patterns found in the existing dataset. When future experiments collect new data, including potential experiments in space, the app can project those runs into the learned feature space and determine whether the observed boiling behavior follows similar or different patterns from the Earth-based experimental data.

II. Background

A. Experimental Setup

The experimental design of this work follows exactly from the experimental setup in M. Khasin, J. Rogers, and V. Osipov, in *Acoustic Data-based Characterization of Incipient Boiling for Space Applications*. A diagram of the experimental setup is shown in Figure 1. The system includes a cryogenic tank with two accelerometers placed on its surface. A controlled heat source is used to induce localized boiling, and the resulting acoustic signals are recorded by the accelerometers. The accelerometers were placed randomly for each run on the tank compared to the heater's position to simulate different positions where boiling may occur in the tank.

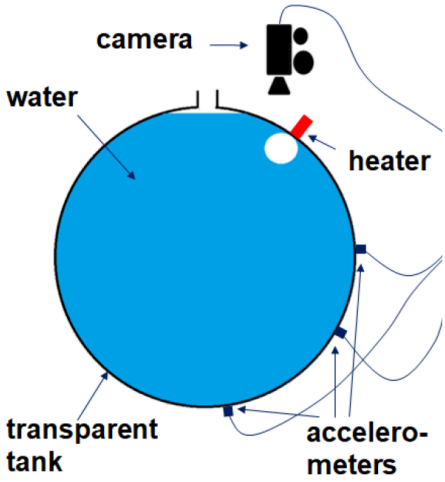


Figure 1.—Experimental setup (M. Khasin, J. Rogers, and V. Osipov, "Acoustic Data-based Characterization of Incipient Boiling for Space Applications, ")

These acoustic signals capture information related to the physical behavior of bubble formation and collapse during boiling. Since boiling behavior produces vibrations that travel through the tank structure, accelerometer data can be used as a non-invasive way to observe changes in the boiling regime.

B. Data Format and Visualizations

The dataset contains 446 experimental runs, with recording lengths ranging from 0.1 seconds to 30 seconds. Each run represents a recording of boiling activity, where signal variations correspond to changes in bubble dynamics over time. Since each experiment includes data from two accelerometers, signals from both sensors were visualized and analyzed together.

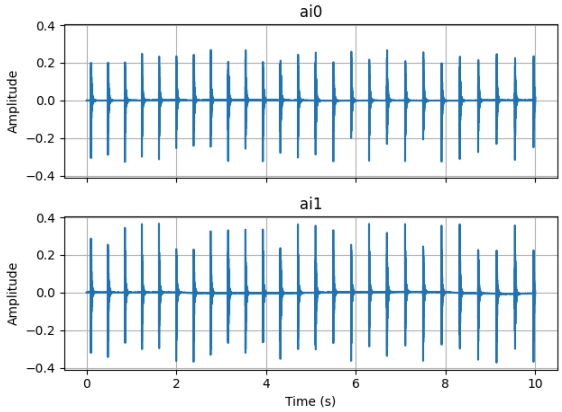


Figure 2: Raw signal from a representative boiling experiment with the data gathered from the first accelerometer above the signal gathered by the second accelerometer.
(Run: MATLAB 3-26 PM Wed, Sep 18, 2024 Run12)

Using both accelerometers provides a more complete view of the boiling behavior. Since the sensors may capture different signal magnitudes, peak timings, or local vibration patterns, analyzing both channels can provide more information than using only one sensor. These differences may help distinguish between boiling regimes that would otherwise appear similar.

C. Previous Research

After reviewing the previous work by Gupta et al. in *Interpretable Machine Learning for Acoustic Classification of Incipient Boiling Regimes*, we identified several important limitations in the earlier unsupervised modeling approach. The previous group used accelerometer data to classify boiling behavior through an interpretable machine learning pipeline. Their work focused on extracting features from the raw signals, reducing the feature space with Principal Component Analysis (PCA), and clustering the data using K-Means.

One important limitation in the previous work was the assumption that the two accelerometer signals did not contain meaningful differences. Thus, the previous pipeline only used the signal from the first accelerometer for each run. However, after exploring the signals ourselves, we found that there were noticeable differences between the two accelerometer channels. In some runs, the two sensors showed different peak magnitudes, different trimming of peak amplitudes, and different overall signal patterns. This suggested that the second accelerometer contained useful information that could help distinguish between boiling regimes. As a result, our project uses both accelerometer signals and added cross-channel features to capture the relationship between the two sensors.

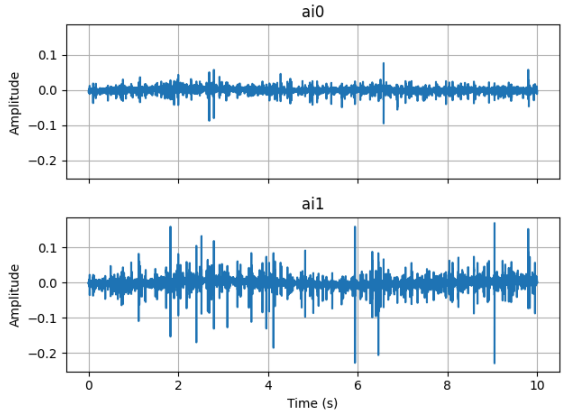


Figure 3: Representative sample showing the large differences in signal between accelerometer 1 and accelerometer 2 in both magnitude and time of peak amplitudes.
(Run: MATLAB 2-25 PM Tue, Jun 4, 2024 Run3)

The previous work also used a smaller set of engineered features than our pipeline. The previous group used 15 features, while our project expanded the feature set to better capture differences in boiling behavior. During our review, we also found that some features, such as maximum amplitude, could negatively affect clustering. Since accelerometer placement was randomized relative to the heater, amplitude differences may reflect sensor position rather than actual differences in boiling regimes. For example, two runs from the same boiling regime could appear different because one

accelerometer was closer to the source of the vibration. To address this, our project focused on features that better captured timing, frequency behavior, peak structure, variability of magnitude, and relationships between the two accelerometer channels.

A major part of the previous pipeline was the use of principal component analysis, or PCA. PCA is a dimensionality reduction technique, meaning it takes a dataset with many features and compresses it into a smaller number of new variables. PCA is useful because it allows high-dimensional feature data to be visualized in two or three dimensions. It is also relatively interpretable because each principal component is made from a linear combination of the original features. By looking at the feature loadings, or the weight each feature has on a principal component, researchers can estimate which features are contributing most to the separation between runs. However, PCA has limitations. PCA is a linear model, which means it works best when most of the important patterns in the data can be described using a straight line. PCA also focuses on explaining variance, but the directions of highest variance are not always the same directions that best separate physical regimes. Because of this, PCA was useful as a baseline and helped make the previous work interpretable, but it may not have fully captured the more complex structure in the boiling data.

After applying PCA, the previous group used K-Means clustering. K-Means is an unsupervised clustering algorithm, meaning it groups data points without using known labels. The algorithm works by choosing a set number of clusters, called k, and assigning each data point to the nearest cluster center. K-Means is simple and easy to interpret, but it also makes assumptions about the structure of the data. It works best when clusters are roughly circular, similar in size, and clearly separated. It also requires the user to choose the number of clusters ahead of time. This was not ideal for this project as the true number of boiling regimes was not known in advance, and the clusters may not have had simple shapes or equal densities. Some runs also act as outliers or fall between regimes. Since K-Means assigns every point to a cluster, it does not naturally identify noise points or uncertain samples. This made it less ideal for our project, where we wanted to detect different boiling behaviors while also accounting for outliers and potential subclusters.

Based on these limitations, our project built on the previous pipeline by using both accelerometer signals and engineering additional timing and frequency-based features, removing features that were overly dependent on raw amplitude and replacing the PCA/K-Means approach with methods better suited for nonlinear structure and outlier detection.

III. Features

A. Feature Filter

A noise-floor threshold was introduced during preprocessing to distinguish active boiling signals from runs with little or no boiling activity. Runs that did not exceed this threshold were excluded from the main clustering workflow, reducing the likelihood of noise-driven clustering behavior. In this pipeline, the threshold was set to an amplitude of 0.1; therefore, all runs were prescreened to ensure that they reached an amplitude greater than 0.1 at least once during the measurement period.

B. Feature Engineering

We use feature engineering to extract more information from each boiling run to produce more distinguishable and interpretable clusters. Our focus was on designing features that capture signal structure (such as frequency and variability) that depend less on amplitude. We also implemented two pairwise features: the number of peak magnitudes with high differences and the correlation between peak magnitudes. This way, differences in signal data between the two accelerometers within a run could be used for clustering. We engineered 36 features from both time and frequency domains for a total of 70 features, 34 on each accelerometer and 2 pairwise features.

To calculate the spectral features, the signal is first high-pass filtered using a Butterworth filter to remove low-frequency drift and baseline variation. The frequency content is then estimated using the Fast Fourier Transform (FFT) and Welch's method for power spectral density (PSD) estimation. Features are extracted from the PSD to describe how the signal's energy is distributed across frequencies over time. In this work, the spectral features include the mean spectral entropy and the variance of the spectral centroid. Spectral entropy measures how spread out or concentrated the signal power is across frequencies, while the spectral centroid represents the "center of mass" of the frequency spectrum. Together, these features capture changes in the complexity and dominant frequency behavior of the boiling signal.

We compute the skewness and kurtosis of each boiling run. Skewness helps us determine the symmetry of the signal, while kurtosis measures the heaviness of the distribution at the tails and the prevalence of extreme values compared to a normal distribution. We also calculate the skew, kurtosis, and inter-quartile range of just the peak magnitudes. We calculate the root mean square change points, which measure how often the overall energy of the signal changes significantly over time. We do this by finding the root mean square and finding how many times a change in root mean square is greater than two standard deviations. We calculate the burstiness of a signal by dividing the standard deviation of inter-peak times by the mean of inter-peak times. This helps us measure how irregular the signal is, as higher values of burstiness correspond to the run having periods of rapid activity followed by slower periods. Crest factor is calculated by dividing the max peak amplitude by the root mean square of the signal. Crest factor tells us how the max peak compares to the overall energy of the signal. We use a wavelet transform with the Morlet wavelet to calculate the low, medium, and high wavelet scales to see how much of a signal's energy lives in each scale. The high wavelet scale corresponds to lower frequencies.

A student from the previous group developed an algorithm that detects the number of boilings, the unused peak

proportion, and the regime dominance. The number of boilings corresponds to the number of detected boiling patterns in the signal. A value of one for the number of boilings means the signal is single rhythmic, whereas a value of two would mean the signal is double rhythmic.

Because runs vary in length, we checked whether any features were correlated with the duration of runs, which could lead to clustering that depends on time instead of signal behavior. We calculated correlations between each feature and the length of a run and found that multiple features are highly correlated with time (see Appendix for a full list of features). To remedy this problem, we used detrending algorithms. The detrending process removes linear trends from each selected feature to ensure that clustering is not biased by systematic changes over time. A linear regression model is fit between each feature and a time variable, and the resulting trend is subtracted from the original feature values. This way, we removed the linear effects of runtime from each feature.

IV. Modeling

A. Introduction

Now that we have introduced the data, it is time to move on to the modeling methods. Our goal is to utilize an unsupervised machine learning algorithm to cluster our data into boiling regimes. To complete this task, we need to utilize a variety of models and techniques, and that process is examined in this section.

B. Dimension Reduction

Given that our data has 70 total features spread across two accelerometers, our data lies within 70 dimensions. This can be a problem for two main reasons. The first one is the curse of dimensionality. As more features are added to a dataset, the "volume" of the space increases so drastically that the available data becomes extremely sparse. The points are quite literally moving further away from one another, causing the concept of distance between points to lose its meaning. If distance becomes meaningless, unsupervised machine learning clustering algorithms that utilize distance metrics can no longer find patterns or group data reliably. The second problem is that we cannot visualize our clusters. The human mind can only comprehend three dimensions, and this means we cannot display our results to our physicist at NASA, Michael Khasin. As we are a team of data scientists with no background in rocket science, it is increasingly important to easily display our results for interpretation. There also lies a third subreason, and that is noise. Our data contains quite a bit of noise between points, making many points overlap, which leads to fuzzy groupings. You can approach this problem by either applying a noise reduction technique or, in our case, utilizing a dimension reduction technique.

Uniform Manifold Approximation and Projection (UMAP) is the dimension reduction technique we chose for our initial modeling. A weighted neighborhood graph is first constructed to represent which observations are most similar in the original high-dimensional space. Each observation is connected more strongly to nearby observations and more weakly to distant observations. A lower-dimensional embedding is then initialized and adjusted so that the neighborhood relationships from the original graph are preserved as closely as possible. The final embedding places similar observations near one another while separating observations with weaker or less consistent relationships. The previously used dimension reduction technique, PCA, is a linear dimensionality reduction method that transforms correlated variables into new uncorrelated components ordered by the amount of variance explained. Compared to PCA, UMAP is often stronger for complex datasets because nonlinear relationships and local neighborhood structure can be preserved more effectively.

UMAP has two main parameters to tune. There is a `number of neighbors`, which determines how much local versus global structure is emphasized, and `minimum distance`, which controls how tightly points are allowed to cluster in the final embedding. When conducting our initial testing, it was important to test random parameter values to get an understanding of the general global structure. Since UMAP reduces the data from 70 to 3 dimensions, we were able to get a visualized reference for each run. After getting a general feel for the data on several different computers, we noticed three main clusters that persisted (see figure 7). Now that the UMAP parameters were tuned, we moved on to clustering.

C. Clustering

Now that UMAP grouped similar points, reduced noise, and mapped our data into three dimensions, we can move on to our clustering algorithm. UMAP takes on the majority of the work when it comes to our modeling. The clustering algorithm must be able to accomplish two tasks. First, it has to be able to identify the groupings of data created by UMAP. Second, it must not have a parameter to preset the number of clusters. This step is important as it allows the clustering algorithm to adapt seamlessly to any changes made to the UMAP.

Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) is the clustering algorithm chosen for the main modeling process. A density-based structure is first constructed to identify regions where observations are closely packed together in the feature space. Dense regions are treated as potential clusters, while sparse regions are treated as boundaries or noise. A hierarchy

of possible clusters is then built across different density levels, allowing stable clusters to be selected without requiring the number of clusters to be specified in advance. The final clustering assigns observations to groups with similar density structure while leaving uncertain or weakly connected observations unassigned as noise. The previously used the clustering algorithm, K-means, a centroid-based method that partitions observations into a fixed number of clusters by minimizing the distance between each observation and its assigned cluster center. Compared to K-means, HDBSCAN is often stronger for complex datasets because irregular cluster shapes, varying densities, and noisy observations can be handled more effectively.

The "H" in HDBSCAN refers to "hierarchical," meaning that clusters are evaluated across multiple density thresholds rather than at one fixed density level. This improves on the popular model DBSCAN by reducing dependence on a single neighborhood-radius parameter and by selecting clusters that remain stable across the hierarchy.

HDBSCAN successfully found the obvious groupings of data that any person would be able to discern with the naked eye. What makes it truly special is the patterns and densities it picks up on as we tune the parameters. HDBSCAN has two main parameters. There is a `minimum cluster size`, which determines the smallest group of observations that can be considered a valid cluster, and a `minimum samples`, which controls how conservative the algorithm is when labeling points as noise. This is quite helpful in noisy data, as it prevents a few extraneous data points from being incorrectly assigned to the nearest cluster. Higher `minimum samples` values usually produce fewer clusters and more noise points, while lower values allow more observations to be assigned to clusters. As we lowered the minimum cluster size, we noticed clusters form within the original three groupings, known as subclusters (see figure 7 in the Results sections). HDBSCAN is able to discover and assign these subclusterings even among strange shapes and densities.

It is important to note that we tested unsupervised clustering without a dimension reduction algorithm. With all three clustering techniques, the result often failed to cleanly distinguish between boiling regimes. Noise and outliers are highly prominent in our feature set, meaning some kind of cleaning or reduction of noise must be present to effectively apply a clustering technique.

D. Combined Model

With the initial selection of UMAP + HDBSCAN done individually, we can now move on to testing the combined model. A prominent feature of UMAP is that it contains processes that are inherently stochastic. UMAP is stochastic because parts of the embedding process depend on random initialization and approximate optimization, so the same data and parameters may not always produce identical coordinates. Small numerical differences, such as floating-point rounding errors, package versions, hardware, and parallel processing behavior, can also change the optimization path. As a result, two different computers may produce slightly different UMAP embeddings even when the same settings are used. A fixed random seed improves reproducibility, but exact equality across systems is not always guaranteed. This raises the need to ensure UMAP contains some repeating pattern, thus, we need to test its stability.

The goal of this stability test is to determine whether the combined UMAP + HDBSCAN model produces consistent cluster assignments when the stochastic parts of UMAP are repeated many times. Since HDBSCAN is applied after UMAP, changes in the low-dimensional embedding can affect the final cluster labels, so the test evaluates whether the same general clustering structure is recovered across different random seeds. To test this, the full UMAP + HDBSCAN pipeline is run 100 times for each dataset size using different random seeds, and each run produces a separate set of cluster labels. The Adjusted Rand Index is then calculated for every possible pair of runs to measure agreement between the resulting cluster assignments, producing 4,950 pairwise ARI comparisons for each dataset size. After each round, one observation is removed from the dataset, and the same procedure is repeated, allowing stability to be measured as the number of observations decreases. The output shows that the mean pairwise ARI remains high near the full dataset size, indicating that the model produces largely repeatable cluster assignments across random seeds (see figure 4). Some fluctuation appears as observations are removed; however, the ARI values remain within a strong range across the displayed dataset sizes. These results provide evidence that the UMAP + HDBSCAN pipeline is reasonably stable and is capturing a repeatable clustering pattern rather than random variation, even as data size decreases.

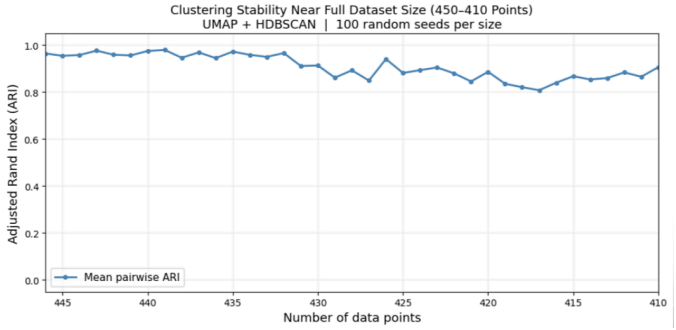


Figure 4: UMAP + HDBSCAN inherent stochastic stability testing.

E. Predictive Assignment of New Samples

The first stage of the modeling process has established a stable clustering structure for the available test data. Under the selected UMAP and HDBSCAN pipeline, observations are assigned to clusters in a manner that is both repeatable across random seeds and robust to moderate changes in dataset size. The next stage of the modeling process addresses the assignment of new incoming observations. In an operational spacecraft setting, repeatedly rerunning the full clustering pipeline as new data becomes available could alter the existing cluster structure and reduce interpretability over time. To avoid this issue, a predictive model can be trained to assign individual new observations to the previously identified clusters without requiring the full clustering process to be rerun.

A Gaussian Mixture Model (GMM) is a probabilistic clustering model that represents the data as a combination of multiple Gaussian distributions. Each Gaussian component is defined by a mean vector, covariance structure, and mixing weight, allowing each component to represent a different region of the feature space. Instead of assigning observations only by distance to a centroid, the model estimates the probability that each observation belongs to each component. This makes the method useful for predictive assignment because a new observation can be assigned to the component with the highest posterior probability.

The GMM is then tested to determine whether it can reliably reproduce the HDBSCAN cluster assignments in the fixed UMAP embedding space. This validation is necessary because the GMM must preserve the original cluster structure closely enough to serve as a stable prediction mechanism for new incoming observations. To test this, the observations are split into training and testing sets across 1,000 random stratified splits, with the GMM fit on the training observations and evaluated on the held-out observations. Since GMM component labels are arbitrary, the predicted labels are aligned with the original HDBSCAN labels before model agreement is measured using the Adjusted Rand Index (ARI). The resulting ARI distribution has a mean of 1.0 and a standard deviation of 0.0, indicating that the GMM perfectly reproduces the HDBSCAN cluster assignments across all tested splits (see figure 5). These results validate the GMM as an appropriate predictive approximation of the HDBSCAN clustering structure within the learned UMAP space.

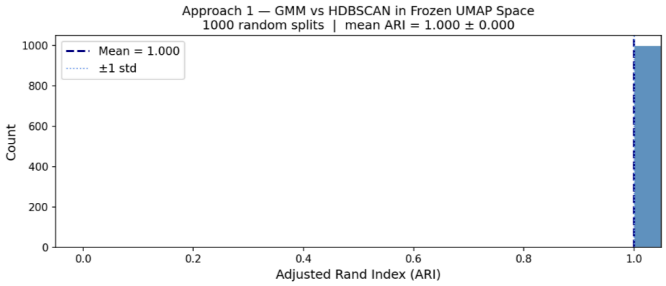


Figure 5: GMM replication of HDBSCAN boundaries and classification testing.

F. Complete Pipeline

The final model satisfies the two requirements established for this section. The UMAP + HDBSCAN pipeline identifies a stable clustering structure in the available data, and the Gaussian Mixture Model reproduces those cluster assignments with perfect agreement in the fixed UMAP space. This allows the discovered clusters to be retained as a consistent reference structure while new observations are assigned through the trained predictive model. The result is a modeling workflow that supports both unsupervised cluster discovery and stable assignment of incoming data without repeatedly altering the original clustering structure (see figure 6).

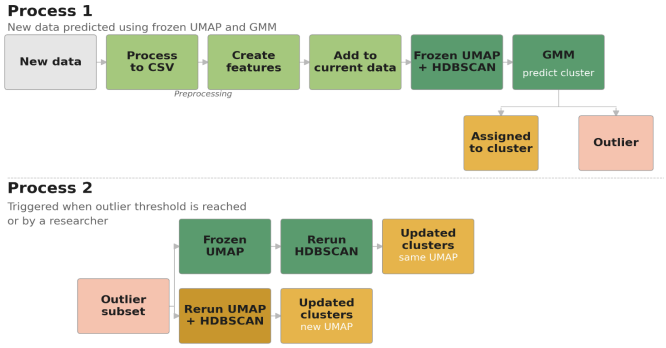


Figure 6: Workflow for processing new boiling runs through the frozen UMAP/GMM classification framework and new-regime discovery pipeline.

V. Results

The final clustering pipeline identified six interpretable boiling regimes from the acoustic accelerometer data. After removing non-active boiling runs through the noise-threshold filtering process, the remaining runs were clustered using UMAP for dimensionality reduction and HDBSCAN for cluster discovery. The first stage of clustering separated the active boiling runs into three main regimes: Sporadic, Chaotic, and Rhythmic. These clusters were interpreted using representative signal plots, feature comparisons, and visual inspection of the boiling behavior within each group.

A. Clusters: Sporadic, Chaotic, and Rhythmic

The Sporadic cluster represents boiling runs with less consistent activity and fewer repeating acoustic patterns. These runs tend to contain isolated boiling events rather than continuous or highly structured behavior. This regime was characterized by low high-frequency power, low mean spectral entropy over time, and a lower ratio between typical and maximum peak magnitude, suggesting less organized boiling activity and fewer dominant acoustic events. The Chaotic cluster represents runs with more irregular and variable activity. These signals contain stronger fluctuations and burst-like behavior, but they do not follow a clear dominant rhythm. This regime was characterized by a high unused peak proportion, high kurtosis, and high crest factor, indicating many irregular acoustic events along with occasional extreme peaks and sharp bursts of activity. The Rhythmic cluster represents runs with repeating acoustic patterns that suggest more structured boiling behavior. This regime was characterized by a high number of peaks per second, a large number of detected boiling patterns, and a low unused peak proportion, indicating frequent and organized boiling activity with strong repeating structure.

To evaluate the reliability of these interpretations, the most distinguishing features for each primary cluster were subjected to stability testing across 100 randomized trials. In each trial, a random subset of runs was removed, with progressively larger portions of the dataset excluded to evaluate sensitivity to missing data. The defining features remained consistent across all trials, demonstrating 100% stability for the main cluster interpretations. This indicates that the primary cluster interpretations were highly stable and not driven by individual runs or random variation.

Figure 7 shows the main clustering result obtained from UMAP and HDBSCAN. The three main clusters provide an interpretable separation of active boiling behavior into Sporadic, Chaotic, and Rhythmic regimes. The separation observed in the embedding space suggests that the engineered features capture meaningful differences in boiling behavior and acoustic structure.

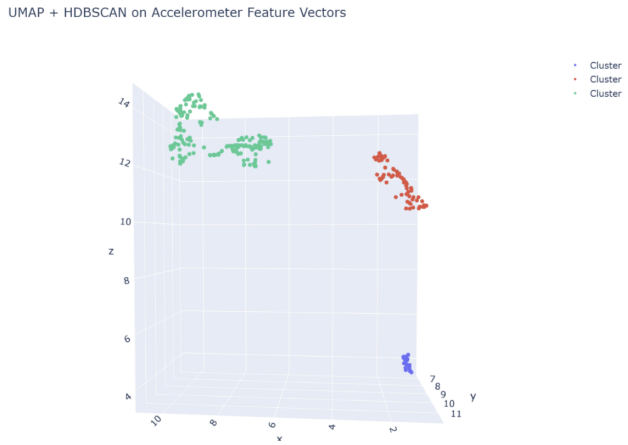
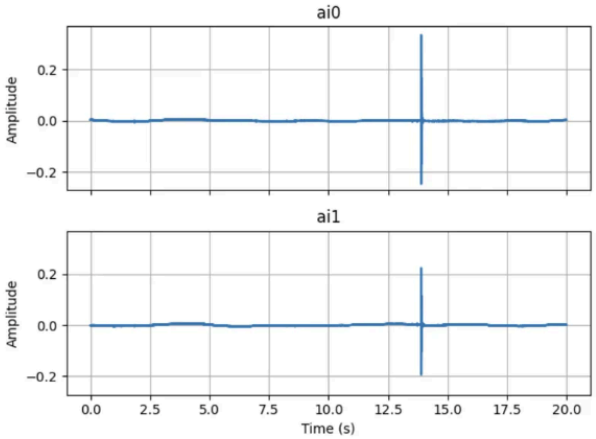
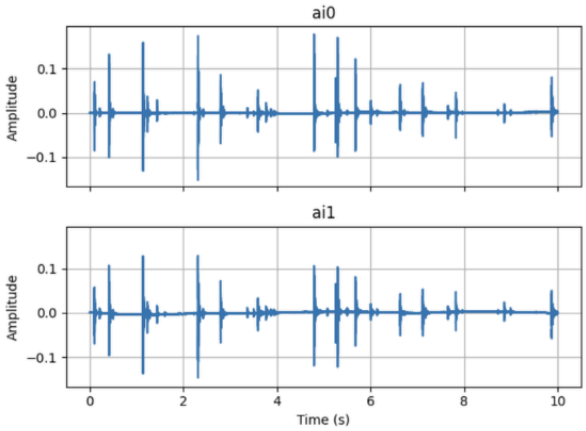


Figure 7: 3D UMAP and HDBSCAN clustering of main active boiling runs. Three clusters were identified: **Blue is Sporadic, Red is Chaotic, and Green is Rhythmic** boiling regimes.

(a) Sporadic



(b) Chaotic



(c) Rhythmic

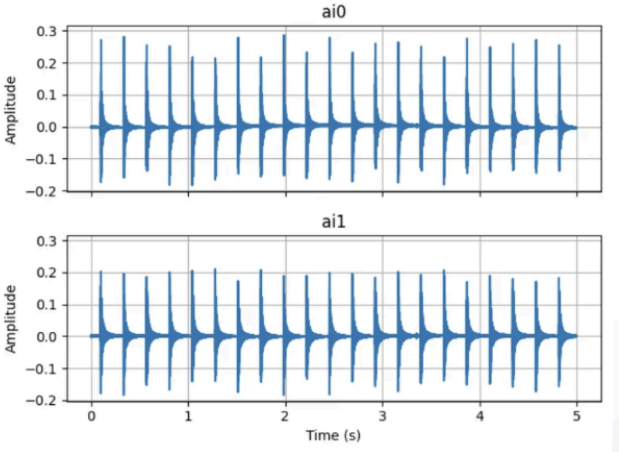


Figure 8: Representative acoustic signals from the three primary boiling regimes: (a) Sporadic, (b) Chaotic, and (c) Rhythmic.

The Rhythmic cluster contained multiple visually distinct signal patterns, thus we performed a second clustering step only on the rhythmic runs. This subclustering step identified three rhythmic sub-regimes: Dense Rhythmic, Double Rhythmic, and Single Rhythmic. These subclusters suggest

that rhythmic boiling contains multiple structured patterns that differ in event density, burst behavior, and dominant rhythm. Figure 9 shows the rhythmic subclustering result. The subclusters were interpreted using feature differences and run inspections.

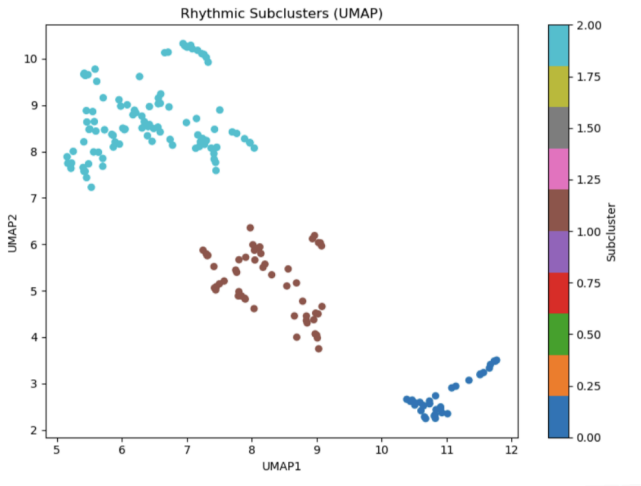


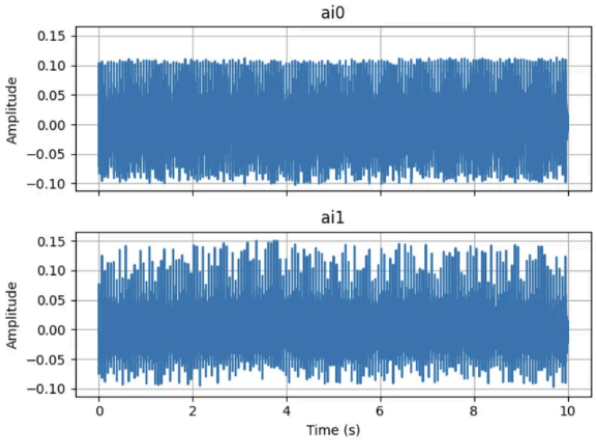
Figure 9: 2D UMAP embedding and HDBSCAN subclustering of the Rhythmic regime. Three rhythmic sub-regimes were identified: Cluster 0 is Dense Rhythmic, Cluster 1 is Double Rhythmic, and Cluster 2 is Single Rhythmic.

B. Subclusters: Dense, Double, and Single Rhythmic

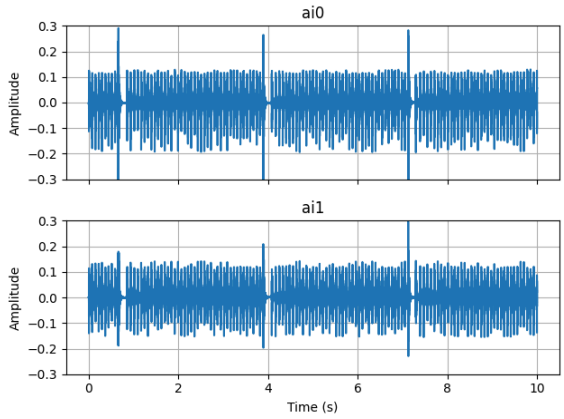
The Dense Rhythmic subcluster was characterized by strong, frequent, repeating acoustic patterns. This subcluster presented low spectral entropy, low spectral centroid, and high low-scale wavelet energy, indicating concentrated frequency content and frequent rapid acoustic activity. The Double Rhythmic subcluster contained many boiling events that tended to occur in bursts rather than following one dominant rhythm. This subcluster was characterized by high burstiness, a large number of boiling patterns, and low regime dominance, suggesting multiple competing rhythmic structures within the same run. The Single Rhythmic subcluster contained fewer boiling events and one dominant, repeating boiling pattern. This subcluster exhibited fewer peaks per second, fewer boiling patterns, and high regime dominance, indicating that most of the boiling behavior follows a single repeating rhythm.

A similar stability analysis was performed for the most distinguishing features of the rhythmic subclusters. Across 100 randomized trials, progressively larger random subsets of runs were removed. The defining features remained highly consistent, with all three subclusters maintaining greater than 80% feature stability. These results show that the discovered sub-regimes represent meaningful and reproducible boiling behaviors. Representative examples of each rhythmic sub-regime are shown in Figure 10.

(a) Dense Rhythmic



(b) Double Rhythmic



(c) Single Rhythmic

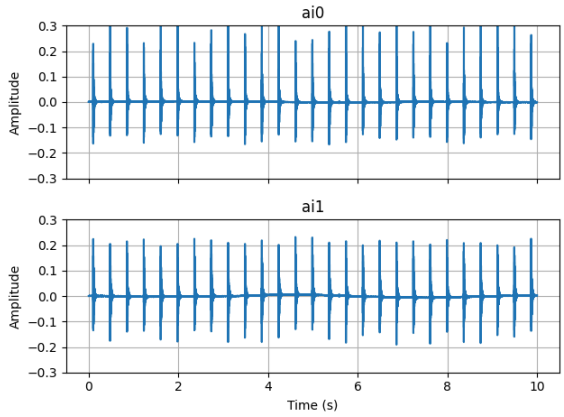


Figure 10: Representative boiling runs from the (a) Dense Rhythmic, (b) Double Rhythmic, and (c) Single Rhythmic sub-regimes.

C. Sub-regime Feature Separation

We analyzed the Dense, Double, and Single Rhythmic subclusters to better understand what features best separate these three groups. The best differentiator between the three is peaks per second (see Figure 11). The Single Rhythmic group has the lowest peaks per second, with the least variability. The Double Rhythmic group had the second-highest peaks per

second but had slightly higher variability. The Dense subcluster had the highest variability in peaks per second, but still tended to have higher values than the single and double rhythmic subclusters.

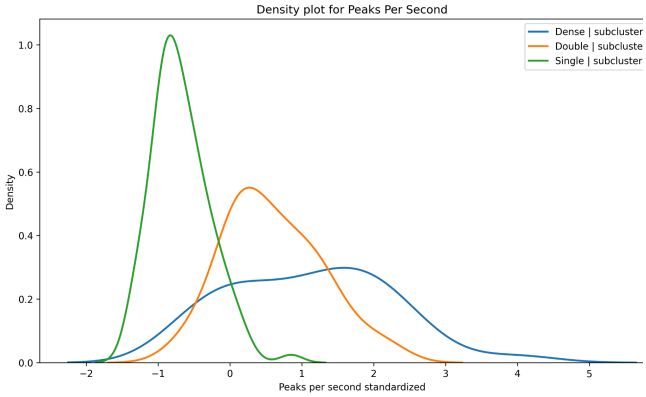


Figure 11: Density Plot of Peaks per second across the Dense Rhythmic, Double Rhythmic, and Single Rhythmic subclusters.

The second best differentiator is crest factor (see Figure 12). Crest factor tells us how the max peak compares to the overall energy of the signal. The Dense subcluster had the lowest crest factor, and the Single Rhythmic had the highest crest factor, with Double Rhythmic in the middle. This means the Dense subcluster tended to have more moderate max peaks compared to the rest of the signal.

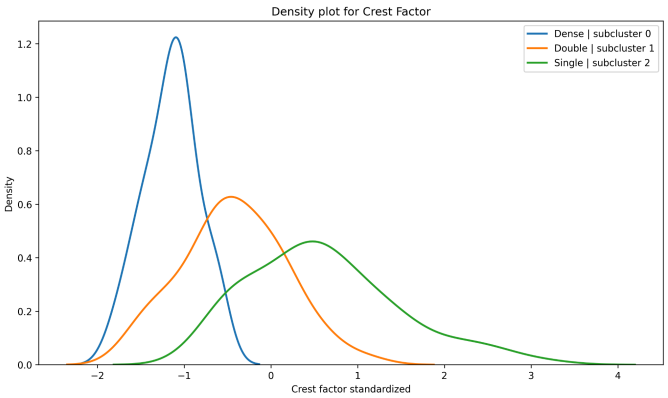


Figure 12: Density Plot of Crest factor across the Dense Rhythmic, Double Rhythmic, and Single Rhythmic subclusters.

Another good differentiator is burstiness (see Figure 13). Burstiness is the standard deviation of inter-peak times divided by the mean of the inter-peak times. A signal that has low times between peaks on one part of the signal and a few spread-out peaks with lots of inter-peak time on another part of the signal will have both a low mean inter-peak time and a high standard deviation of inter-peak times. This means the signal will have high burstiness. Since the Dense subcluster has consistently low burstiness, this means that its inter-peak times are more consistent. This makes sense for a dense signal, as many peaks with low mean inter-peak times would not have a high standard deviation of inter-peak times and therefore would have a low burstiness.

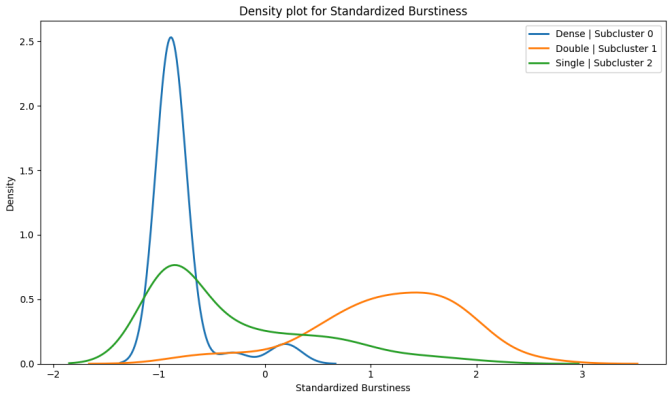


Figure 13: Density Plot of Burstiness across the Dense Rhythmic, Double Rhythmic, and Single Rhythmic subclusters.

D. Cluster validation

One way of assessing cluster stability is by testing to see if similar clusters can be achieved with fewer boiling runs. We assume that if similar clusters can be found with less data, then our clustering approach is relatively stable and robust.

We tested this by taking out a certain number of points from the full data set and then running UMAP and HDBSCAN on the reduced data set with the same hyperparameters to get cluster labels for the reduced data. We then compared the cluster labels from the reduced data to the labels of the same points in the full data set using the RAND index by calculating a similarity score. We repeated this process fifty times, taking different observations out each time, and computed the mean RAND index for the N number of points taken out.

For the Chaotic, Sporadic, and Rhythmic clusters, the mean RAND index remained around 90% for up to 70 runs removed out of 284 runs (see Figure 14). We found similar results for the rhythmic subclusters, which had a nearly 80% RAND index for up to 100 runs removed, out of 189 runs (see Figure 15). This supports that UMAP and HDBSCAN form similar clusters with fewer observations.

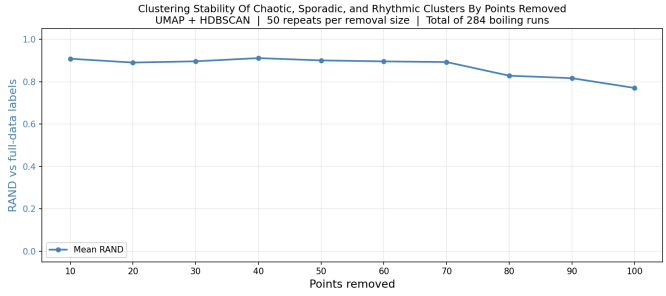


Figure 14: Mean RAND index for the main clusters as increasing numbers of runs were removed.

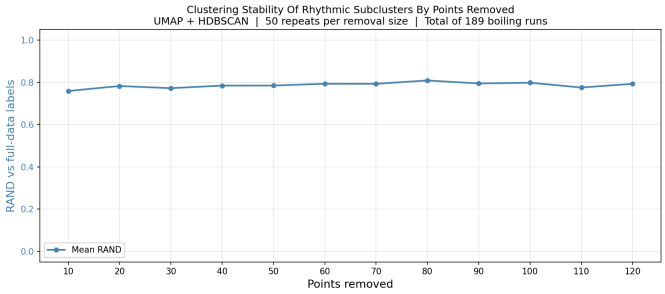


Figure 15: Mean RAND index for the rhythmic subclusters as increasing numbers of runs were removed.

In the setup for the experiment, the test was run on a different computer from the one that had produced the original full labels. This is important because when UMAP is run on one machine, it will not look the same as the same UMAP run on another. This means that the test actively incorporates machine-to-machine variability. Therefore, our UMAP and HDBSCAN parameters can be used to find similar clusters, even with less data available, on a different machine.

VI. Discussion

We have demonstrated a clear distinction between rhythmic and non-rhythmic boiling using our methods. The subclustering results suggest that rhythmic boiling is not a single behavior but instead consists of three distinct sub-regimes: Dense Rhythmic, Double Rhythmic, and Single Rhythmic. These sub-regimes were supported by both representative signal inspection and differences in key acoustic features. Together with the primary Sporadic, Chaotic, and Rhythmic clusters, the final pipeline identified six interpretable boiling regimes.

To evaluate the robustness of the clustering framework, stability testing was performed using leave-one-out experiments, dataset-size testing, and signal amplitude testing. These analyses showed that the discovered regimes remained reasonably consistent under multiple sources of variation, suggesting that the clustering structure captures meaningful boiling behavior rather than artifacts of a particular dataset or parameter selection.

In addition, a Gaussian Mixture Model (GMM) classification framework was developed and integrated into an application for assigning new boiling runs to the discovered regimes. By combining the GMM with a frozen UMAP embedding, new data can be projected into the existing clustering space and assigned to the most likely boiling regime while also providing classification probabilities.

Future work will focus on refining the interpretation of the discovered regimes, determining classification thresholds for identifying potential new boiling behaviors, and evaluating how well the framework generalizes to future experimental datasets.

VII. Limitations

This project had several limitations. The true number of boiling regimes was not known in advance, and there were no ground-truth labels for the data. Because of this, the clustering results could not be evaluated using traditional accuracy metrics. Instead, the clustering ability had to be evaluated using comparison metrics, visual inspection, and interpretation of the acoustic plots. Some of the observed differences between clusters may also be affected by experimental conditions that were not fully captured in the feature set. In addition, deciding whether a subgroup was physically meaningful still required human review of the plots and cluster behavior.

Another limitation was the experimental setup. During the experiments, the accelerometer placement was randomized relative to the heater position. Thus, signal amplitude could vary between runs even when the underlying boiling regime was similar. These amplitude differences may have affected the model's ability to cluster runs based on regime behavior, since two similar runs could appear different due to the sensor placement rather than actual differences in boiling behavior. Similarly, the recording length was not consistent across all runs. Shorter recordings may only capture a small portion of the signal, so a run could appear sporadic even if a rhythmic pattern would have been visible over a longer period of time.

Lastly, by using a dimension reduction algorithm, we were able to visualize our clusters in 3 dimensions. This was important for understanding the data and providing interpretable results. However, reducing the data from a high-dimensional feature space into three dimensions can lead to a loss of information. Subtle patterns in the original features may be compressed, distorted, or left out of the final visualization.

VIII. Future Directions

There are several directions this project could take beyond the approach our team chose. In our current pipeline, we used the full dataset to extract handcrafted features from each run, then used those features as inputs to our clustering and classification workflow. This approach made the clusters more interpretable because individual features could be connected back to physical signal characteristics, such as peak timing, spectral behavior, burstiness, and cross-channel relationships.

Another direction this project could have taken is to use a 1D Convolutional Autoencoder, or 1D-CNN autoencoder. This type of unsupervised deep learning model is designed for one-dimensional sequential data and can learn compressed representations of raw time-series signals automatically. A 1D-CNN autoencoder could potentially discover patterns that are difficult to capture with manually engineered features, and it could also be useful for anomaly detection. Our project did not focus on this method because we prioritized physical interpretability, but it could be used for comparing learned representations against our engineered-feature approach.

Finally, future work could include additional validation of the cluster labels. Since the current labels are based on

unsupervised patterns in the data, repeated experiments and reviews would help determine whether the clusters correspond to meaningful physical behaviors. This would bolster the connection between our model outputs and the underlying physical causes.

IX. References

1. M. Khasin, J. Rogers, and V. Osipov, "Acoustic Data-based Characterization of Incipient
Boiling for Space Applications," NASA Technical Memorandum NASA/TM–20250009668,
September 2025.

2. Gupta, K., et al., "Interpretable Machine Learning for Acoustic Classification of Incipient Boiling Regimes," NASA Technical Memorandum NASA/TM–20250011359, Dec. 2025.

3. M. Khasin, "Acoustic and Visual Data for Incipient Boiling at Local Heat Leaks in a Water
Tank (v1.0)," NASA Intelligent Systems Division Datasets, NASA Ames Research Center, last updated Nov. 25, 2025. Available:
https://www.nasa.gov/intelligent-systems-division/datasets/

X. Appendix

Features Subject to Detrending

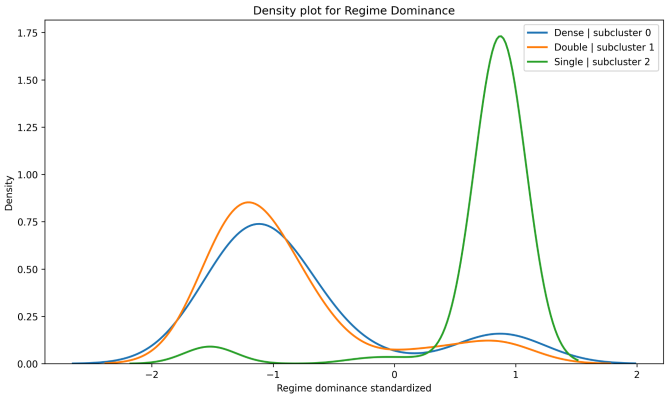
The following features were selected for detrending during preprocessing:

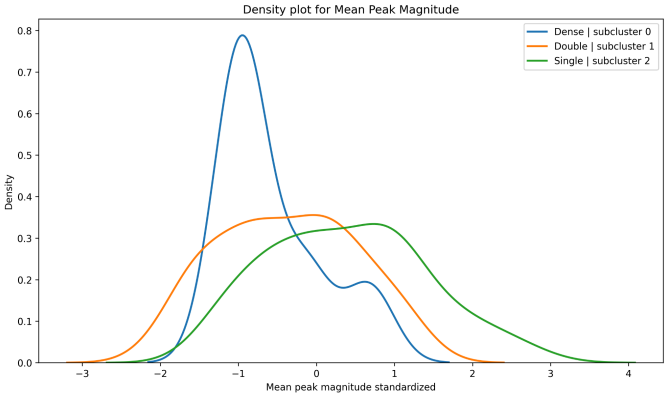
1. RMS change points
2. High to low ratio
3. Spectral flatness
4. Peaks per second
5. Number of boilings
6. Mean time difference
7. Median time difference

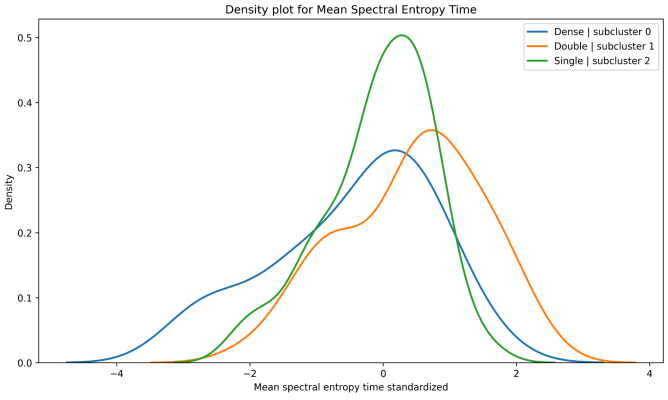
Link to Features - Feature Explanations

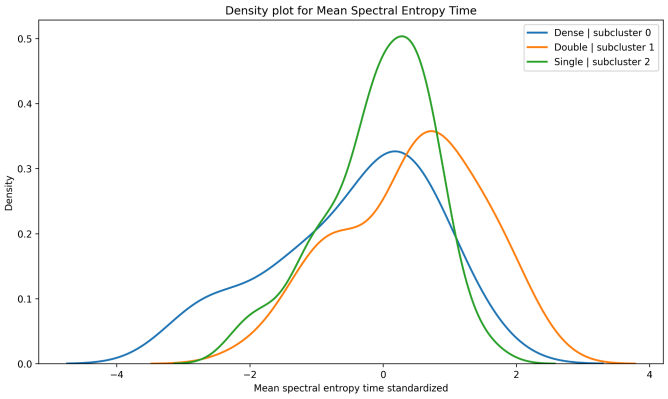
Slides: https://canva.link/g501eiqloans8we

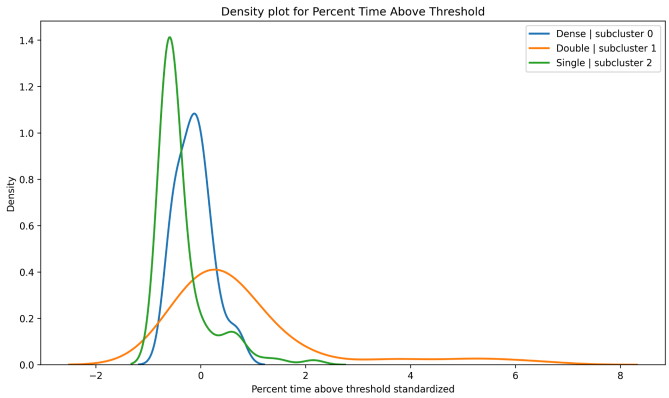
Select density plots for Dense, Double, and Single rhythmic sub-regime features:

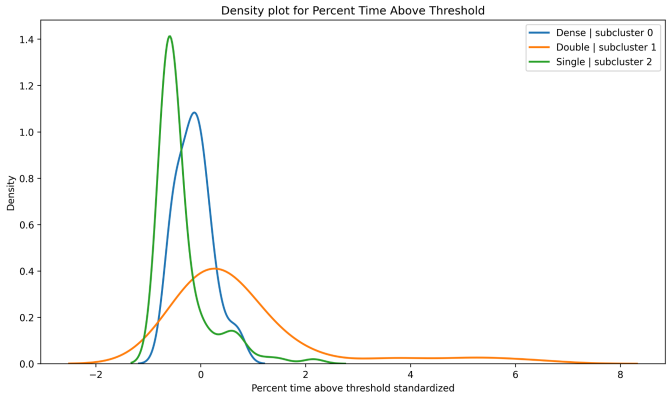


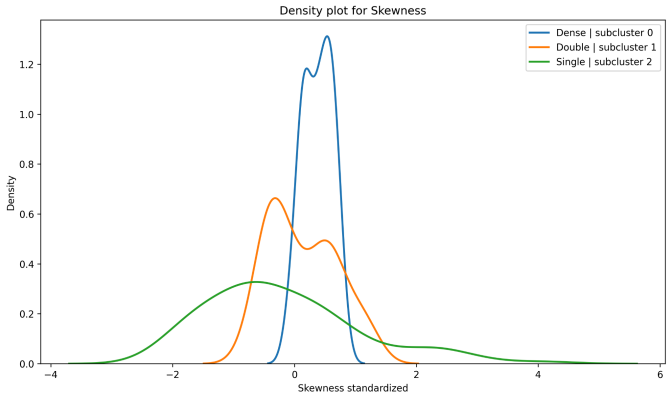


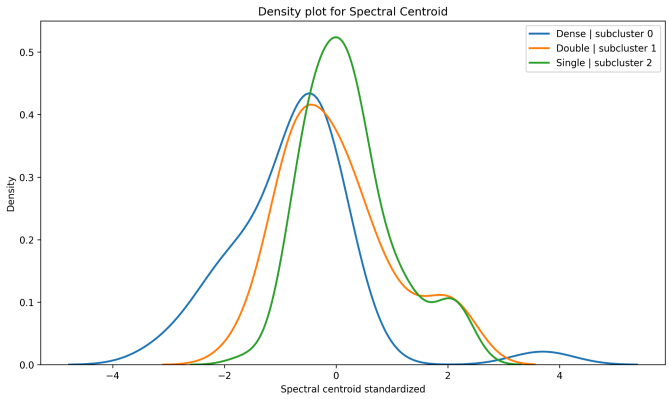


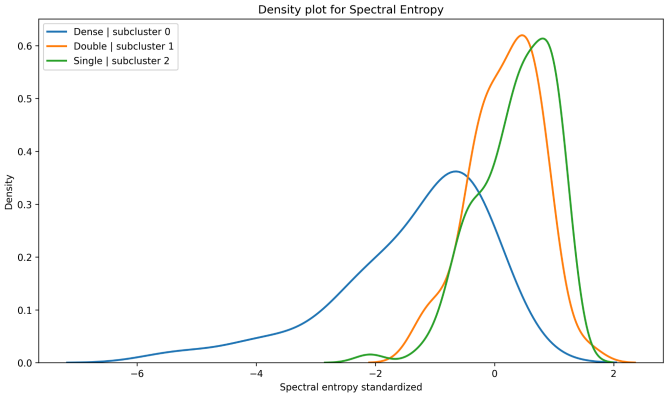


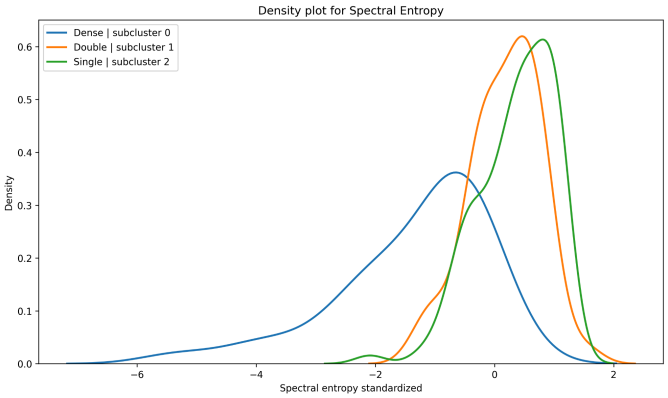


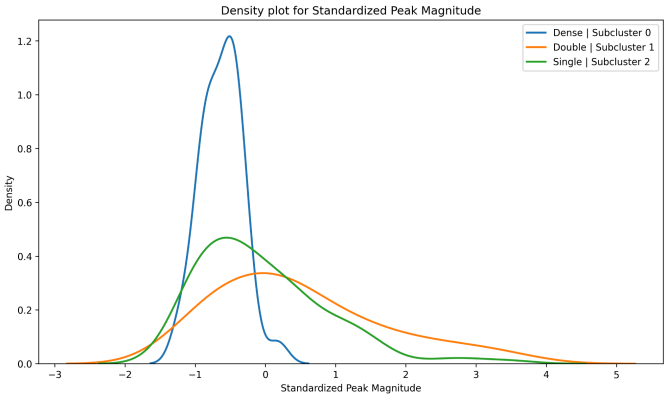







Density plot for Spectral Entropy
Dense | subcluster 0
Double | subcluster 1
Single | subcluster 2
Density
Spectral entropy standardized


Density plot for Spectral Entropy
Dense | subcluster 0
Double | subcluster 1
Single | subcluster 2
Density
Spectral entropy standardized


Density plot for Standardized Peak Magnitude
Dense | Subcluster 0
Double | Subcluster 1
Single | Subcluster 2
Density
Standardized Peak Magnitude