Part 1

I. Code Updates for 3D Data and Visualization

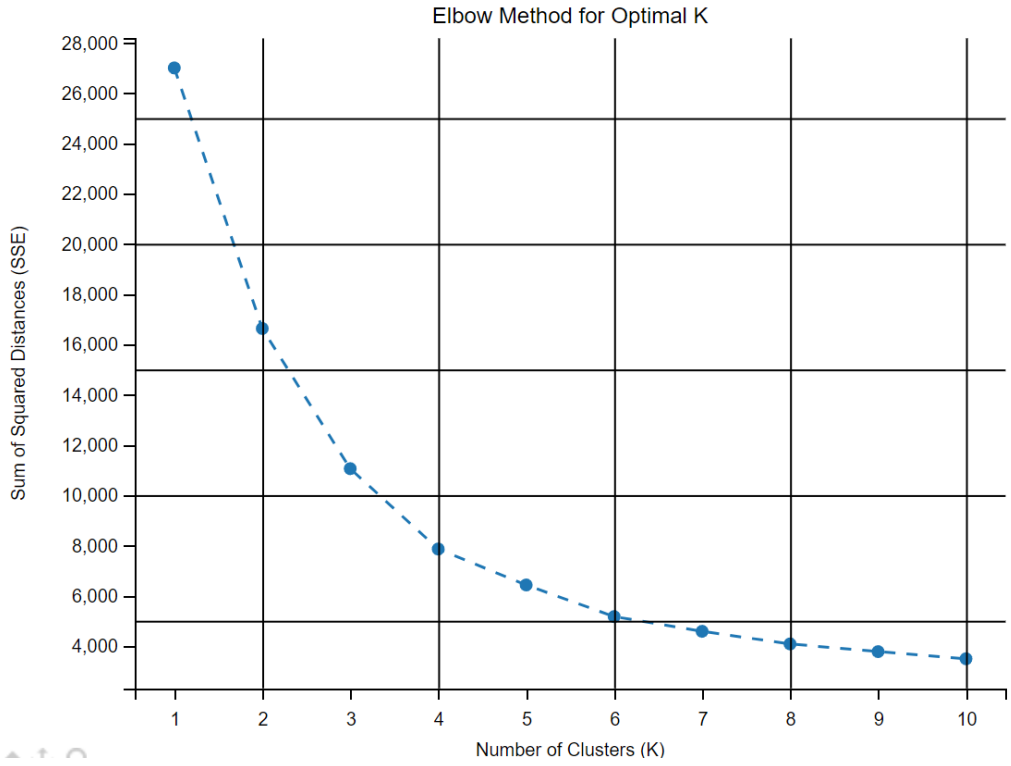
1. **Data Reading and Selection**:
    - The code reads the Spotify_YouTube.csv file using pd.read_csv("Spotify_YouTube.csv"). It then selects three columns Liveness, Energy, and Loudness from the dataset into a new dataframe X using X = data[["Liveness", "Energy", "Loudness"]]. This is to prepare the data for 3D clustering.
2. **Data Standardization**:
    - Since the features might have different scales, it's important to standardize the data.
      The StandardScaler from sklearn.preprocessing is used. The code first creates a scaler object scaler = StandardScaler() and then applies it to the data X_scaled = scaler.fit_transform(X). This ensures that each feature has a mean of 0 and a standard deviation of 1, which helps in the performance of the K - Means algorithm.
3. **Elbow Method for Optimal K**:
    - A loop is used to calculate the Sum of Squared Distances (SSE) for different values of K from 1 to 10 (max_k = 10). For each value of K, a KMeans object is created, fit to the scaled data, and the inertia_ (SSE) value is appended to the sse list. Then, a plot is created with the number of clusters (K) on the x - axis and the SSE on the y - axis. This is called the elbow method, which helps in visually determining the optimal number of clusters.
4. **3D Visualization**:
    - After determining the optimal K (in this case, K = 3), the KMeans algorithm is run with this value on the scaled data. The predicted cluster labels are obtained using y_km = kmeans.fit_predict(X_scaled).
    - To create a 3D visualization, a 3D subplot is added to the figure using ax = fig.add_subplot(111, projection='3d'). Different colors and markers are defined for each cluster. For each cluster, the data points belonging to that cluster are plotted using ax.scatter(). The cluster

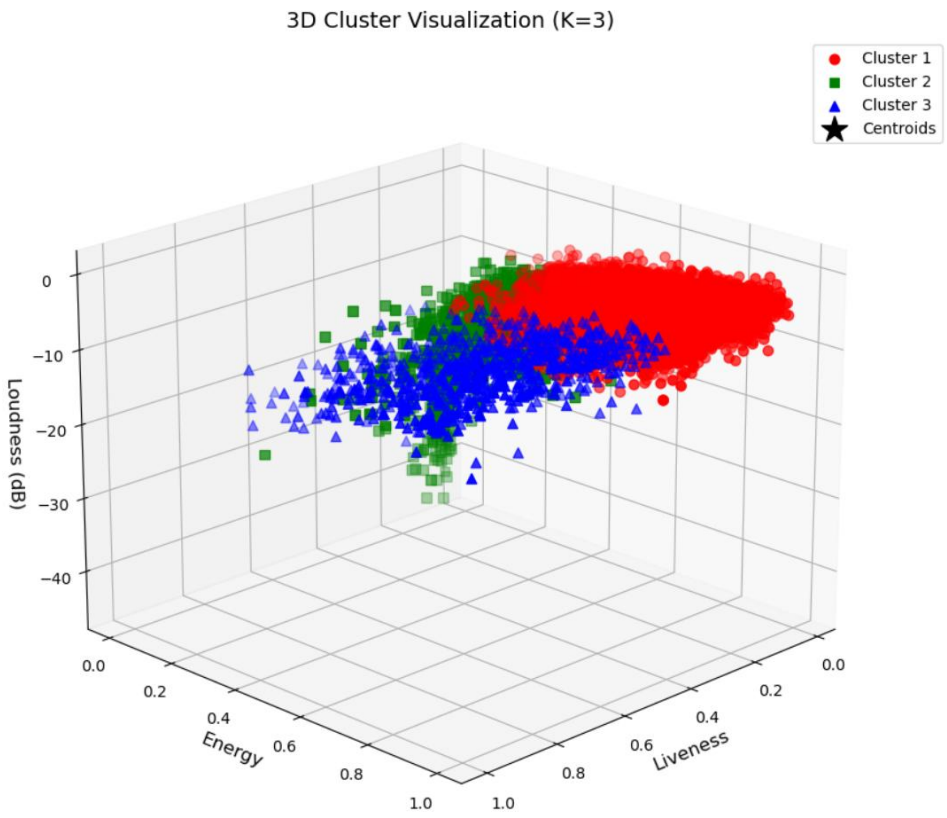
centers, which were initially calculated on the scaled data, are converted back to the original scale
using scaler.inverse_transform(cluster_centers_scaled) and then plotted as well. Appropriate axis labels, a title, and a legend are added to make the visualization clear.

II. Optimal K

The optimal number of K found using the elbow method is K = 3. In the elbow plot, the SSE value drops sharply as K increases from 1 to 3, and then the rate of decrease slows down significantly. The "elbow" in the plot indicates that K = 3 is a good choice as adding more clusters does not lead to a proportionally large reduction in SSE.

III. Graph Explanation





1. **Elbow Method Graph**:
    - The x - axis represents the number of clusters (K), ranging from 1 to 10. The y - axis represents the Sum of Squared Distances (SSE). As K increases, the SSE generally decreases because adding more clusters allows the algorithm to better fit the data. However, the goal is to find the point where adding more clusters does not significantly improve the fit. In this case, the sharp drop in SSE stops around K = 3, indicating that K = 3 is the optimal number of clusters.
    - The elbow method graph shows the relationship between the number of clusters (K) and the Sum of Squared Distances (SSE). SSE measures how well the data points are clustered around their respective centroids. A lower SSE indicates a better fit of the clustering model to the data.
    - As the number of clusters K increases, the SSE generally decreases because more clusters allow for a closer approximation of the data distribution. However, we are looking for the "elbow" point where adding additional clusters no longer significantly reduces the SSE. In this case, when K reaches 3, the rate of decrease in SSE slows down considerably. This suggests that using 3 clusters strikes a good

balance between fitting the data well and avoiding over - clustering (using too many clusters). In essence, 3 clusters can capture the major groupings in the data without overly complicating the model.

2. **3D Cluster Visualization Graph**:
    - The x - axis represents the Liveness feature, the y - axis represents the Energy feature, and the z - axis represents the Loudness (dB) feature. Each data point in the graph represents a data entry from the Spotify_YouTube.csv dataset. The points are colored and marked according to the cluster they belong to (red for Cluster 1, green for Cluster 2, blue for Cluster 3). The black stars represent the centroids of each cluster. This visualization shows how the data is grouped into three clusters based on the three features (Liveness, Energy, and Loudness). It provides a visual understanding of how the K - Means algorithm has partitioned the data in a 3 - dimensional space.

Part 2

I. Hierarchical Clustering for Individual Columns

1. Methodology for Hierarchical Clustering

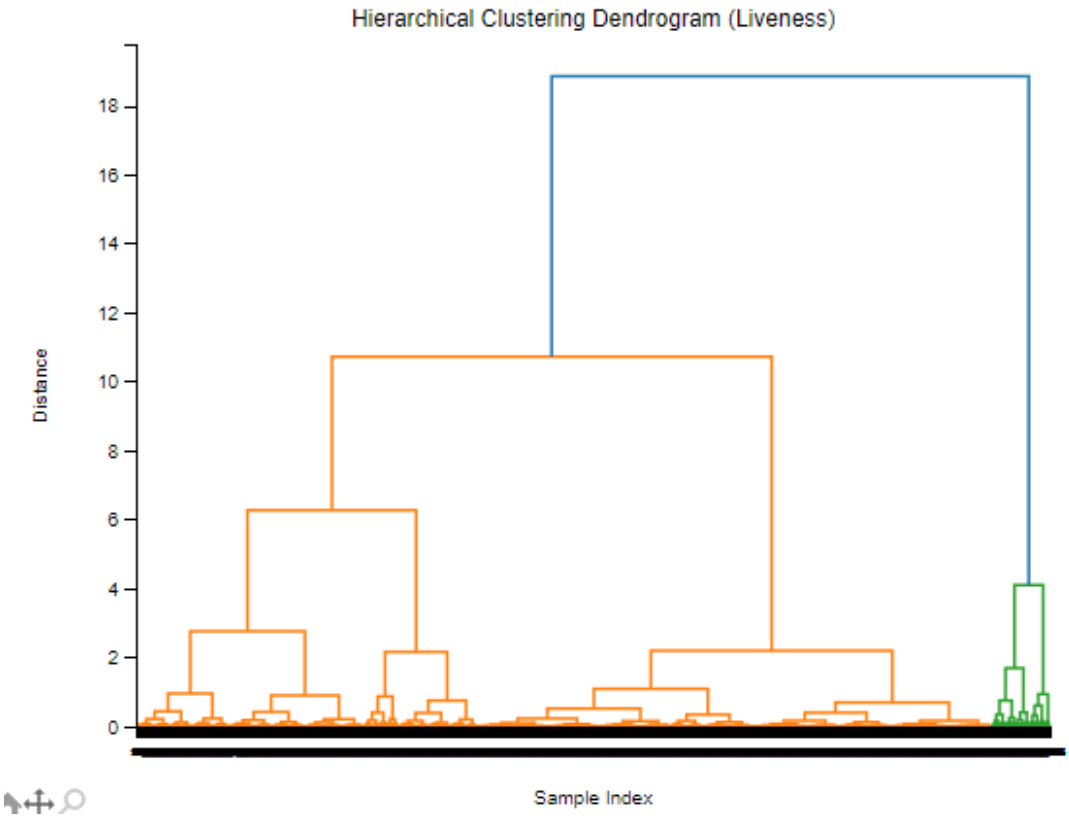
- **Dendrogram Generation**: For each feature column (Liveness, Energy, Loudness), the code first calculates the distance matrix using pdist(feature_data.reshape(-1, 1)). This matrix represents pairwise distances between data points for the specific feature. Then, hierarchical clustering is performed using the linkage function with the Ward method (by default), which aims to minimize the increase in within - cluster variance when merging clusters. The resulting linkage matrix Z is used to plot the dendrogram. The dendrogram is oriented with the top - down approach, labels are assigned to each sample index, and the distances are sorted in descending order. The show_leaf_counts option is set to True to display the number of samples in each leaf node. This dendrogram visualization helps in understanding the hierarchical merging process of data points and in determining the appropriate number of clusters.
- **Cluster Assignment and Analysis**: After observing the dendrogram to decide on the number of clusters (in this case, 3 for each feature), the AgglomerativeClustering class from sklearn.cluster is used. It is initialized with the specified number of clusters, Euclidean affinity (distance measure), and Ward linkage. The clustering algorithm is then fit to the scaled data

(scaled using StandardScaler), and cluster labels are predicted for each data point.

2. Results for Each Feature

Liveness

Dendrogram Interpretation: As seen in the dendrogram for Liveness, three main branches are evident. These branches indicate the formation of distinct clusters. The branching structure shows how data points with similar Liveness values are grouped together hierarchically.



- **Cluster Characteristics**:

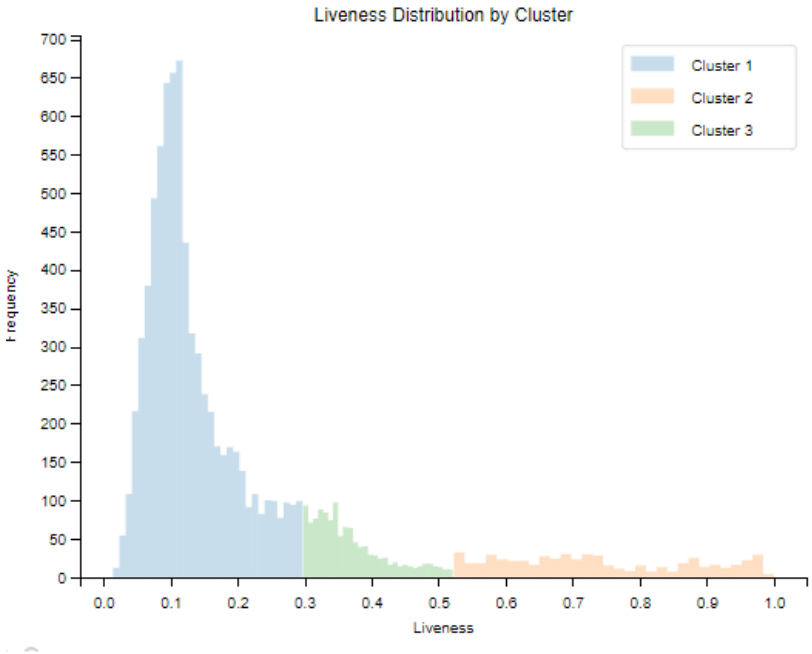
```
Cluster Characteristics for Liveness:
Cluster 1:
  Count: 7247
  Mean: 0.13
  Range: 0.01 - 0.30
  Std Dev: 0.06

Cluster 2:
  Count: 571
  Mean: 0.74
  Range: 0.52 - 1.00
  Std Dev: 0.14

Cluster 3:
  Count: 1181
  Mean: 0.37
  Range: 0.30 - 0.52
  Std Dev: 0.05
```

    - **Cluster 1**: With 7247 data points, it has a mean Liveness value of 0.13, ranging from 0.01 to 0.30, and a standard deviation of 0.06. This cluster represents data points with relatively low Liveness values. For example, in the context of music, these could be songs that sound less like they were recorded in a live performance setting.
    - **Cluster 2**: Comprising 571 data points, it has a mean of 0.74, a range of 0.52 - 1.00, and a standard deviation of 0.14. This cluster represents data points with high Liveness values, possibly indicating songs that have a strong live - performance - like quality.
    - **Cluster 3**: Having 1181 data points, it has a mean of 0.37, a range of 0.30 - 0.52, and a standard deviation of 0.05. This cluster represents data points with moderate Liveness values, falling between the other two clusters.

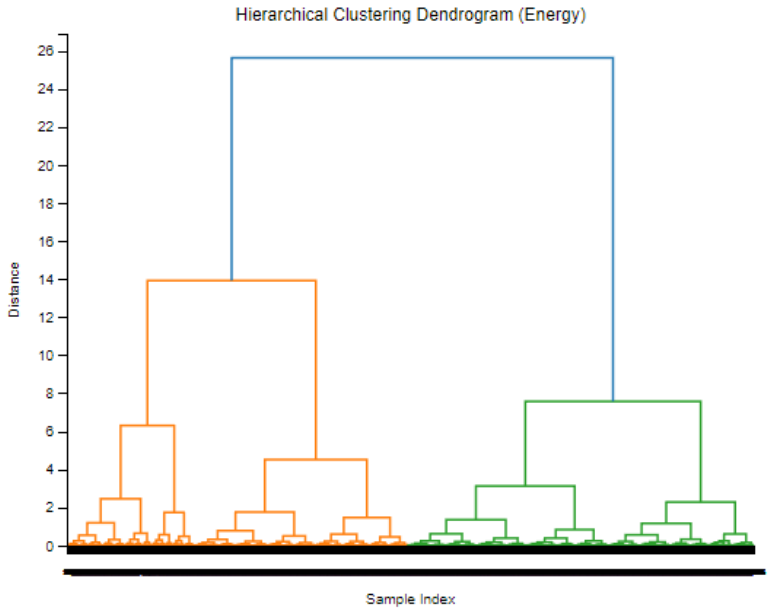
- **Cluster Distribution**:



The distribution plot for Liveness shows the frequency of Liveness values within each cluster. It clearly demonstrates the separation in the distribution of Liveness values among the three clusters. Cluster 1 has a peak at lower Liveness values, Cluster 2 at higher values, and Cluster 3 in the intermediate range.

Energy

- **Dendrogram Interpretation**:



The dendrogram for Energy also shows distinct branching, suggesting the presence of three separate clusters. The pattern of branching indicates how data points with similar Energy levels are aggregated hierarchically.

- **Cluster Characteristics**:

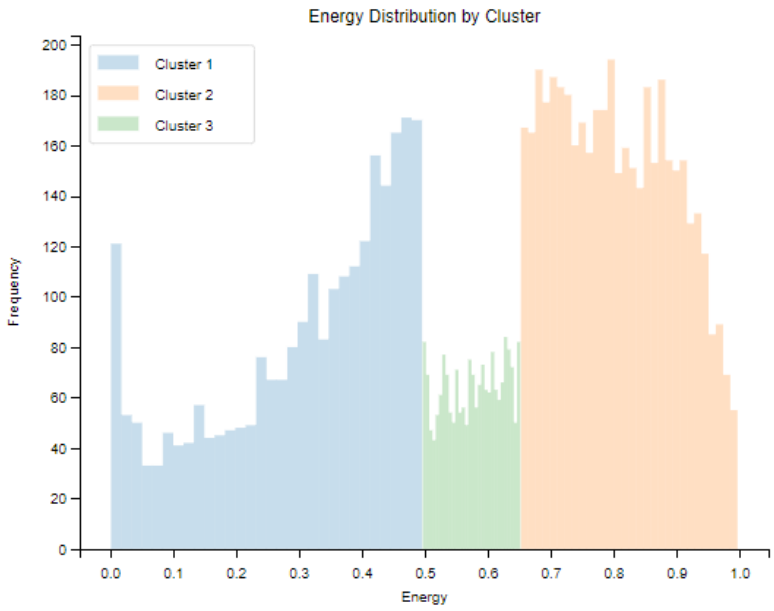
```
Cluster Characteristics for Energy:
Cluster 1:
  Count: 2532
  Mean: 0.31
  Range: 0.00 - 0.50
  Std Dev: 0.15

Cluster 2:
  Count: 4536
  Mean: 0.81
  Range: 0.65 - 1.00
  Std Dev: 0.09

Cluster 3:
  Count: 1931
  Mean: 0.58
  Range: 0.50 - 0.65
  Std Dev: 0.05
```

  - **Cluster 1**: With 2532 data points, it has a mean Energy value of 0.31, a range from 0.00 to 0.50, and a standard deviation of 0.15. This cluster corresponds to data points with low Energy levels, perhaps representing more mellow or subdued songs.
  - **Cluster 2**: Containing 4536 data points, it has a mean of 0.81, a range of 0.65 - 1.00, and a standard deviation of 0.09. It represents data points with high Energy levels, likely corresponding to energetic and lively songs.
  - **Cluster 3**: Having 1931 data points, it has a mean of 0.58, a range of 0.50 - 0.65, and a standard deviation of 0.05. This cluster represents data points with moderate Energy values.

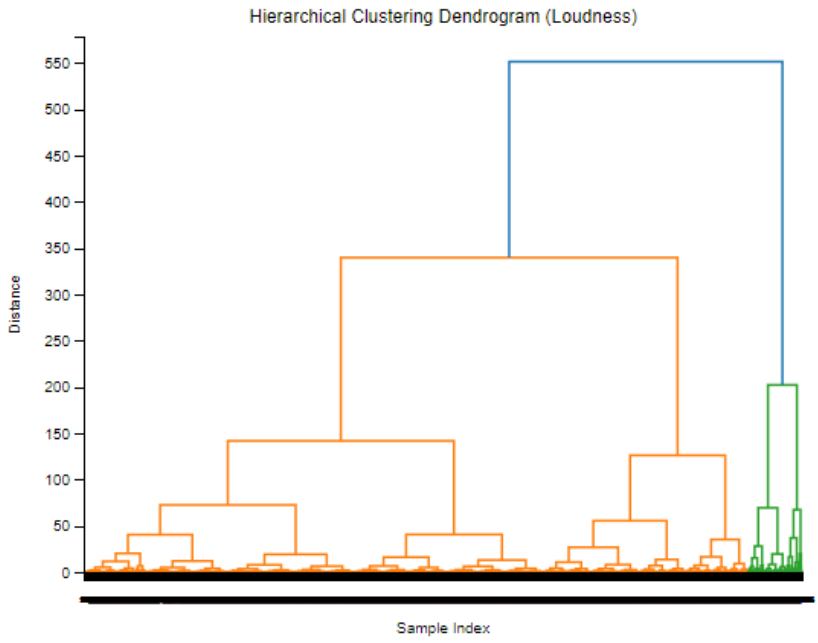
- **Cluster Distribution**:



The distribution plot for Energy illustrates the different ranges of Energy values within each cluster. Cluster 1 has a concentration of lower Energy values, Cluster 2 of higher values, and Cluster 3 in the middle range.

Loudness

- **Dendrogram Interpretation**:



The dendrogram for Loudness shows a hierarchical structure that can be

divided into three main clusters. The way the branches are formed indicates the hierarchical clustering of data points based on their Loudness values.

- **Cluster Characteristics**:

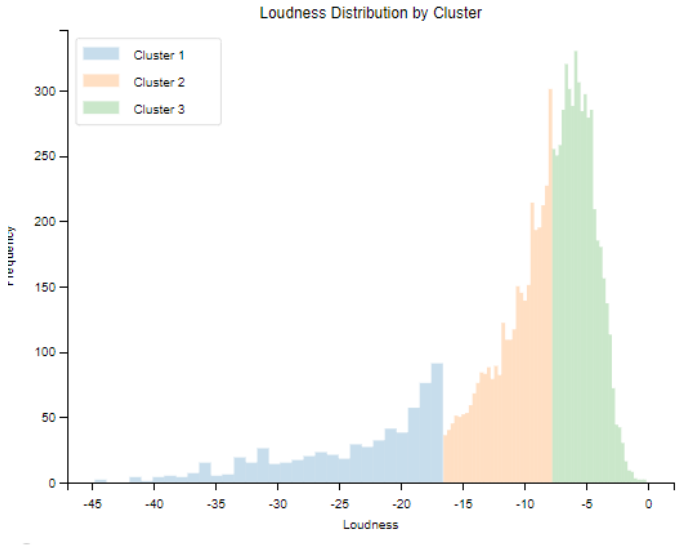
```
Cluster Characteristics for Loudness:
Cluster 1:
  Count: 632
  Mean: -23.65
  Range: -44.76 - -16.60
  Std Dev: 6.30

Cluster 2:
  Count: 3419
  Mean: -10.79
  Range: -16.55 - -7.75
  Std Dev: 2.35

Cluster 3:
  Count: 4948
  Mean: -5.44
  Range: -7.75 - -0.14
  Std Dev: 1.42
```

  - **Cluster 1**: With 632 data points, it has a mean Loudness value of -23.65 dB, a range from -44.76 to -16.60 dB, and a standard deviation of 6.30 dB. This cluster represents data points with very low Loudness levels, which could be soft - sounding songs.
  - **Cluster 2**: Comprising 3419 data points, it has a mean of -10.79 dB, a range of -16.55 to -7.75 dB, and a standard deviation of 2.35 dB. It represents data points with moderate Loudness values.
  - **Cluster 3**: Having 4948 data points, it has a mean of -5.44 dB, a range of -7.75 to -0.14 dB, and a standard deviation of 1.42 dB. This cluster represents data points with relatively high Loudness levels.

- **Cluster Distribution**:



The distribution plot for Loudness highlights the differences in Loudness levels among the three clusters. Cluster 1 has a concentration of lower Loudness values, Cluster 2 in the middle range, and Cluster 3 at higher Loudness values.

II. Discussion

The hierarchical clustering results for each individual column have successfully identified distinct groups within the Liveness, Energy, and Loudness features. These clusters can provide valuable insights into the distribution and characteristics of these audio - related features in the Spotify - YouTube dataset.

For example, in the context of music, the clusters for Liveness can help classify songs based on their perceived live - performance - like quality. The Energy clusters can distinguish between low - energy, high - energy, and moderately - energetic tracks, which is useful for music recommendation systems or playlist curation. The Loudness clusters can aid in understanding the volume characteristics of different songs, which may be relevant for audio engineering or user preferences regarding listening levels.

Overall, these findings can be further utilized in various applications such as music analysis, user behavior understanding, and content - based music retrieval systems.

III. Conclusion

Hierarchical clustering on the individual columns of Liveness, Energy, and Loudness has revealed distinct groups with identifiable characteristics. The appropriately labeled dendrograms and cluster distribution plots have provided a clear visual representation of the clustering results. The analysis of cluster characteristics has

offered meaningful interpretations of the data within each feature, contributing to a better understanding of the dataset's audio - related aspects.